

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«САХАЛИНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
ПОЛИТЕХНИЧЕСКИЙ КОЛЛЕДЖ

МЕТОДИЧЕСКИЕ УКАЗАНИЯ

для студентов по выполнению
практических работ
по дисциплине

ОП. 15 WEB-ПРОГРАММИРОВАНИЕ

укрупненная группа:

09.00.00 ИНФОРМАТИКА И ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА

специальность: 09.02.03 Программирование в
компьютерных системах

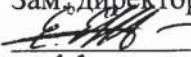
базовый уровень подготовки

Квалификация: техник-программист
Форма обучения: очная

Южно-Сахалинск
2020

УТВЕРЖДАЮ

Зам. директора по НМР

 Е.Н. Ермолаева

« 22 » мая 2020 г.

Разработчик: Вилькер Е.Ю., преподаватель информационных дисциплин

Одобрено на заседании ПЦК
информационных дисциплин

Протокол № 9 от «15» мая 2020г.

Председатель ПЦК

 О.Б. Куприяшина

СО Д Е Р Ж А Н И Е

№		Страница
1	ПОЯСНИТЕЛЬНАЯ ЗАПИСКА	6
2	ПЕРЕЧЕНЬ ЗАДАНИЙ К ПРАКТИЧЕСКОЙ АУДИТОРНОЙ РАБОТЕ СТУДЕНТОВ	
	№ 1. Запуск и настройка программы браузера. Установка web – сервера	8
	№ 2. Разработка структуры титульной страницы тематического сайта	10
	№ 3. Разработка системы навигации тематического сайта	11
	№ 4. Подготовка изображения для навигационной карты	13
	№ 5. Разработка навигационной карты для системы навигации	14
	№ 6. Разработка фреймовой структуры web - ресурса	16
	№ 7. Разработка HTML формы	19
	№ 8. Подключение каскадной таблицы стилей к разработанному web – ресурсу.Использование CSS для форматирования свойств текста, шрифта, цвета и фона страницы	20
	№ 9. Блочная верстка	24
	№ 10. Использование различных способов встраивания сценария JavaScript в HTML документ	26
	№ 11. Решение линейных задач с помощью JavaScript	29
	№ 12. Решение задач на ввод/вывод данных	31
	№ 13. Решение задач с использованием операторов ветвления и операторов сравнения	32
	№ 14. Решение задач с использованием операторов условного перехода	34
	№ 15. Решение задач с использованием операторов цикла	36
	№ 16. Разработка файла обработчика HTML - формы	37
	№ 17. Решение задач линейной структуры. Вывод результата в окно браузера	38
	№ 18. Решение задач с использованием оператора ветвления	40
	№ 19. Решение задач с использованием оператора выбора	42
	№ 20. Решение задач на обработку массива. Вывод заданного массива в окно браузера	44
	№ 21. Решение задач на обработку ассоциативного массива. Вывод элементов массива в окно браузера	48
	№ 22. Разработка файла обработчика HTML - формы	52
	№ 23. Проектирование web – баз данных	53

№ 24. Создание однотабличной БД	55
№ 25. Функции получения доступа, функции открытия/закрытия соединения с БД	56
№ 26. Инструмент PHP MyAdmin	57
№ 27. Создание однотабличной БД средствами PHP MyAdmin	59
№ 28. Типы данных, привилегии	61
№ 29. Подключение к БД с помощью PHP	62
№ 30. Выборка данных - оператор SELECT	66
№ 31. Выборка данных (сложный запрос, ограничение на количество выбранных записей, выборка в алфавитном порядке)	67
№ 32. Выборка данных, вывод данных в цикле	68
№ 33. Редактирование, обновление и удаление данных	69
№ 34. Совместное использование web- форм и базы данных	77
№ 35. Разработка формы для ввода /вывода данных в БД	79
№ 36. Ввод данных в БД, вывод информации из БД.	80
№ 37. Проектирование клиентской части	81
№ 38. Проектирование структуры web-приложения	82
№ 39. Разработка клиентской части web - приложения	83
№ 40. Разработка системы навигации	85
№ 41. Обработка контента, ввод и верстка	86
№ 42. Проектирование стилистических решений	87
№ 43. Подключение и создание CSS стилей	88
№ 44. Проектирование серверной части	90
№ 45. Проектирование реляционной БД	91
№ 46. Разработка реляционной базы данных	92
№ 47. Ввод данных в БД	92
№ 48. Обеспечение взаимодействия БД и web - приложения	93
№ 49. Организация вывода данных из БД в окно программы браузера	94
№ 50. Разработка web-форм (формы HTML)	95
№ 51. Ввод данных в БД с помощью передачи простой формы	96

	№ 52. Извлечение содержимого таблиц	97
	№ 53. Определение количества записей в результирующей таблицы	98
	№ 54. Система регистрации	98
	№ 55. Система авторизации	99
	№ 56. Базовая HTTP - авторизация	100
	№ 57. Пользователи on-line	101
	№ 58. Дифференцированный зачет в форме защиты проекта	102
3	ИНФОРМАЦИОННЫЕ ИСТОЧНИКИ	104

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Методические указания по выполнению практических работ по дисциплине «Web-программирование» предназначены для реализации государственных требований к минимуму содержания и уровню подготовки выпускников по специальности 09.02.03 «Программирование в компьютерных системах» среднего профессионального образования.

В результате изучения дисциплины и выполнения практических работ **студент должен уметь:**

- Проектировать структуру web – ресурса;
- Разрабатывать систему навигации по web ресурсу;
- Разрабатывать статичные web страницы используя языки разметки web-страниц;
- Разрабатывать стилевое оформление web ресурса на основе CSS;
- Разрабатывать динамические web-страницы с использованием языков программирования JavaScript, PHP;
- Разрабатывать web – приложения по работе с базами данных;
- Устанавливать и настраивать ПО предназначенное для разработки web ресурсов

знать:

- Состав, структуру, принципы реализации и функционирования технологии «клиент - сервер»
- Web технологии и языки веб – разработки (HTML, XHTML, CSS, JavaScript, PHP, MySQL, инструментарий технологии web-программирования);
- Основы Программирование на стороне Клиента;
- Основы Программирование на стороне сервера;
- Принципы Разработки web – приложений по работе с базами данных;

Сформировать общие компетенции:

- ОК 1. Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес;
- ОК 2. Организовывать собственную деятельность, выбирать типовые методы и способы выполнения профессиональных задач, оценивать их эффективность и качество;
- ОК 3. Принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность;
- ОК 4. Осуществлять поиск и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личного развития;
- ОК 5. Использовать информационно-коммуникационные технологии в профессиональной деятельности;
- ОК 6. Работать в коллективе и команде, эффективно общаться с коллегами, руководством, потребителями;
- ОК 7. Брать на себя ответственность за работу членов команды (подчиненных), за результат выполнения заданий;

- ОК 8. Самостоятельно определять задачи профессионального роста и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации;
- ОК 9. Ориентироваться в условиях частой смены технологий в профессиональной деятельности;

Сформировать профессиональные компетенции:

- ПК 1.1. Выполнять разработку спецификаций отдельных компонент;
- ПК 1.2. Осуществлять разработку кода программного продукта на основе готовых спецификаций на уровне модуля;
- ПК 1.3. Выполнять отладку программных модулей с использованием специализированных программных средств;
- ПК 1.4. Выполнять тестирование программных модулей;
- ПК 1.5. Осуществлять оптимизацию программного кода модуля;
- ПК 2.1. Разрабатывать объекты базы данных;
- ПК 2.2. Реализовывать базу данных в конкретной системе управления базами данных (СУБД);
- ПК 2.3. Решать вопросы администрирования базы данных;
- ПК 2.4. Реализовывать методы и технологии защиты информации в базах данных.

Важным направлением профессиональной подготовки специалиста в области программирования в компьютерных системах является формирование умений решать задачи с использованием компьютера и его программного обеспечения.

Структура учебной работы студентов в рамках данного практикума предусматривает практические занятия в компьютерном классе, в ходе которых под руководством преподавателя осуществляется разработка и реализация на компьютере конкретного программного продукта в соответствии с предложенной тематикой.

Умения в области web-программирования, сформированные в процессе выполнения практических работ, составят базу, на основе которой в дальнейшем могут быть эффективно решены задачи практической подготовки. У студентов в ходе изучения соответствующего учебного материала должны быть сформированы знания и практические умения, достаточные для самостоятельной разработки программного продукта

На практические работы по дисциплине «Web-программирование» отводится 116 часов.

Методические указания включают в себя:

- Темы;
- Задания;
- Формы контроля;
- Критерии оценок;
- Информационные источники.

ПЕРЕЧЕНЬ ЗАДАНИЙ К ПРАКТИЧЕСКОЙ АУДИТОРНОЙ РАБОТЕ СТУДЕНТОВ

Работа № 1.

ТЕМА: Запуск и настройка программы браузера. Установка web - сервера

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере. Ответы на вопросы даются письменно в тетради.

ФОРМА КОНТРОЛЯ: Экспертиза качества выполнения практической работы, Проверка правильности письменных ответов на вопросы.

ЗАДАНИЯ:

Выполнить установку и настройку ПО необходимого для разработки web- ресурса:

Браузер;

Web-сервер;

СМС WordPress.

Ход работы:

1. Запустить программу Microsoft Internet Explorer;
2. Выполнить настройку свойств обозревателя:
 - a. Настройка домашней страницы;
 - b. Установить параметры работы с временными файлами ИНТЕРНЕТА:

параметр	значение параметра
а. проверка наличия обновления сохраняемых страниц	при каждом посещении Веб - узла
б. объем используемого места на диске	111 Мб
в. срок хранения в журнале списка посещенных узлов	10 дней

3. Опишите назначение кнопок диалогового окна (письменно в тетради):



4. Выполнить смену кодировки для отображаемых веб – страниц, какие кодировки подходят для работы с русскоязычным сегментом Интернет – ресурсов?
5. Осуществить настройку параметров, управляющих степенью риска приёма из Интернета информации, потенциально опасной для компьютера:
 - a. Установить для узлов Интернета уровень безопасности – выше среднего;
 - b. Чем отличаются уровни безопасности?

уровень безопасности	особенности (заполнить в тетради)
средний	
выше среднего	
высокий	

- c. Установить для узлов Интернета, которые могут причинить вред вашему компьютеру уровень безопасности – высокий;

6. Выполнить Настройку ограничений просмотра Веб-узлов, установите ограничение на просмотр узлов содержащих информацию о курении, употреблении алкоголя, сценах насилия и т.п.

а. Чем отличаются степени ограничения?

степень	описание (заполнить в тетради)
отсутствует	
ограничено	
не ограничено	

7. Установите параметры автозаполнения аналогично тому, как представлено на рис.1:

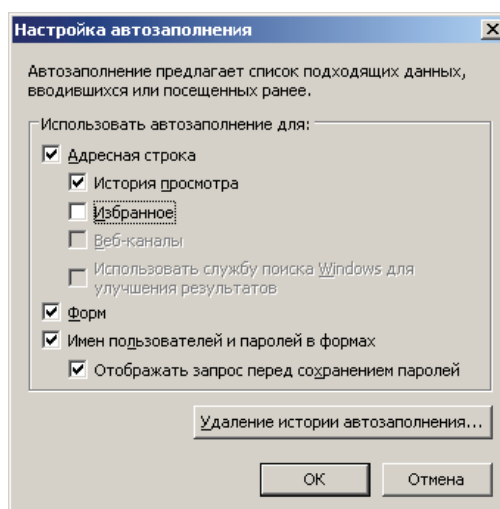


Рисунок 1 Установка параметров автозаполнения

8. Измените или заново настрой параметры соединения с Интернетом, таким образом, как показано на рис.2:

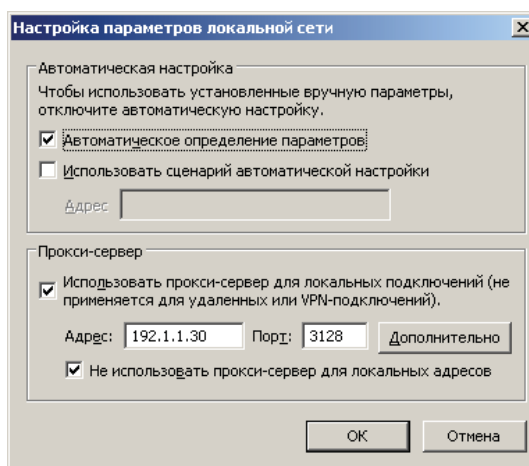


Рисунок 2 Настройка подключения

9. Выполнить установку веб – сервера на персональный компьютер;
10. Выполнить установку СМC WordPress на локальный компьютер.
11. Письменно ответить на контрольные вопросы:
 - а. Проведите обзор программ – браузеров;
 - б. Выполните сравнительную характеристику функциональных возможностей различных программ – браузеров;
 - в. Объясните причины большого числа предложений на рынке программ такого класса;

- d. Чем обусловлен ваш личный выбор программы браузера на домашнем персональном компьютере?

КРИТЕРИЙ ОЦЕНКИ:

- Оценка «5» - ставится, если студент выполнил все практические задания, указанные в ходе работы (1-10), письменно ответил на контрольные вопросы, выписал в тетрадь отличия уровней безопасности и степеней ограничений при работе с программой MicrosoftInternetExplorer;
- Оценка «4» - ставится, если студент выполнил все практические задания, указанные в ходе работы (1-10), однако теоретическую часть работы в тетради выполнить не успел, но ориентируется в вопросах и может ответить на все вопросы устно;
- Оценка «3» - ставится, если студент выполнил работу по настройке программы MicrosoftInternetExplorer, но при установке веб – сервера допустил ошибки. Ориентируется в теоретической части работы и может устно отвечать на вопросы;
- Оценка «2» - ставится, если студент справился с менее чем на 50 % работы (задания 1-4), не может отвечать на теоретические вопросы, задания 5б., 6 а – не выполнены, веб – сервер – не установлен.

Работа № 2.

ТЕМА: Разработка структуры титульной страницы тематического сайта

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка гипертекстовой разметки HTMLи сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

номер задачи	текст задачи
задача 1	А) Используя теги описания структуры HTML-документа создать файл с именем Index.html; Б) В заголовке окна документа должна отражаться следующая информация: "Лабораторная работа №1. студент гр. П-201 (ФИО)" В) Созданный файл сохранить в папку: WEB/LR1
задача 2	Используя для макетирования страницы табличную структуру сформируйте структуру в соответствии со своим вариантом.
задача 3	Заполните созданную web страницу контентом: А) Определите тематическое содержание и в верхней части страницы поместите: наименование web-ресурса, логотип; Б) в зависимости от тематики сформируйте меню, используя теги для формирования списков; В) некоторые пункты меню сформировать с помощью графических файлов с изображением кнопок; Г) в зависимости от тематики наполнить текстовым и графическим контентом;
задача 4	В зависимости от содержания меню (задание 3 Б) создать для каждого пункта меню отдельнуюweb-страницу.

	Структуру, оформление и информационное наполнение разработать по своему усмотрению в соответствии с тематикой web - ресурса.
задача 5	Разработать систему навигации по пунктам меню, используя теги создания гиперссылок

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Не переходите к выполнению следующего задания, пока не убедитесь, что задача решена правильно (результат соответствует условию задачи).

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал).

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - ставится, если студент правильно выполнил практические задания (1-4).

Оценка «4» - ставится, если студент правильно выполнил практические задания (1-3).;

Оценка «3» - ставится, если студент правильно выполнил практические задания (1-2).;

Оценка «2» - ставится, если студент правильно выполнил практические задания (1-2), но имеются грубые ошибки при выполнении или студент предоставив работу на проверку самостоятельно не может прокомментировать значение HTML кода .

Работа № 3.

ТЕМА: Разработка системы навигации тематического сайта

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка гипертекстовой разметки HTMLи сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

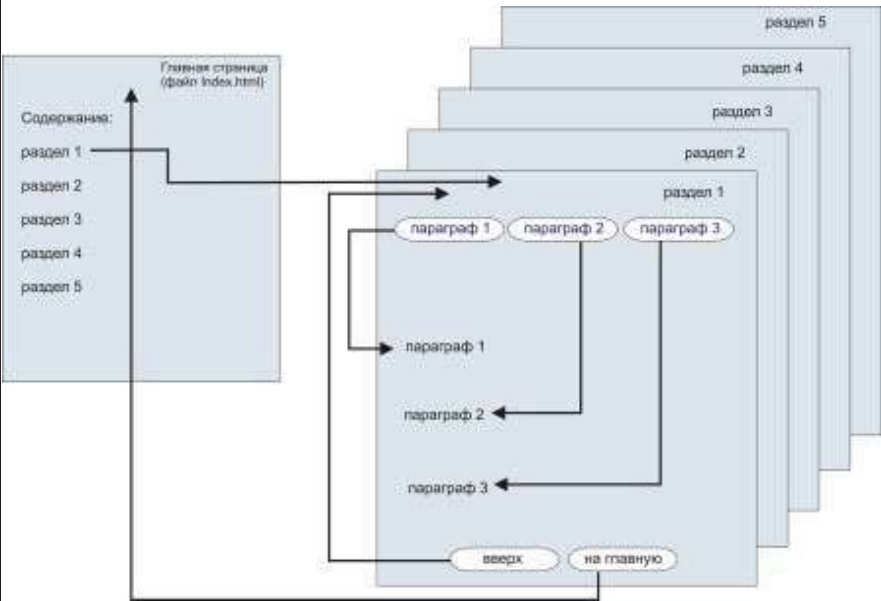
Условия контроля: отсрочено, вначале урока при проведении следующей практической работы.

Работа выполняется в ходе занятия с возможностью доработки самостоятельно (внеаудиторно).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

Номер задачи	текст задачи
задача 1	А) Помимо файла Index.html (результат пр. работы №1) создать 5 файлов отражающих содержание будущего меню; (т.е. заголовок каждого созданного файла в последствии должен стать пунктом меню в файле Index.html).

	Пример структуры (рис.1):  Б) В заголовке каждого созданного окна документа должна отражаться информация о содержимом файла. В) Созданные файлы сохранить в папку: WEB/LR2
задача 2	Создайте в файле Index.html содержание будущего меню в соответствии с названиями файлов из задачи 1-А. Расположите содержание в зависимости от варианта: 1 - левая колонка таблицы, 2 - правая колонка таблицы, 3 - третья строка таблицы, 4 - четвертая строка таблицы, 5 - вторая строка таблицы, 6 - вторая строка таблицы, 7 - левый столбец таблицы, 8 - правый столбец таблицы, 9 - левый столбец таблицы, 10 второй столбец таблицы, 11 - левый столбец таблицы, 12 - правый столбец таблицы, 13 - левый столбец таблицы, 14 - левый столбец таблицы, 15 - средний столбец таблицы, 16 - средний столбец таблицы, 17 - левый столбец таблицы, 18 - правый столбец таблицы, 19 - левый столбец таблицы, 20 - левый столбец таблицы
задача 3	Заполните созданные web страницы контентом: А) Определите тематическое содержание каждой веб-страницы; Б) В верхней части страницы поместите наименование раздела web-ресурса; В) после заголовка в верхней части окна каждого документа сформируйте меню для перемещения по меткам внутри самой страницы (аналогично перемещению по параграфам внутри раздела); Г) внизу окна с помощью графических файлов с изображением кнопок разместить ссылки: вверх и на главную;

	Д) в зависимости от тематики наполнить текстовым и графическим контентом;
задача 4	в нижней части окна каждого файла разместить информацию об авторе разрабатываемого ресурса с указанием адреса электронной почты. адрес электронной почты должен быть оформлен в виде ссылки.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Не переходите к выполнению следующего задания, пока не убедитесь, что задача решена правильно (результат соответствует условию задачи).

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - ставится, если студент правильно выполнил все практические задания (1-4).

Оценка «4» - ставится, если студент правильно выполнил практические задания (1-3).;

Оценка «3» - ставится, если студент правильно выполнил практические задания (1-2).;

Оценка «2» - ставится, если студент правильно выполнил практические задания (1-2), но имеются грубые ошибки при выполнении или студент, предоставив работу на проверку, самостоятельно не может прокомментировать значение HTML кода.

Работа № 4.

ТЕМА: Подготовка изображения для навигационной карты

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка гипертекстовой разметкиHTMLи сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Работа выполняется в ходе занятия с возможностью доработки самостоятельно (внеаудиторно).

ФОРМА КОНТРОЛЯ: отсрочено, вначале урока при проведении следующей практической работы.

ЗАДАНИЯ:

номер задачи	текст задачи
задача 1	А) загрузите графический редактор (CorelDraw, Photoshop или любой другой, который позволяет создавать графические изображения в формате gif, jpg, png. Б) Подготовьте изображение, которое по своему содержанию будет

	визуально отражать логическую структуру вашего web-ресурса.
задача 2	А) В файле Index.html (результат пр. работы №1) разместить графическое изображение, отражающее структуру разрабатываемого web - ресурса 

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Не переходите к выполнению следующего задания, пока не убедитесь, что задача решена правильно (результат соответствует условию задачи).

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - ставится, если студент правильно выполнил все практические задания (1-2).

Оценка «4» - ставится, если студент выполнил практические задания (1-2), однако, подготовленное изображение нуждается в доработке .;

Оценка «3» - выполнение практической работы соответствует оценке «хорошо», однако в выполнении задания №2 имеются ошибки, которые студент может исправить самостоятельно при указании на них преподавателя.;

Оценка «2» - практическая работа выполнена с большим количеством замечаний, которые студент не может исправить самостоятельно при указании на них преподавателя .

Работа № 5.

ТЕМА: Разработка навигационной карты для системы навигации

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка гипертекстовой разметки HTML и сохраняется в личной директории.

Не переходите к выполнению следующего задания, пока не убедитесь, что задача решена правильно (результат соответствует условию задачи).

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - ставится, если студент правильно выполнил все практические задания (2 Б, 2В).

Оценка «4» - ставится, если студент выполнил практические задания (2 Б, 2В), однако, имеются незначительные ошибки при описании координат активных областей;

Оценка «3» - выполнение практической работы соответствует оценке «хорошо», однако в выполнении задания №2Б имеются ошибки, которые студент может исправить самостоятельно при указании на них преподавателя.;

Оценка «2» - практическая работа выполнена с большим количеством замечаний, которые студент не может исправить самостоятельно при указании на них преподавателя .

Работа № 6.

ТЕМА: Разработка фреймовой структуры web - ресурса

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка гипертекстовой разметкиHTMLи сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: отсрочено, вначале урока при проведении следующей практической работы.

Работа выполняется в ходе занятия с возможностью доработки самостоятельно (внеаудиторно).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

Номер задачи	текст задачи		
задача 1	Создать четыре веб-страницы: Head.html, Menu.html, Karta.html, feuilleton.html.		
	Содержание страниц следующее:		
	Имя файла	Содержимое файла	Где взять
	Head.html	заголовок, логотип и кнопочное меню ресурса	файл Index.html (практическая работа 1)
	Menu.html	текстовое меню ресурса	файл Index.html (практическая работа №2)

	<table border="1"><tr><td>Karta.html</td><td>навигационная карта ресурса</td><td>файл Index.html (практическая работа №3)</td></tr><tr><td>Feuilleton.html</td><td>справочная информация о ресурсе: втор, ссылка на почтовый ящик</td><td>файл Index.html (практическая работа №2)</td></tr></table>	Karta.html	навигационная карта ресурса	файл Index.html (практическая работа №3)	Feuilleton.html	справочная информация о ресурсе: втор, ссылка на почтовый ящик	файл Index.html (практическая работа №2)
Karta.html	навигационная карта ресурса	файл Index.html (практическая работа №3)					
Feuilleton.html	справочная информация о ресурсе: втор, ссылка на почтовый ящик	файл Index.html (практическая работа №2)					
задача 2	Изменить структуру файла Index.html (В соответствии с вариантом) таким образом, чтобы все четыре файла из задачи №1 отображались в окне браузера (структура фреймов). вариант 1 Заголовок главная стр. 1 стр. 2 стр. 3 стр. 4 стр. 5 Содержание: стр. 1 стр. 2 стр. 3 стр. 4 стр. 5 навигационная карта автор сайта Иванов И.И., все вопросы по почте: ivan@mail.ru вариант 2 Заголовок главная стр. 1 стр. 2 стр. 3 стр. 4 стр. 5 Содержание: стр. 1 стр. 2 стр. 3 стр. 4 стр. 5 навигационная карта автор сайта Иванов И.И., все вопросы по почте: ivan@mail.ru вариант 3 Заголовок главная стр. 1 стр. 2 стр. 3 стр. 4 стр. 5 Содержание: стр. 1 стр. 2 стр. 3 стр. 4 стр. 5 навигационная карта автор сайта Иванов И.И., все вопросы по почте: ivan@mail.ru вариант 4						

	Заголовок главная стр. 1 стр. 2 стр. 3 стр. 4 стр. 5 навигационная карта автор сайта Иванов И.И., все вопросы по почте: Ivan@mail.ru Содержание: стр. 1 стр. 2 стр. 3 стр. 4 стр. 5 вариант 5 Заголовок главная стр. 1 стр. 2 стр. 3 стр. 4 стр. 5 навигационная карта автор сайта Иванов И.И., все вопросы по почте: Ivan@mail.ru Содержание: стр. 1 стр. 2 стр. 3 стр. 4 стр. 5
Задача 3	Организовать систему навигации ресурса таким образом, чтобы при выборе любой гиперссылки (текстовой, кнопочной) информация загружалась в окно файла Index.html в то место, где располагается навигационная карта.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Не переходите к выполнению следующего задания, пока не убедитесь, что задача решена правильно (результат соответствует условию задачи).

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - ставится, если студент правильно выполнил все практические задания (1-3).

Оценка «4» - ставится, если студент выполнил практические задания (1-3), однако, имеются незначительные ошибки которые студент исправил самостоятельно;

Оценка «3» - ставится, если студент выполнил практические задания (1-3), однако, имеются ошибки которые студент может исправить при указании на них преподавателя с использованием методического материала (МУ к практическим, справочник, учебник);

Оценка «2» - практическая работа выполнена с большим количеством замечаний, которые студент не может исправить самостоятельно при указании на них преподавателя .

Работа № 7.

ТЕМА: Разработка HTML формы

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка гипертекстовой разметки HTML и сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно, 10-15 минут в конце занятия.

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

Задание 1. Разработать HTML форму для ввода следующей информации:

- ФИО студента (3 текстовых поля);
- специальность (поле списка);
- группа (поле списка);
- пол (переключатель);
- курс (переключатели);
- дисциплины по выбору: 5-6 дисциплин (флажки);
- комментарии «о себе» (текстовый блок);
- поле для прикрепления файла(фотографии)
- кнопка подтверждения.
- кнопка сброса.

Задание 2. Разместить все элементы формы таким образом, чтобы форма выглядела законченной, все элементы должны быть выровнены, элементы одного типа, должны иметь одинаковые параметры.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Не переходите к выполнению следующего задания, пока не убедитесь, что задача решена правильно (результат соответствует условию задачи).

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник

информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

- Оценка «5» - Практическая работа (задание 1,2) выполнены правильно. Студент ориентируется в HTML коде, может самостоятельно изменить расположение и параметры любого элемента HTML формы.
- Оценка «4» - Практическая работа (задание 1,2) выполнены правильно. Студент ориентируется в HTML коде, может изменить расположение и параметры любого элемента HTML формы с использованием методических указаний, справочников, учебников.;
- Оценка «3» - Студент правильно выполнил задание 1. при выполнении задания 2 имеются недоработки, студент затрудняется с изменением типа и параметров некоторых элементов формы, однако при указании на ошибки может объяснить в чем ошибка, и что необходимо исправить.
- Оценка «2» - В практической работе имеются грубые ошибки (недостаточное количество элементов формы, студент не может исправить тип элемента, его расположение и внешний вид без подсказки преподавателя).

Работа № 8.

ТЕМА: Подключение каскадной таблицы стилей к разработанному web - ресурсу. Использование CSS для форматирования свойств текста, шрифта, цвета и фона страницы

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка гипертекстовой разметки HTML и CSS, сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (в начале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

задача 1

1. Создать файл index.html содержащий следующие два абзаца текста:

CascadingStyleSheet

CSS (CascadingStyleSheet) или Каскадные таблицы стилей – это технология, позволяющая определять стили для документов HTML. В связи с тем, что HTML – это содержимое или контент, каскадные стили – это его визуальное представление.

Стили соотносятся только с HTML. Формат стиля имеет вид "свойство: значение", и большинство свойств могут применяться к большинству тегов HTML.

Способы подключения стилей при работе с HTML:

- внедрение стиля на уровне элемента;
- описание стиля на уровне документа;
- подключение внешних стилей.

HTML	CSS
Язык разметки гипертекста. предназначен для описания структуры расположения контента в web-странице.	Язык описания стилей. Позволяет описать внешний вид объектов контента. основное назначение - автоматизация стилевого оформления множества страниц по единым стилевым правилам.

2. Внедрить на уровне элемента стили описания для фрагментов текста:

фрагмент	стиль
CSS	Шрифт - Impact, Размер - 12pt, Цвет - красный
Каскадные таблицы стилей	Шрифт - ARIAL, Размер - 14 pt, Цвет - зеленый Начертание - курсив.

задача 2

на уровне документа определить стили для следующих элементов:

элемент	стиль
Абзац	Шрифт - Times New Roman Размер - 10 pt Цвет - темно синий
Ссылка	Цвет - фиолетовый
Заголовок 1-го уровня	Шрифт - Impact Размер - 14 pt Цвет - бордовый Выравнивание - по центру

задача 3

Создать внешнюю таблицу стилей style.css. Подключить созданную таблицу к файлу index.html. В таблице стилей задать параметры стиля для следующих элементов:

элемент	стиль
Фон страницы	Темно серый
Текст абзаца	Светло серый
Заголовок любого уровня	Текст - белый Фон - черный Рамка 3px, цвет рамки белый
Ссылка	Цвет ссылки - желтый Цвет просмотренной ссылки - голубой
Список	Шрифт - Impact Размер - 14 pt Цвет - бордовый Выравнивание - по центру
Таблица	Фон - черный

задача 4

После сдачи выполненных задач №1-3 преподавателю, измените содержание файла style.css таким образом, чтобы оформление страницы Index.html было предназначено для следующих целевых аудиторий:

1. детская страничка;
 2. страница для чтения офисным работником;
 3. страничка для дизайнерского агентства.
 4. страница сайта строительной фирмы;
 5. страница школьного сайта.
1. Разместить на странице следующий контент:

Изменение свойств текста

Ниже описываются свойства, при помощи которых можно изменять размер и форму текста веб-страницы.

font-family

Этот шрифт, такой как Times New Roman, Arial или Verdana.

Определяемые шрифты должны быть установлены на компьютере пользователя, поэтому совершенно не разумно использовать редкие и мало известные шрифты. Существуют несколько отборных "безопасных" шрифтов (наиболее часто используются шрифты arial, verdana, timesnewroman), но вы можете определять более одного шрифта, разделяя их запятыми. Это делается для того, чтобы в том случае, когда у пользователя не установлен первый шрифт из определенных в свойстве, то браузер будет искать следующий по списку шрифт, затем следующий, и так до тех пор, пока не будет найден установленный у пользователя шрифт, либо будет достигнут конец списка. Такая возможность очень полезна, так как иногда на разных компьютерах может устанавливаться разный набор шрифтов. Таким образом, например, декларация **font-family: arial, helvetica** может использоваться для определения соответствующего шрифта на PC (на котором традиционно устанавливается шрифт arial, но нет шрифта helvetica) и на AppleMac (на котором традиционно нет шрифта arial, и в этом случае будет использован шрифт helvetica).

Внимание: Если имя шрифта состоит из нескольких слов, то его следует заключать в кавычки, например, **font-family: "TimesNewRoman"**.

font-size

Размер шрифта. Будьте с этим свойством осторожны: не стоит использовать в качестве заголовков параграфы, отображаемые шрифтом большого размера; для этого есть специальные теги (h1, h2 и т.д.), которые имеют массу достоинств, даже если в действительности шрифт параграфа будет больше шрифта заголовка (что не должны делать разумные люди).

font-weight

Это свойство позволяет установить жирность шрифта. На практике обычно используют два его состояния: **font-weight: bold** и **font-weight: normal**. Хотя в теории оно также может принимать значения **bolder**, **lighter**, **100**, **200**, **300**, **400**, **500**, **600**, **700**, **800** и **1000**. Однако, посмотрев на то, как большинство браузеров отказываются понимать эти значения, вы поймете, что гораздо безопаснее иметь дело всего лишь со значениями **bold** и **normal**.

font-style

Данное свойство определяет, будет ли текст отображаться курсивом или нет. Оно может быть либо **font-style: italic**, либо **font-style: normal**.

text-decoration

Это свойство определяет подчеркнутость текста. Оно может быть:

- **text-decoration: overline** – помещает горизонтальную линию над текстом.
- **text-decoration: line-through** – помещает горизонтальную линию через текст, создавая эффект перечеркивания.
- **text-decoration: underline** - помещает горизонтальную линию под текст, создавая эффект подчеркивания. Не стоит использовать, так как обычно пользователи предполагают, что подчеркнутый текст – ссылка.

Как правило, данное свойство используется для оформления ссылок, в частности чтобы убрать подчеркивание при помощи значения **text-decoration: none**.

text-transform

Это свойство позволяет изменить регистр букв текста.

- **text-transform: capitalize** – переводит первые буквы всех слов в верхний регистр.
- **text-transform: uppercase** – переводит все буквы в верхний регистр.
- **text-transform: lowercase** - переводит все буквы в нижний регистр.
- **text-transform: none** – попробуйте сами это значение.

Разбивка текста

Свойства letter-spacing и word-spacing определяют расстояние между буквами и словами соответственно. Их значением может быть число в любых единицах измерения, либо ключевое слово **normal**.

Свойство line-height определяет высоту строки элемента, не влияя на размер шрифта. Значением может быть число (которое будет множителем размера шрифта. Например, "2" означает, что высота строки в два раза больше размера шрифта.), либо проценты, либо слово **normal**.

Свойство text-align выравнивает текст внутри элемента. Принимает значения **left** (выравнивание по левому краю), **right** (выравнивание по правому краю), **center** (выравнивание по центру) или **justify** (равномерное выравнивание по ширине).

Свойство text-indent создает отступ на заданную величину первой строки параграфа. Подобное форматирование традиционно для полиграфии и редко в цифровых средствах информации.

задача 2

2. Подключите к созданному файлу каскадную таблицу стилей style_2.css. На примере текста из задачи №1 продемонстрируйте описание стилевых решений для:

- font-family
- font-size
- font-weight
- font-style
- text-decoration
- text-transform
- letter-spacing

задача 3

3. в файле style_2.css опишите стилевое решение для определения бирюзового фона страницы.

задача 4

4. после выполнения задач 1,2,3 и сдачи работы преподавателю, измените стилевое решение для фона страницы таким образом, чтобы в качестве фона использовался файл - изображение. продемонстрируйте преподавателю различные варианты использования

фоновых картинок, чтобы оформление страницы Index_9.html было предназначено для следующих целевых аудиторий:

- детская страничка;
- страница для чтения офисным работником;
- страничка для дизайнерского агентства.
- страница сайта строительной фирмы;
- страница школьного сайта.

5. Попробуйте использовать разные варианты размещения фоновых изображений (мозаичное размещение фонового изображения, размещение изображения вверху/внизу/слева/справа в пределах страницы);

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Не переходите к выполнению следующего задания, пока не убедитесь, что задача решена правильно (результат соответствует условию задачи).

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задание 1,2, 3, 4) выполнены правильно. Студент ориентируется в CSS коде, может самостоятельно изменить любые стилевые решения, добавлять новые. Задание №4 может проверяться отсрочено, вначале урока следующего занятия.

Оценка «4» - Практическая работа (задание 1, 2, 3) выполнены правильно. Студент ориентируется в CSS коде, может самостоятельно изменить любые стилевые решения, добавлять новые. Задание №4 не выполнено.

Оценка «3» - Практическая работа (задание 1, 2, 3) выполнены правильно. Студент ориентируется в CSS коде, однако для изменения имеющихся стилевых решений нуждается в подсказке (справочная литература, лекционный материал)

Оценка «2» - В практической работе выполнено менее 3 задач. Студент затрудняется в демонстрации возможности изменить стилевые решения при указании преподавателя.

Работа № 9.

ТЕМА: Блочная верстка

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка гипертекстовой разметки HTML и CSS, сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (вначале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

задача 1

Создайте файл Index_10.html, содержащий следующую информацию:

способы использования CSS

- [на уровне элемента](#)
- [на уровне документа](#)
- [на уровне внешнего файла](#)

гиперссылки на три страницы, описывающие способы использования стилевых описаний для

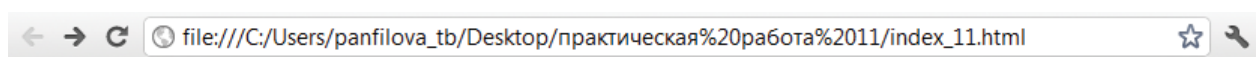
Cascading Styles Sheet

CSS (Cascading Styles Sheet) или Каскадные таблицы стилей – это технология, позволяющая определять стили для документов HTML. В связи с тем, что HTML – это содержимое или контент, каскадные стили – это его визуальное представление. Стили соотносятся только с HTML. Формат стиля имеет вид "свойство: значение", и большинство свойств могут применяться к большинству тегов HTML.

(внимательно прочитайте лекционный материал! При необходимости обратитесь к интернет-источникам указанным в МУ к выполнению практических работ)

задача 2

Создайте текстовый файл style_10.css в котором опишите позиционированное размещение контента таким образом, чтобы гиперссылки находились в левой части окна браузера, основной текстовый контент справа от гиперссылок:



способы использования CSS

- [на уровне элемента](#)
- [на уровне документа](#)
- [на уровне внешнего файла](#)

Cascading Styles Sheet

CSS (Cascading Styles Sheet) или Каскадные таблицы стилей – это технология, позволяющая определять стили для документов HTML. В связи с тем, что HTML – это содержимое или контент, каскадные стили – это его визуальное представление. Стили соотносятся только с HTML. Формат стиля имеет вид "свойство: значение", и большинство свойств могут применяться к большинству тегов HTML.

(внимательно прочитайте лекционный материал! При необходимости обратитесь к интернет-источникам указанным в МУ к выполнению практических работ)

задача 3

Подключите созданную каскадную таблицу стилей к файлу Index_10.html, убедитесь, что размещение контента выполнено правильно. Выполните подключение всех гиперссылок. Контент содержащийся в файлах связанных по гиперссылкам, должен при перемещении открываться в том же самом месте что и основной текст (справа от гипертекстового меню).

задача 4

измените позиционирование размещения контента таким образом, чтобы в главной странице Index_10.html справа размещалось меню шириной 10em. меню должно содержать ссылки на автора, e-meil (при нажатии на ссылку **автор** - должна открываться информация об авторе практической работы, при нажатии на ссылку **e-meil** - должен запускаться почтовый клиент для отправки письма автору работы).

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Не переходите к выполнению следующего задания, пока не убедитесь, что задача решена правильно (результат соответствует условию задачи).

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задание 1,2, 3, 4) выполнены правильно. Студент ориентируется в CSS коде, может самостоятельно изменить любые стилевые решения, добавлять новые. Задача №4 может проверяться отсрочено, вначале урока следующего занятия.

Оценка «4» - Практическая работа (задание 1, 2, 3) выполнены правильно. Студент ориентируется в CSS коде, может самостоятельно изменить любые стилевые решения, добавлять новые. Задача №4 не выполнена.

Оценка «3» - Практическая работа (задание 1, 2, 3) выполнены правильно. Студент ориентируется в CSS коде, однако для изменения имеющихся стилевых решений нуждается в подсказке (справочная литература, лекционный материал).

Оценка «2» - В практической работе выполнено менее 3 задач. Студент затрудняется в демонстрации возможности изменить стилевые решения при указании преподавателя.

Работа № 10.

ТЕМА: Использование различных способов встраивания сценария JavaScript в HTML документ

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка гипертекстовой разметки HTML и CSS, сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (вначале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

Задача 1

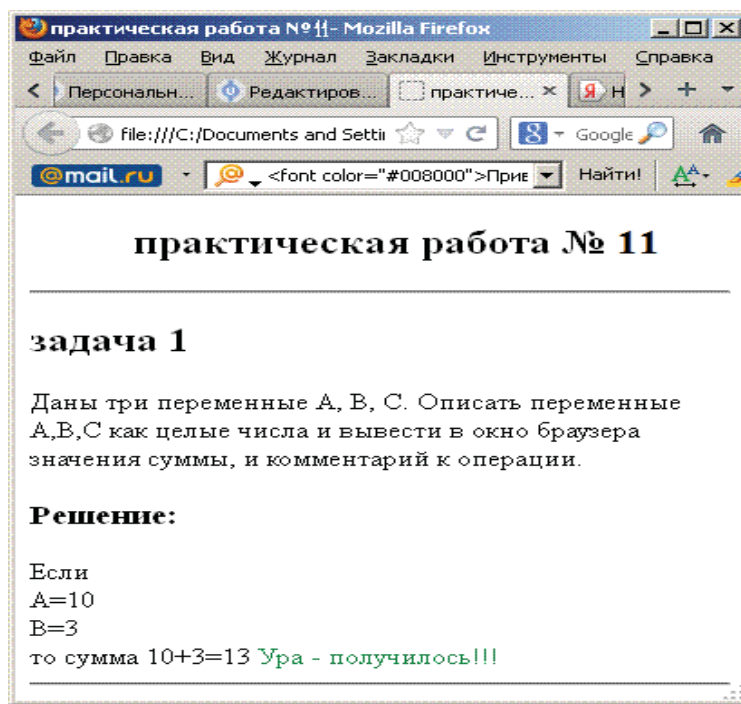
Разместить в теле HTML – документа (между тегами body), скрипт который:

Описывает переменные A, B и выводит на экран значение C, которое вычисляется в соответствии с вариантом.

Вывести в окно браузера значение исходных данных и результат, а также комментарий.

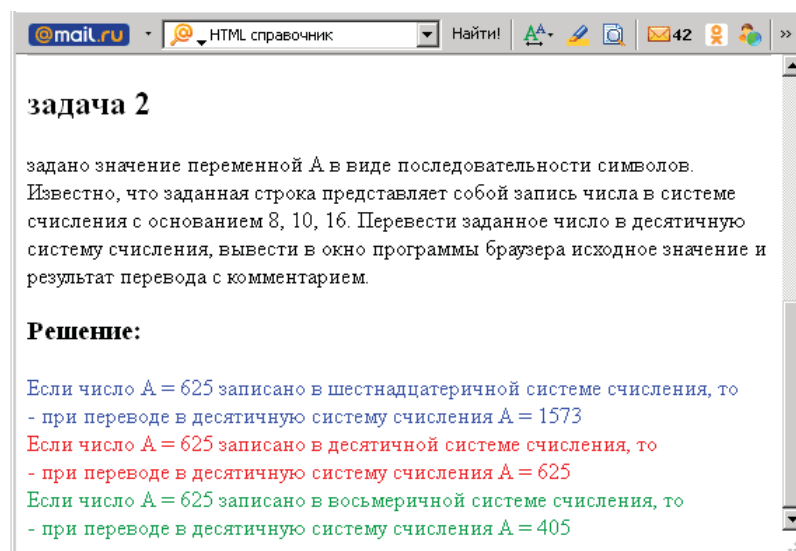
вариант	тип A	тип B	операция	комментарий!
1	целое	вещественное	умножение	Ура получилось!
2	вещественное	вещественное	вычитание	Где наша не пропадала!
3	строковое	целое	сложение	Что-то тут не то!
4	логическое	логическое	дизъюнкция	Это просто ЛОГИКА!
5	целое	целое	деление по модулю	Не боги горшки обжигали!

Например:



Задача 2 (Включение файлов со сценариями JavaScript:)

Задано значение переменной A в виде последовательности символов. Известно, что заданная строка представляет собой запись числа в системе счисления с основанием 8, 10 или 16. Перевести заданное число в десятичную систему счисления, вывести в окно программы браузера исходное значение и результат перевода с комментарием для каждого из оснований. Например:



Задача 3 (Определение обработчика события.)

1. Разместить в web странице текст "Clickme!", который в ответ на щелчок мыши должен становиться красным;
2. Разместить в web странице текст, состоящий из маркированного списка цветов радуги (красный, оранжевый, желтый, зеленый, голубой, синий, фиолетовый). При нажатии левой кнопкой мыши по названию цвета, фон просматриваемой страницы должен окрашиваться в выбранный щелчком мыши цвет.

Задача 4

Разместить в web странице собственную фотографию, изображающую ваш образ в деловом стиле. Написать JavaScript- скрипт таким образом, чтобы при наведении курсора мыши на фотографию, менялось изображение фотографии на шуточное изображение в карикатурном стиле.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Не переходите к выполнению следующего задания, пока не убедитесь, что задача решена правильно (результат соответствует условию задачи).

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задание 1, 2, 3.1, 3.2, 4) выполнены правильно. Студент ориентируется в CSS коде, может самостоятельно изменить любые стилевые решения, добавлять новые. Задача №4 может проверяться отсрочено, вначале урока следующего занятия.

Оценка «4» - Практическая работа (задание 1, 2, 3.1, 3.2) выполнены правильно. Студент ориентируется в CSS коде, может самостоятельно изменить любые стилевые решения, добавлять новые. Задача №4 не выполнена.

Оценка «3» - Практическая работа (задание 1, 2, 3.1, 3.2) выполнены правильно. Студент ориентируется в CSS коде, однако для изменения имеющихся стилевых решений нуждается в подсказке (справочная литература, лекционный материал).

Оценка «2» - В практической работе выполнено менее 3 задач. Студент затрудняется в демонстрации возможности изменить стилевые решения при указании преподавателя.

Работа № 11.

ТЕМА: Решение линейных задач с помощью JavaScript

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка гипертекстовой разметки HTML и CSS, сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (в начале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

Задача 1

ввести значение 2-х переменных. Поменять местами значения переменных не используя ветвление. Результат работы программы вывести на экран в виде диалогового окна.

Задача 2

Ввести значение 3 - х переменных. продемонстрировать результат конкатенации. Результат работы программы вывести на экран в виде диалогового окна.

Задача 3

Ввести значение 4 - х переменных. продемонстрировать результат математических операций (сложение, вычитание, умножение, деление). Результат работы программы вывести на экран в виде диалогового окна.

Задача 4

Найти значение выражения в соответствии со своим вариантом:

$$1. \frac{b + \sqrt{b^2 + 4ac}}{2a} - a^3c + b^{-2};$$

$$2. \frac{a}{c} \frac{b}{d} - \frac{ab - c}{cd};$$

$$3. \frac{\sin x + \cos y}{\cos x - \sin y} \operatorname{tg} xy;$$

$$4. \frac{x + y}{y + 1} - \frac{xy - 12}{34 + x};$$

$$5. \frac{3 + e^{y-1}}{1 + x^2 |y - \operatorname{tg} x|};$$

$$6. x - \frac{x^3}{3} + \frac{x^5}{5};$$

$$7. \ln \left| \left(y - \sqrt{|x|} \right) \left(x - \frac{y}{x + \frac{x^2}{4}} \right) \right|;$$

$$8. (1 - \operatorname{tg} x)^{\operatorname{ctg} x} + \cos(x - y)$$

$$9. \frac{\ln|\cos x|}{\ln(1 + x^2)};$$

$$10. \left(\frac{x+1}{x-1} \right)^x + 18xy^2;$$

$$11. \left(1 + \frac{1}{x^2} \right)^x - 12x^2y;$$

$$12. \frac{x^2 - 7x + 10}{x^2 - 8x + 12};$$

$$13. \frac{\cos x}{\pi - 2x} + 16x \cos(xy) - 2;$$

$$14. 2^{-x} - \cos x + \sin(2xy);$$

$$15. 2\operatorname{ctg}(3x) - \frac{1}{12x^2 + 7x - 5};$$

$$16. |x^2 - x^3| - \frac{7x}{x^3 - 15x};$$

$$17. x \ln x + \frac{y}{\cos x - \frac{x}{3}};$$

$$18. \sin \sqrt{x+1} - \sin \sqrt{x-1};$$

$$19. e^x - \frac{y^2 + 12xy - 3x^2}{18y - 1};$$

$$20. \frac{1 + \sin \sqrt{x+1}}{\cos(12y - 4)};$$

$$21. 2\operatorname{ctg}(3x) - \frac{\ln \cos x}{\ln(1 + x^2)};$$

$$22. e^x - x - 2 + (1 + x)^x;$$

$$23. 3^x - 4x + (y - \sqrt{|x|});$$

$$24. x - 10 \sin x + |x^4 - x^5|;$$

$$25. x - 10^{\sin x} + \cos(x - y);$$

$$26. \frac{1 + \sin^2(x + y)}{2 + \left| x - \frac{2x}{1 + x^2 y^2} \right|} + x;$$

$$27. \cos^2 \left(\sin \frac{1}{z} \right);$$

$$28. \frac{\cos^2 x}{\sin x} - xyz + \frac{ax^2 + bx + c}{dx^3 - f}.$$

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Не переходите к выполнению следующего задания, пока не убедитесь, что задача решена правильно (результат соответствует условию задачи).

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задание 1, 2, 3, 4) выполнены правильно. Студент ориентируется в коде скриптов, может самостоятельно изменить любые решения, добавлять новые. Задача №4 может проверяться отсрочено, вначале урока следующего занятия.

Оценка «4» - Практическая работа (задание 1, 2, 3) выполнены правильно. Студент ориентируется в коде скриптов, может самостоятельно изменить любые стиливые решения, добавлять новые. Задача №4 не выполнена.

Оценка «3» - Практическая работа (задание 1, 2, 3) выполнены правильно. Студент ориентируется в коде скриптов, однако для изменения скрипта по просьбе преподавателя нуждается в подсказке (справочная литература, лекционный материал).

Оценка «2» - В практической работе выполнено менее 3 задач. Студент затрудняется в демонстрации возможности изменить код скрипта при указании преподавателя.

Работа № 12.

ТЕМА Решение задач на ввод/вывод данных

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка JavaScript, сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (вначале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

Задание: написать скрипт на языке JavaScript

Задача 1

Для ввода-вывода данных в. Проверить три стандартных метода для ввода и вывода данных: alert, prompt и confirm.

Задача 2

Задать значение переменной А, В. Вывести на экран текстовую строку, описывающую вычисление гипотенузы прямоугольного треугольника с катетами А, В. (решение +результат).

Задача 3

Задать значение переменной А, В, С. Вывести на экран текстовую строку, описывающую вычисление площади треугольника по формуле Герона. (решение +результат)

Задача 4

Задать значение переменной А. Вывести на экран текстовую строку, в зависимости от условия: значение переменной может попадать в три разных диапазона. Условие сформулировать самостоятельно. Решение задачи продемонстрировать для значения переменной А из каждого диапазона.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Не переходите к выполнению следующего задания, пока не убедитесь, что задача решена правильно (результат соответствует условию задачи).

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задание 1, 2, 3, 4) выполнены правильно. Студент ориентируется в коде скриптов, может самостоятельно изменить любые решения, добавлять новые. Задача №4 может проверяться отсрочено, вначале урока следующего занятия.

Оценка «4» - Практическая работа (задание 1, 2, 3) выполнены правильно. Студент ориентируется в коде скриптов, может самостоятельно изменить любые стиливые решения, добавлять новые. Задача №4 не выполнена.

Оценка «3» - Практическая работа (задание 1, 2, 3) выполнены правильно. Студент ориентируется в коде скриптов, однако для изменения скрипта по просьбе преподавателя нуждается в подсказке (справочная литература, лекционный материал).

Оценка «2» - В практической работе выполнено менее 3 задач. Студент затрудняется в демонстрации возможности изменить код скрипта при указании преподавателя.

Работа № 13.

ТЕМА: Решение задач с использованием операторов ветвления и операторов сравнения

Работа выполняется на компьютере с использованием языка JavaScript, сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (вначале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

Задание: написать скрипт на языке JavaScript

Задача 1

Написать скрипт, содержащий условный оператор для вычисления нового значения Y по формуле (рис.2)

Примечание: Значения A , B , C задаются в начале выполнения скрипта, с помощью присваивания в соответствии с вариантом:

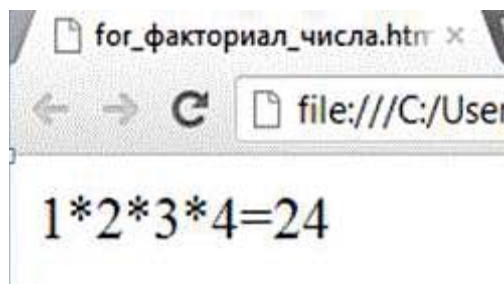
Вариант	A	B	C
Вариант 1	8	6	0
Вариант 2	9	7	0
Вариант 3	10	12	0
Вариант 4	11	45	0
Вариант 5	7	4	0

Вариант	Формула
Вариант 1	$Y = \begin{cases} A, & \text{если } (A > B) \& (B \geq 3); \\ B, & \text{если } (A > B) \& (B < 3); \\ C, & \text{в остальных случаях} \end{cases}$
Вариант 2	$Y = \begin{cases} \sqrt{A-B}, & \text{если } (A > B) \& (B \geq 3); \\ (3-B)^2, & \text{если } (A > B) \& (B < 3); \\ C, & \text{в остальных случаях} \end{cases}$
Вариант 3	$Y = \begin{cases} \sqrt{(A-B)+2}, & \text{если } (A > B) \& (B \geq 3); \\ A^2 \cdot (3-B)^{\frac{1}{2}}, & \text{если } (A > B) \& (B < 3); \\ C, & \text{в остальных случаях} \end{cases}$
Вариант 4	$Y = \begin{cases} \frac{\sqrt{(A-B)}}{(A+B)}, & \text{если } (A > B) \& (B \geq 3); \\ (3-B)^{\frac{1}{4}}, & \text{если } (A > B) \& (B < 3); \\ C, & \text{в остальных случаях} \end{cases}$
Вариант 5	$Y = \begin{cases} \frac{\sqrt{(A \cdot B)}}{(A-B)}, & \text{если } (A > B) \& (B \geq 3); \\ (10-B)^2, & \text{если } (A > B) \& (B < 3); \\ C, & \text{в остальных случаях} \end{cases}$

рис. 1

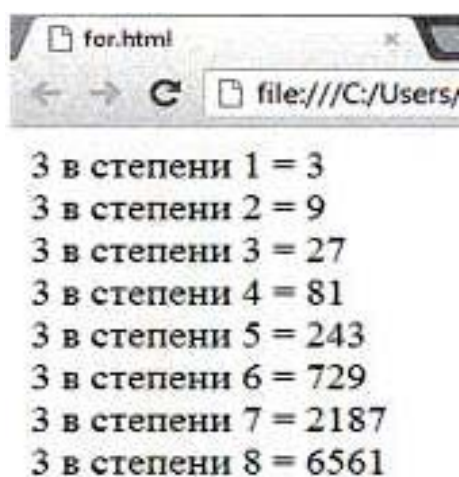
Задача 2

Вывести на экран факториал числа $n!$
(вывести как само значение факториала, так и запись его решения).



Задача 3

Необходимо вывести на экран все значение X в степени $1, 2, \dots, Y$:



Задача 4

Модифицировать код решения задач 1-3, таким образом, чтобы исходные данные задавались путем ввода с клавиатуры.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Не переходите к выполнению следующего задания, пока не убедитесь, что задача решена правильно (результат соответствует условию задачи).

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задание 1, 2, 3, 4) выполнены правильно. Студент ориентируется в коде скриптов, может самостоятельно изменить любые решения, добавлять новые. Задача №4 может проверяться отсрочено, вначале урока следующего занятия.

Оценка «4» - Практическая работа (задание 1, 2, 3) выполнены правильно. Студент ориентируется в коде скриптов, может самостоятельно изменить любые стилевые решения, добавлять новые. Задача №4 не выполнена.

Оценка «3» - Практическая работа (задание 1, 2, 3) выполнены правильно. Студент ориентируется в коде скриптов, однако для изменения скрипта по просьбе преподавателя нуждается в подсказке (справочная литература, лекционный материал).

Оценка «2» - В практической работе выполнено менее 3 задач. Студент затрудняется в демонстрации возможности изменить код скрипта при указании преподавателя.

Работа № 14.

ТЕМА: Решение задач с использованием операторов условного перехода

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка JavaScript, сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (вначале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

Написать скрипт на языке JavaScript

Задача 1

Создать файл index.HTML.

1. описать скрипт который в окно программы браузера выводит таблицу описывающую решение квадратного уравнения. В исходных данных коэффициенты A,B,C, в результате значение корней. Таблица в окне браузера должна содержать три строки исходных данных:

для варианта, когда уравнение не имеет действительных решений, имеет единственное решение, имеет 2 решения.

2. описать условную инструкцию с помощью которой вывести на экран Браузера строку «допуск к сессии получен» если значение переменной \$hvest = 0 или строки «двоечникам среди программистов не место», если значение переменной \$hvest > 0. Предусмотреть множество вариантов вывода комментариев в зависимости от значения переменной \$hvest используя инструкцию SWITCH
3. описать условную инструкцию с помощью, которой вывести на экран Браузера строку «допуск к сессии получен» если значение переменной \$hvest = 0 или строки «двоечникам среди программистов не место», если значение переменной \$hvest > 0. Предусмотреть передачу значения переменной \$hvest через адресную строку с помощью метода GET.
4. описать условную инструкцию с помощью, которой вывести на экран Браузера строку «допуск к сессии получен» если значение переменной \$hvest = 0 или строки «двоечникам среди программистов не место», если значение переменной \$hvest > 0. Предусмотреть передачу значения переменной \$hvest с помощью поля ввода и вывод результата по кнопке с надписью «результат сессии».
5. описать программу калькулятор для вычисления значений при проведении арифметических операций. Программа должна позволить пользователю ввести два числа с клавиатуры и указать операцию, выполняемую над ними.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Не переходите к выполнению следующего задания, пока не убедитесь, что задача решена правильно (результат соответствует условию задачи).

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задание 1, 2, 3, 4, 5) выполнены правильно. Студент ориентируется в коде скриптов, может самостоятельно изменить любые решения, добавлять новые. Задача №5 может проверяться отсрочено, вначале урока следующего занятия.

Оценка «4» - Практическая работа (задание 1, 2, 3, 4) выполнены правильно. Студент ориентируется в коде скриптов, может самостоятельно изменить любые стилевые решения, добавлять новые. Задача №4 не выполнена.

Оценка «3» - Практическая работа (задание 1, 2, 3) выполнены правильно. Студент ориентируется в коде скриптов, однако для изменения скрипта по просьбе преподавателя нуждается в подсказке (справочная литература, лекционный материал).

Оценка «2» - В практической работе выполнено менее 3 задач. Студент затрудняется в демонстрации возможности изменить код скрипта при указании преподавателя.

Работа № 15.

ТЕМА: Решение задач с использованием операторов цикла

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка JavaScript, сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (вначале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

Написать скрипт на языке JavaScript

Задача 1

Дано уравнение вида $a \cdot x^2 + b \cdot x + c = 0$.

Пользователь с клавиатуры вводит значение коэффициентов. Скрипт определяет, есть ли решение квадратного уравнения при введенных значениях коэффициентов. По окончании выдает решение и запрашивает у пользователя подтверждение завершения работы. Если подтверждение есть (ОК) работа завершена, если подтверждения нет (ОТМЕНА) работа возобновляется с запроса о вводе коэффициентов.

Задача 2

Продемонстрировать решение задачи №1 с помощью разных операторов цикла.

Задача 3

Решить задачу №1 используя метод половинного деления. Для нахождения корней уравнения на отрезке от А до В. Значения границ отрезка вводятся с клавиатуры.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Не переходите к выполнению следующего задания, пока не убедитесь, что задача решена правильно (результат соответствует условию задачи).

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задание 1, 2, 3) выполнены правильно. Студент ориентируется в коде скриптов, может самостоятельно изменить любые решения, добавлять новые. Задача №3 может проверяться отсрочено, вначале урока следующего занятия.

Оценка «4» - Практическая работа (задание 1, 2) выполнены правильно. Студент ориентируется в коде скриптов, может самостоятельно изменить любой фрагмент кода на языке JavaScript при указании преподавателя. Задача №3 не выполнена.

Оценка «3» - Практическая работа (задание 1, 2) выполнены правильно. Студент ориентируется в коде скриптов, однако для изменения скрипта по просьбе преподавателя нуждается в подсказке (справочная литература, лекционный материал).

Оценка «2» - В практической работе выполнено менее 2 задач. Студент затрудняется в демонстрации возможности изменить код скрипта при указании преподавателя.

Работа № 16.

ТЕМА: Разработка файла обработчика HTML - формы

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка JavaScript, сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (в начале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

Написать скрипт на языке JavaScript

Задача 1

Создать файл Test.html, содержащий тестирование из 10 вопросов по дисциплине Web - программирование. При оформлении теста использовать элементы форм (поле ввода, переключатели, флажки, поля со списком).

Задача 2

Создать скрипт на языке JavaScript, который обрабатывает результаты выполнения теста и в окно браузера выдает результат: количество правильных ответов, на какие вопросы дан не правильный ответ и выставляет оценку

«отлично» - если количество правильных ответов более или равно 9;

«хорошо» – если количество верных ответов в пределах 7-8;

«удовлетворительно» – если количество верных ответов в пределах 5-6;

«неудовлетворительно» – если количество правильных ответов менее 5.

Задача 3

При обработке результатов теста предусмотреть не только вывод количества правильных и перечня неправильных ответов, а и вывод неправильных ответов с отметками какой вариант правильный.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Не переходите к выполнению следующего задания, пока не убедитесь, что задача решена правильно (результат соответствует условию задачи).

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задание 1, 2, 3) выполнены правильно. Студент ориентируется в коде скриптов, может самостоятельно изменить любые решения, добавлять новые. Задача №3 может проверяться отсрочено, вначале урока следующего занятия.

Оценка «4» - Практическая работа (задание 1, 2) выполнены правильно. Студент ориентируется в коде скриптов, может самостоятельно изменить любой фрагмент кода на языке JavaScript при указании преподавателя. Задача №3 не выполнена.

Оценка «3» - Практическая работа (задание 1, 2) выполнены правильно. Студент ориентируется в коде скриптов, однако для изменения скрипта по просьбе преподавателя нуждается в подсказке (справочная литература, лекционный материал).

Оценка «2» - В практической работе выполнено менее 2 задач. Студент затрудняется в демонстрации возможности изменить код скрипта при указании преподавателя.

Работа № 17.

ТЕМА: Решение задач линейной структуры. Вывод результата в окно браузера

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка программирования PHP, сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (вначале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

Задача №1

1. Создать папку: home/praktikum.ru
2. В папке praktikum.ru - создать файл Index.php
3. В файл Index.php поместить php – скрипт, который выдаст в окно браузера следующую информацию:

«Здравствуйте студент группы П-201, вы начинаете изучать язык PHP»

Задача №2

Задать в коде php – скрипта переменные следующих типов: целое, вещественное, строковое, логическое; вывести в окно браузера значения переменных с указанием их типов. Продемонстрировать возможности работы с функциями определения и изменения типа переменной.

Задача №3

Создать скрипт для решения линейной задачи:

вариант 1

Угол α задан в градусах, минутах и секундах. Найти его величину в радианах (с максимально возможной точностью)

вариант 2

Длина отрезка задана в дюймах ($1 \text{ дюйм} = 2,54$). Перевести значение длины в метрическую систему, т.е. выразить в метрах, сантиметрах и миллиметрах.

вариант 3

Заданы моменты начала и конца некоторого промежутка времени в часах, минутах и секундах. (в пределах одних суток). Найти продолжительность этого промежутка в тех же единицах измерения.

вариант 4

Коммерсант имея стартовый капитал K рублей, занялся торговлей, которая ежемесячно увеличивает капитал на $p\%$. Через сколько лет он накопит сумму S , достаточную для покупки собственного магазина.

вариант 5

Селекционер вывел новый сорт зерновой культуры и снял с опытной делянки K кг семян. Посеяв 1 кг. семян можно за сезон собрать p кг. семян. Через сколько лет селекционер сможет засеять новой культурой поле площадью S га? Если норма высева n кг/га.

примечание: для решения задачи №3 разработать интерфейс ввода данных, средствами HTML – форм, определить формат вывода в соответствии с условиями задачи.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Не переходите к выполнению следующего задания, пока не убедитесь, что задача решена правильно (результат соответствует условию задачи).

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задача 1, 2, 3) выполнена правильно. Студент ориентируется в коде скриптов, может самостоятельно изменить любые решения, добавлять новые.

Для задачи №3 разработана HTML форма для ввода исходных данных. Задача №3 может проверяться отсрочено, вначале урока следующего занятия.

Оценка «4» - Практическая работа (задача 1, 2,3) выполнена правильно. Студент ориентируется в коде скриптов, может самостоятельно изменить любой фрагмент кода на языке PHP при указании преподавателя. Задача №3 выполнена без HTML формы для ввода данных.

Оценка «3» - Практическая работа (задача 1, 2) выполнена правильно. Студент ориентируется в коде скриптов, может самостоятельно изменить любой фрагмент кода на языке PHP при указании преподавателя

Оценка «2» - В практической работе выполнено менее 2 задач. Студент затрудняется в демонстрации возможности изменить код скрипта при указании преподавателя.

Работа № 18.

ТЕМА: Решение задач с использованием оператора ветвления

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка программирования PHP, сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (вначале урока).

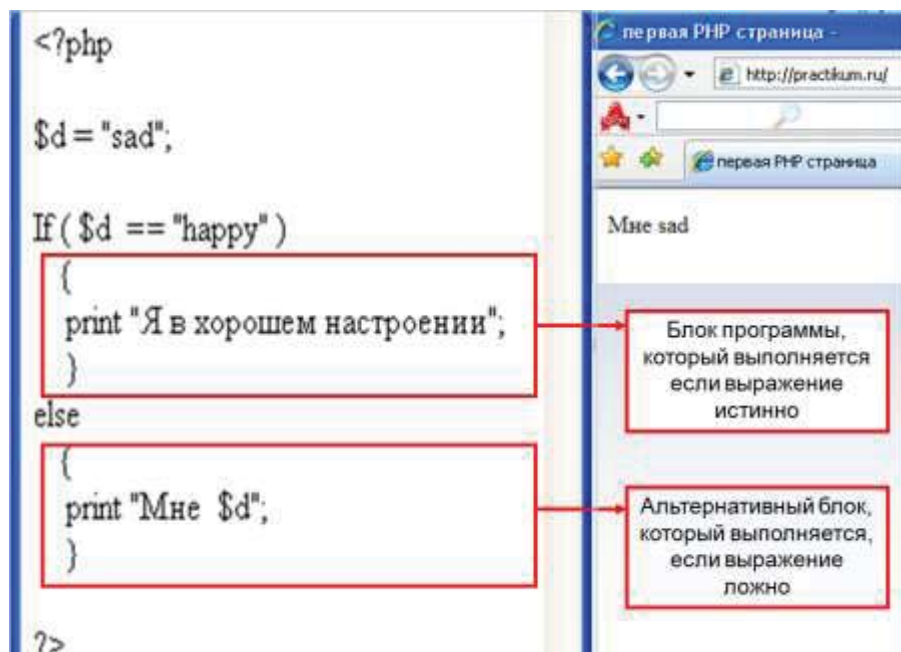
Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

Задача №1

1. Создать папку: home/praktikum.ru
2. В папке praktikum.ru - создать файл Index.php
3. В файл Index.php поместить php – скрипт, который реализует пример рассмотренный на лекционном занятии:



Задача №2

Разработать форму для ввода исходного значения переменной \$d\$ в задаче №1;
Изменить количество вариантов ответа, используя оператор `elseif`.

Задача №3

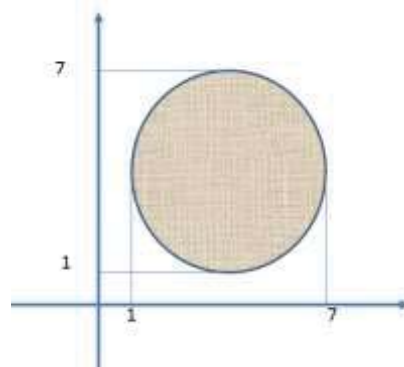
Создать скрипт для решения задачи:

Вариант 1. Даны два действительных числа X и Y . Если $X > Y$, необходимо вычислить произведение этих чисел, иначе их сумму. Разработать РНР скрипт для решения этой задачи. Протестировать алгоритм на трёх различных тестах ($X > Y$, $X = Y$ и $X < Y$).

Вариант 2. Разработать РНР скрипт для подсчета количества отрицательных чисел среди целых чисел a , b , c . Протестировать алгоритм для всех возможных случаев (когда количество отрицательных чисел равно 0, 1, 2 и 3).

Вариант 3. Треугольник задан длинами сторон A , B , C . Разработать РНР скрипт, определяющий, существует ли данный треугольник. Если треугольник существует, то установить вывести в окно браузера фразу «Ура, он существует!», иначе «Такого чуда в природе нет ☺». Для решения этой задачи использовать сложные логические условия. Протестировать скрипт для следующих исходных данных: а) $A=3$, $B=4$, $C=5$ б) $A=1$, $B=1$, $C=1$ в) $A=0$, $B=4$, $C=5$ г) $A=-3$, $B=6$, $C=5$ д) $A=2$, $B=1$, $C=8$

Вариант 4. Точка A задана координатами X, Y . Разработать РНР скрипт, который определит принадлежность точки заштрихованной области и если точка принадлежит заштрихованной области выдает в окно браузера фразу «Попадание точно в цель», при попадании точки на границе областей в окне браузера должна появиться фраза «конечно ДА, но можно и лучше» и в противном случае – «Вы попали в МОЛОКО». Протестировать скрипт для каждого из условий.



Вариант 5. Определить, является ли значение целочисленной переменной X кратным 3.

Вариант 6. Составьте алгоритм и программу, выбирающую из трех чисел то, которое лежит между двумя другими.

Вариант 7. Написать программу, которая выводит пример на умножение двух однозначных чисел, запрашивает ответ пользователя, проверяет его и выводит сообщение «Правильно!» или «Вы ошиблись» и правильный результат.

Например: Сколько будет $6 \cdot 7$? Введите ответ и нажмите <Enter> ? 56 Вы ошиблись.
 $6 \cdot 7 = 42$.

Вариант 8. Написать программу, которая упорядочивает по возрастанию значения трех переменных. Введите в одной строке через запятую значения переменных a , b и c

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Не переходите к выполнению следующего задания, пока не убедитесь, что задача решена правильно (результат соответствует условию задачи).

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задача 1, 2, 3) выполнена правильно. Студент ориентируется в коде скриптов, может самостоятельно изменить любые решения, добавлять новые. Для задачи №3 разработана HTML форма для ввода исходных данных. Задача №3 может проверяться отсрочено, вначале урока следующего занятия.

Оценка «4» - Практическая работа (задача 1, 2,3) выполнена правильно. Студент ориентируется в коде скриптов, может самостоятельно изменить любой фрагмент кода на языке PHP при указании преподавателя. Задача №3 выполнена без HTML формы для ввода данных.

Оценка «3» - Практическая работа (задача 1, 2) выполнена правильно. Студент ориентируется в коде скриптов, может самостоятельно изменить любой фрагмент кода на языке PHP при указании преподавателя

Оценка «2» - В практической работе выполнено менее 2 задач. Студент затрудняется в демонстрации возможности изменить код скрипта при указании преподавателя.

Работа № 19.

ТЕМА: Решение задач с использованием оператора выбора

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка программирования PHP, сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (вначале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

Задача №1

1. Создать папку: home/praktikum.ru
2. В папке praktikum.ru - создать файл Index.php
3. В файл Index.php поместить php – скрипт, который реализует пример рассмотренный на лекционном занятии:


```

<?php
$d = "day";

Switch( $d )
{
case "happy":
    print "Я в хорошем настроении";
    break;
case "sad":
    print "не отчаивайся";
    break;
default:
    print "не понятно, просто $d";
}

?>

```

Задача №2

Разработать форму для ввода исходного значения переменной \$d в задаче №1; модифицировать код таким образом, чтобы переменная \$d отвечала за полученную оценку по практической работе. И реагировала на значение переменной следующим образом:

Значение	Текстовая строка, выводимая в окно браузера
1	Вы неуч, стыдитесь!
2	А стоило ли начинать, если лень учиться?
3	Еще не хорошо, но уже получается!
4	Почти молодец, но все-таки где-то поленился?
5	Молодец! Так держать!
Любое число, отличное от оценки по пятибалльной шкале	Сударь, такой оценки не существует, в каких облаках вы витаете?
Текстовая строка	Сударь, введите оценку в числовом формате!

Задача №3

Создать скрипт для решения задачи:

Вариант 1.

Вычислить стоимость покупки с учетом скидки. Скидка предоставляется по номеру дисконтной карты:

Номер карты	Скидка
101	3%
102	5%
103	5%
104	10%
105	10%
106	15%
107	15%
108	20%
109	20%
110	25%
111	25%
112	25%
113	25%

Вариант 2.

Вычислить стоимость разговора по телефону с учетом скидки, предоставляемой в зависимости от дня недели:

День недели	Скидка
Понедельник	Нет скидки
Вторник	5%
Среда	5%
Четверг	10%
Пятница	15%
Суббота	20%
Воскресенье	25%

Исходные данные, вводимые пользователем:

количество минут разговора (целое число), день недели, тарифный план (предусмотреть разные варианты стоимости одной минуты разговора).

примечание:

для решения задачи №3 разработать интерфейс

ввода данных, средствами HTML – форм, определить формат вывода в соответствии с условиями задачи.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Не переходите к выполнению следующего задания, пока не убедитесь, что задача решена правильно (результат соответствует условию задачи).

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задача 1, 2, 3) выполнена правильно. Студент ориентируется в коде скриптов, может самостоятельно изменить любые решения, добавлять новые. Для задачи №3 разработана HTML форма для ввода исходных данных. Задача №3 может проверяться отсрочено, вначале урока следующего занятия.

Оценка «4» - Практическая работа (задача 1, 2,3) выполнена правильно. Студент ориентируется в коде скриптов, может самостоятельно изменить любой фрагмент кода на языке PHP при указании преподавателя. Задача №3 выполнена без HTML формы для ввода данных.

Оценка «3» - Практическая работа (задача 1, 2) выполнена правильно. Студент ориентируется в коде скриптов, может самостоятельно изменить любой фрагмент кода на языке PHP при указании преподавателя

Оценка «2» - В практической работе выполнено менее 2 задач. Студент затрудняется в демонстрации возможности изменить код скрипта при указании преподавателя.

Работа № 20.

ТЕМА: Решение задач на обработку массива. Вывод заданного массива в окно браузера

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка программирования PHP, сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (вначале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

Задача №1

1. В файл Index.php поместить php – скрипт, который создает одномерный числовой массив и выводит элементы массива в окно браузера.

Требования к формированию массива:

ВАРИАНТ	ТРЕБОВАНИЯ
1.	Количество элементов – 20 Значение элемента: Если значение индекса – четное число, элемент равен 1, в противном случае – элемент равен квадрату индекса.
2.	Количество элементов – 15 Значение элемента: Если значение индекса – не четное число, элемент равен 0, в противном случае – элемент равен квадрату индекса.
3.	Количество элементов – 25 Значение элемента: Если значение индекса – четное число, элемент равен 0, в противном случае – элемент вычисляется по формуле: $a[i] = \sqrt{N} \times (i^2 - 1)$, где N – количество элементов, а i – индекс текущего элемента.
4.	Количество элементов – 36 Значение элемента: Если значение индекса – четное число, элемент равен 0, в противном случае – элемент вычисляется по формуле: $a[i] = \sqrt{N} \times (i^3 + 1)$, где N – количество элементов, а i – индекс текущего элемента.
5.	Количество элементов – 16 Значение элемента: Если значение индекса – четное число, элемент равен 0, в противном случае – элемент вычисляется по формуле: $a[i] = \sqrt{N} \times (i^3 - 1)$, где N – количество элементов, а i – индекс текущего элемента.
6.	Количество элементов – 16 Значение элемента: Если значение индекса – четное число, элемент равен 0, в противном случае – элемент вычисляется по формуле: $a[i] = \frac{\sqrt[3]{2 \times N}}{(i^3 - 1)}$, где N – количество элементов, а i – индекс текущего элемента. Замечание: значение элемента $a[i]$ должно быть целым, т.е. результат вычисления необходимо округлить до ближайшего целого числа.

7.	Количество элементов – 16 Значение элемента: Если значение индекса – четное число, элемент равен 0, в противном случае – элемент вычисляется по формуле: $a[i] = \frac{\sqrt[3]{2 \times N}}{(i^2+1)}$, где N – количество элементов, а i – индекс текущего элемента. Замечание: значение элемента $a[i]$ должно быть целым, т.е. результат вычисления необходимо округлить до ближайшего целого числа.
8.	Количество элементов – 36 Значение элемента: Если значение индекса – четное число, элемент равен 0, в противном случае – элемент вычисляется по формуле: $a[i] = \frac{\sqrt[5]{2 \times N}}{(i^3+2 \times i)}$, где N – количество элементов, а i – индекс текущего элемента. Замечание: значение элемента $a[i]$ должно быть целым, т.е. результат вычисления необходимо округлить до ближайшего целого числа.
9.	Количество элементов – 16 Значение элемента: Если значение индекса – четное число, элемент равен 0, в противном случае – элемент вычисляется по формуле: $a[i] = \frac{\sqrt[2]{2 \times N}}{(i^3+1)^2}$, где N – количество элементов, а i – индекс текущего элемента. Замечание: значение элемента $a[i]$ должно быть целым, т.е. результат вычисления необходимо округлить до ближайшего целого числа.
10.	Количество элементов – 27 Значение элемента: Если значение индекса – четное число, элемент равен 0, в противном случае – элемент вычисляется по формуле: $a[i] = \frac{\sqrt[3]{3 \times N}}{(i^3+i)^3}$, где N – количество элементов, а i – индекс текущего элемента. Замечание: значение элемента $a[i]$ должно быть целым, т.е. результат вычисления необходимо округлить до ближайшего целого числа.
11.	Количество элементов – 16 Значение элемента: Если значение индекса – четное число, элемент равен 0, в противном случае – элемент вычисляется по формуле: $a[i] = \frac{i}{N^{\frac{1}{3}}}$, где N – количество элементов, а i – индекс текущего элемента. Замечание: значение элемента $a[i]$ должно быть целым, т.е. результат вычисления необходимо округлить до ближайшего целого числа.
12.	Количество элементов – 16 Значение элемента: Если значение индекса – четное число, элемент равен 0, в противном случае – элемент вычисляется по формуле: $a[i] = (i^2 + \sqrt{N})^2$, где N – количество элементов, а i – индекс текущего элемента. Замечание: значение элемента $a[i]$ должно быть целым, т.е. результат вычисления необходимо округлить до ближайшего целого числа.
13.	Количество элементов – 16 Значение элемента: Если значение индекса – четное число, элемент равен 0, в противном случае – элемент вычисляется по формуле: $a[i] = (i^2 + \sqrt{N})^{\frac{1}{2}}$, где N – количество элементов, а i – индекс текущего элемента. Замечание: значение элемента $a[i]$ должно быть целым, т.е. результат вычисления необходимо округлить до ближайшего целого числа.
14.	Количество элементов – 36

	Значение элемента: Если значение индекса – четное число, элемент равен 0, в противном случае – элемент вычисляется по формуле: $a[i] = (i^2 + \sqrt{N})^{-2}$, где N – количество элементов, а i – индекс текущего элемента. Замечание: значение элемента $a[i]$ должно быть целым, т.е. результат вычисления необходимо округлить до ближайшего целого числа.
15.	Количество элементов – 16 Значение элемента: Если значение индекса – четное число, элемент равен 0, в противном случае – элемент вычисляется по формуле: $a[i] = \frac{(i^2 + \sqrt{N})^2}{(i^2 - \sqrt{N})^2}$, где N – количество элементов, а i – индекс текущего элемента. Замечание: значение элемента $a[i]$ должно быть целым, т.е. результат вычисления необходимо округлить до ближайшего целого числа.
16.	Количество элементов – 16 Значение элемента: Если значение индекса – четное число, элемент равен 0, в противном случае – элемент вычисляется по формуле: $a[i] = \frac{(i^2 - \sqrt{N})^4}{(i^2 + \sqrt{N})^3}$, где N – количество элементов, а i – индекс текущего элемента. Замечание: значение элемента $a[i]$ должно быть целым, т.е. результат вычисления необходимо округлить до ближайшего целого числа.
17.	Количество элементов – 16 Значение элемента: Если значение индекса – четное число, элемент равен 0, в противном случае – элемент вычисляется по формуле: $a[i] = \frac{(i^2 + \sqrt{N})^2}{2} + (i^2 - \sqrt{N})^2$, где N – количество элементов, а i – индекс текущего элемента. Замечание: значение элемента $a[i]$ должно быть целым, т.е. результат вычисления необходимо округлить до ближайшего целого числа.
18.	Количество элементов – 16 Значение элемента: Если значение индекса – четное число, элемент равен 0, в противном случае – элемент вычисляется по формуле: $a[i] = \frac{(i^3 - \sqrt{N})^2}{2} - (i^3 - \sqrt{N})^2$, где N – количество элементов, а i – индекс текущего элемента. Замечание: значение элемента $a[i]$ должно быть целым, т.е. результат вычисления необходимо округлить до ближайшего целого числа.
19.	Количество элементов – 16 Значение элемента: Если значение индекса – четное число, элемент равен 0, в противном случае – элемент вычисляется по формуле: $a[i] = \sqrt{\frac{(i^4 - \sqrt{N})^2}{4}}$, где N – количество элементов, а i – индекс текущего элемента. Замечание: значение элемента $a[i]$ должно быть целым, т.е. результат вычисления необходимо округлить до ближайшего целого числа.

Задача №2

Для задачи №1, определить формат вывода элементов массива следующим образом:

1-й элемент = 3;
2-й элемент = 5;
....

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Не переходите к выполнению следующего задания, пока не убедитесь, что задача решена правильно (результат соответствует условию задачи).

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задача 1, 2) выполнена правильно. Студент ориентируется в коде скриптов, может самостоятельно изменить любые решения, добавлять новые.

Оценка «4» - Практическая работа (задача 1, 2) выполнена правильно. Студент ориентируется в коде скриптов, может самостоятельно изменить фрагмент кода на языке PHP при указании преподавателя, однако на изменение тратит время и нуждается в использовании учебников, справочников и других источников.

Оценка «3» - Практическая работа (задача 1) выполнена правильно. Студент ориентируется в коде скриптов. При требовании изменить скрипт испытывает трудности. Задача 2 решена неверно или не решена.

Оценка «2» - Задача №1 не выполнена или выполнена с грубыми ошибками, которые студент не может исправить самостоятельно.

Работа № 21.

ТЕМА: Решение задач на обработку ассоциативного массива. Вывод элементов массива в окно браузера

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка программирования PHP, сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (в начале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

Задача №1

1. В файл Index.php поместить php – скрипт, который создает двухмерный числовой массив и выводит элементы массива в окно браузера.

Требования к формированию массива:

Вариант1						Вариант 2					
	1	2	3	4	5		1	2	0	2	1
	16	17	18	19	6		0	1	0	1	0
	15	24	25	20	7		0	2	1	2	0
	14	23	22	21	8		0	1	0	1	0
	13	12	11	10	9		1	2	0	2	1
Вариант3						Вариант 4					
	1	2	3	4	5		1	2	3	4	5
	6	7	8	9	10		6	1	8	9	10
	11	12	13	14	15		11	12	1	14	15
	16	17	18	19	20		16	17	18	1	20
	21	22	23	24	25		21	22	23	24	1
Вариант5						Вариант 6					
	5	4	3	2	1		1	2	3	4	5
	10	9	8	1	6		0	7	8	9	10
	15	14	1	12	11		0	0	13	14	15
	20	1	18	17	16		0	0	0	19	20
	1	24	23	22	21		0	0	0	0	25
Вариант 7						Вариант 8					

<table><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>7</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>13</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>19</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>25</td></tr></table>	1	0	0	0	0	0	7	0	0	0	0	0	13	0	0	0	0	0	19	0	0	0	0	0	25	<table><tr><td>0</td><td>2</td><td>3</td><td>4</td><td>0</td></tr><tr><td>6</td><td>0</td><td>8</td><td>0</td><td>10</td></tr><tr><td>11</td><td>12</td><td>0</td><td>14</td><td>15</td></tr><tr><td>16</td><td>0</td><td>18</td><td>0</td><td>20</td></tr><tr><td>0</td><td>22</td><td>23</td><td>24</td><td>0</td></tr></table>	0	2	3	4	0	6	0	8	0	10	11	12	0	14	15	16	0	18	0	20	0	22	23	24	0
1	0	0	0	0																																															
0	7	0	0	0																																															
0	0	13	0	0																																															
0	0	0	19	0																																															
0	0	0	0	25																																															
0	2	3	4	0																																															
6	0	8	0	10																																															
11	12	0	14	15																																															
16	0	18	0	20																																															
0	22	23	24	0																																															
Вариант 9	Вариант 10																																																		
<table><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>6</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>11</td><td>12</td><td>0</td><td>0</td><td>0</td></tr><tr><td>16</td><td>0</td><td>18</td><td>0</td><td>0</td></tr><tr><td>0</td><td>22</td><td>23</td><td>24</td><td>0</td></tr></table>	0	0	0	0	0	6	0	0	0	0	11	12	0	0	0	16	0	18	0	0	0	22	23	24	0	<table><tr><td>1</td><td>4</td><td>9</td><td>16</td><td>25</td></tr><tr><td>36</td><td>49</td><td>64</td><td>81</td><td>100</td></tr><tr><td>121</td><td>144</td><td>169</td><td>196</td><td>225</td></tr><tr><td>256</td><td>289</td><td>324</td><td>361</td><td>400</td></tr><tr><td>441</td><td>484</td><td>529</td><td>576</td><td>625</td></tr></table>	1	4	9	16	25	36	49	64	81	100	121	144	169	196	225	256	289	324	361	400	441	484	529	576	625
0	0	0	0	0																																															
6	0	0	0	0																																															
11	12	0	0	0																																															
16	0	18	0	0																																															
0	22	23	24	0																																															
1	4	9	16	25																																															
36	49	64	81	100																																															
121	144	169	196	225																																															
256	289	324	361	400																																															
441	484	529	576	625																																															
Вариант 11	Вариант 12																																																		
<table><tr><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td></tr><tr><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td></tr><tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td></tr><tr><td>16</td><td>17</td><td>18</td><td>19</td><td>20</td></tr><tr><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td></tr></table>	5	4	3	2	1	6	7	8	9	10	15	14	13	12	11	16	17	18	19	20	25	24	23	22	21	<table><tr><td>25</td><td>16</td><td>9</td><td>4</td><td>1</td></tr><tr><td>36</td><td>49</td><td>64</td><td>81</td><td>100</td></tr><tr><td>225</td><td>196</td><td>169</td><td>144</td><td>121</td></tr><tr><td>256</td><td>289</td><td>324</td><td>361</td><td>400</td></tr><tr><td>625</td><td>576</td><td>529</td><td>484</td><td>441</td></tr></table>	25	16	9	4	1	36	49	64	81	100	225	196	169	144	121	256	289	324	361	400	625	576	529	484	441
5	4	3	2	1																																															
6	7	8	9	10																																															
15	14	13	12	11																																															
16	17	18	19	20																																															
25	24	23	22	21																																															
25	16	9	4	1																																															
36	49	64	81	100																																															
225	196	169	144	121																																															
256	289	324	361	400																																															
625	576	529	484	441																																															
Вариант13	Вариант 14																																																		

5	4	3	2	1	25	16	9	4	1
6	19	18	17	16	36	361	324	289	256
7	20	25	24	15	49	400	625	576	225
8	21	22	23	14	64	441	484	529	196
9	10	11	12	13	81	100	121	144	169

Замечание: Вывести в окно браузера все элементы в формате
(пример вывода соответствует варианту 14):

$a[1,1] = 25$

$a[1,2] = 16$

.....

$a[5,5] = 169$

Задача №2

Для задачи №1, определить формат вывода построчно через пробел. Каждая строка таблицы в новой строке в окне браузера (пример вывода соответствует варианту 14):

25 16 9 4 1 36 361 324 289 256 49 400 625 576 225 64 441 484 529 196
81 100 121 144 169

Задача №3

Для задачи №1, изменить формат вывода на формирование двумерной таблицы.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Не переходите к выполнению следующего задания, пока не убедитесь, что задача решена правильно (результат соответствует условию задачи).

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задача 1, 2) выполнена правильно. Студент ориентируется в коде скриптов, может самостоятельно изменить любые решения, добавлять новые.

Оценка «4» - Практическая работа (задача 1, 2) выполнена правильно. Студент ориентируется в коде скриптов, может самостоятельно изменить фрагмент кода на языке PHP при указании преподавателя, однако на изменение тратит время и нуждается в использовании учебников, справочников и других источников.

Оценка «3» - Практическая работа (задача 1) выполнена правильно. Студент ориентируется в коде скриптов. При требовании изменить скрипт испытывает трудности. Задача 2 решена неверно или не решена.

Оценка «2» - Задача №1 не выполнена или выполнена с грубыми ошибками, которые студент не может исправить самостоятельно.

Работа № 22.

ТЕМА: Разработка файла обработчика HTML - формы

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка программирования PHP, сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (в начале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

Задача №1 Для задач практической работы №22 разработать HTML форму, для ввода исходных данных при формировании массива (ввод количества элементов).

После нажатия кнопки «Сформировать массив» в окно браузера должен выводиться соответствующий массив.

Задача №2

Для практической работы №23 разработать HTML форму, для ввода исходных данных при формировании массива (ввод количества элементов N^xN).

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Не переходите к выполнению следующего задания, пока не убедитесь, что задача решена правильно (результат соответствует условию задачи).

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задача 1, 2) выполнена правильно. Студент ориентируется в коде скриптов, может самостоятельно изменить любые решения, добавлять новые.

Оценка «4» - Практическая работа (задача 1, 2) выполнена правильно. Студент ориентируется в коде скриптов, может самостоятельно изменить фрагмент кода на языке PHP при указании преподавателя, однако на изменение тратит время и нуждается в использовании учебников, справочников и других источников.

Оценка «3» - Практическая работа (задача 1) выполнена правильно. Студент ориентируется в коде скриптов. При требовании изменить скрипт испытывает трудности. Задача 2 решена неверно или не решена.

Оценка «2» - Задача №1 не выполнена или выполнена с грубыми ошибками, которые студент не может исправить самостоятельно.

Работа № 23.

ТЕМА: Проектирование web – баз данных

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется в тетради с использованием языка программирования PHP.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (в начале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

Задача 1

Определить набор сущностей достаточный для хранения информации в базе данных при решении следующей задачи:

ВАРИАНТ	ЗАДАЧА
1.	Проектирование БД для учета игрушек Интернет - магазина
2.	Проектирование БД для учета сотовых телефонов Интернет - магазина
3.	Проектирование БД для учета товаров (одежда) Интернет - магазина
4.	Проектирование БД для учета товаров (коллекционное оружие) Интернет - магазина
5.	Проектирование БД для учета товаров (лыжи и лыжные аксессуары) Интернет - магазина
6.	Проектирование БД для учета товаров (коллекционные монеты) Интернет - магазина
7.	Проектирование БД для учета товаров (коллекционные марки) Интернет - магазина
8.	Проектирование БД для учета товаров (коллекционные открытки) Интернет - магазина
9.	Проектирование БД для учета товаров (коллекционные куклы) Интернет - магазина
10.	Проектирование БД для учета товаров (Компьютеры и комплектующие) Интернет - магазина
11.	Проектирование БД для учета товаров (книги) Интернет - магазина

12.	Проектирование БД для учета товаров (носители электронной информации) Интернет - магазина
13.	Проектирование БД для учета товаров (мелкая бытовая техника) Интернет - магазина
14.	Проектирование БД для учета товаров (сувениры и декоративные товары) Интернет - магазина
15.	Проектирование БД для учета товаров (коллекционные часы) Интернет - магазина
16.	Проектирование БД для учета товаров (бижутерия) Интернет - магазина
17.	Проектирование БД для учета товаров (драгоценности) Интернет - магазина
18.	Проектирование БД для учета товаров (полудрагоценные камни) Интернет - магазина
19.	Проектирование БД для учета товаров (велосипеды) Интернет - магазина
20.	Проектирование БД для учета товаров (косметика) Интернет - магазина

Задача 2

Для каждой выделенной сущности перечислить набор атрибутов, который необходимо хранить в БД.

Задача 3

Для каждой сущности, выделить первичные ключи

ЗАМЕЧАНИЕ: задачи выполняются в тетради.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Не переходите к выполнению следующего задания, пока не убедитесь, что задача решена правильно (результат соответствует условию задачи).

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задание 1, 2, 3) выполнены правильно.

Оценка «4» - Практическая работа (задание 1, 2, 3) выполнены правильно. Имеются ошибки при определении атрибутов или ключей, которые студент может исправить самостоятельно при указании на них преподавателя

Оценка «3» - Практическая работа (задание 1, 2) выполнены правильно. Имеются ошибки при определении атрибутов, студент может исправить ошибки, только после значительного времени выделенного на доработку. Ориентируется в решении при изменении условий преподавателем.

Оценка «2» - В практической работе выполнено менее 2 задач. Студент испытывает затруднения и не может исправить ошибки даже при указании на них преподавателя.

Работа № 24.

ТЕМА: Создание однотобличной БД

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка программирования PHP, сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (в начале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

Задача 1. Создать файл index.php, в котором записать скрипт, создающий БД на локальном сервере в соответствии с вариантом из практической работы 25.

Задача 2. Создать одну таблицу.

Задача 3. Определить типы полей и ключевое поле

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Не переходите к выполнению следующего задания, пока не убедитесь, что задача решена правильно (результат соответствует условию задачи).

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задание 1, 2, 3) выполнена правильно.

Оценка «4» - Практическая работа (задание 1, 2, 3) выполнена правильно. Имеются ошибки при определении атрибутов или ключей, которые студент может исправить самостоятельно при указании на них преподавателя

Оценка «3» - Практическая работа (задание 1, 2) выполнена правильно. Имеются ошибки при определении атрибутов, студент может исправить ошибки, только после значительного времени выделенного на доработку. Ориентируется в решении при изменении условий преподавателем.

Оценка «2» - В практической работе выполнено менее 2 задач. Студент испытывает затруднения и не может исправить ошибки даже при указании на них преподавателя.

Работа № 25.

ТЕМА: Функции получения доступа, функции открытия/закрытия соединения с БД

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка программирования PHP, сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (в начале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

Задача 1 Создать файл `bd.php`, в котором записать скрипт, создающий подключение к БД на локальном сервере в соответствии с вариантом из практической работы 25.

Задача 2 Реализовать все таблицы в соответствии со своим вариантом.

Задача 3 Определить нового пользователя и назначить ему полные права.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Не переходите к выполнению следующего задания, пока не убедитесь, что задача решена правильно (результат соответствует условию задачи).

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задание 1, 2, 3) выполнена правильно.

Оценка «4» - Практическая работа (задание 1, 2, 3) выполнена правильно. Имеются ошибки при определении атрибутов или ключей, которые студент может исправить самостоятельно при указании на них преподавателя

Оценка «3» - Практическая работа (задание 1, 2) выполнена правильно. Имеются ошибки при определении атрибутов, студент может исправить ошибки, только после значительного времени выделенного на доработку. Ориентируется в решении при изменении условий преподавателем.

Оценка «2» - В практической работе выполнено менее 2 задач. Студент испытывает затруднения и не может исправить ошибки даже при указании на них преподавателя.

Работа № 26.

ТЕМА: Инструмент PHP MyAdmin

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка программирования PHP, сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (вначале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

1. Запустить СУБД MySQL:

Рабочий стол | ярлык StartDenwer ► окно запуска Denwer; дождаться закрытия окна

2. Открыть среду PHPMyAdmin:

Запустить браузер ► адресная строка ← `http://localhost` ► окн. «Ура, заработало!» | выбр. `http://localhost/Tools/phpMyAdmin` ► окн. phpMyAdmin; Server: localhost

3. Создать новую базу данных *Education*:

ст. MySQL localhost | Create database: | ввести *Education* | Collation ← `utf8_unicode_ci` |

кн. Create ► окн. Server: localhost; Database: Education |

Database Education has been created.

4. В базе данных с помощью конструктора таблиц создать структуру таблицы *Students* (справочник студентов):

– создать новую таблицу *Students* с 11 полями:

окн. Server: localhost; Database: Education | Create new table on database Education: | Name ← *Students*; Number of fields ← 11 | кн. Go ► окн. Server: localhost; Database: Education; Table: Students;

– задать поле идентификатора студента (счетчик, первичный ключ):

ст. Field ← *stud_ID*; Type ← *int*; уст. *A_I*; Index ← *Primary*; перейти на след. строку;

– задать поле номера зачетной книжки (уникальное, 6 цифр):

ст. Field ← *no_zk*; Type ← *varchar*; Length / Values¹ ← 6;

Index ← *Unique*; перейти на след. строку;

– задать поля фамилии, имени, отчества (строковые):

ст. Field ← *fam*; Type ← *varchar*; Length / Values¹ ← 20; перейти на след. строку;

аналогично – для имени (*im*) и отчества (*ot*);

– создать уникальный индекс *ФИО* из полей *fam*, *im*, *ot*:

ст. Field *fam* ← Index ← *Index*; перейти на след. строку;

ст. Field *im* ← Index ← *Index*; перейти на след. строку;

ст. Field *ot* ← Index ← *Index*;

– задать поле кода специальности (до 7 букв):

ст. Field ← *spec*; Type ← *varchar*; Length / Values¹ ← 7; перейти на след. строку;

– задать поле курса (одна цифра от 1 до 6):

ст. Field $\leftarrow$ *kurs*; Type $\leftarrow$ *varchar*; Length / Values¹ $\leftarrow$ 1; перейти на след.строку;

– задать поле номера группы (две цифры):

ст. Field $\leftarrow$ *gr*; Type $\leftarrow$ *varchar*; Length / Values¹ $\leftarrow$ 2; перейти на след.строку;

– задать поле даты рождения:

ст. Field $\leftarrow$ *data_r*; Type $\leftarrow$ *date*; перейти на след.строку;

– задать поле биографии (мемо):

ст. Field $\leftarrow$ *biogr*; Type $\leftarrow$ *longtext*; уст. *Null*; перейти на след. строку;

– задать поле фотографии (объект):

ст. Field $\leftarrow$ *foto*; Type $\leftarrow$ *blob*; уст. *Null*; перейти на след. строку;

кн. Save ► окн. Table `Education`.`Students` has been created.

5. Аналогичным образом создать структуру таблицы *Subjects* (справочник предметов):

– создать новую таблицу базы данных *Education*:

пан. phpMyAdmin | Database $\leftarrow$ *Education* (1);

пан. Create new table on database Education: | Name $\leftarrow$ *Subjects*; Number of fields $\leftarrow$ 5 |

кн. Go ► окн. Server: localhost; Database: Education; Table: Subjects.

– задать поле идентификатора предмета (счетчик, первичный ключ):

ст. Field $\leftarrow$ *predm_ID*; Type $\leftarrow$ *int*; уст. *A_I*; Index $\leftarrow$ *Primary*; перейти на след. строку;

– аналогичным образом создать остальные поля таблицы:

– *name* (название предмета (строковое));

– *cycle* (цикл дисциплин, к которым относится предмет (строковое));

– *hours*(количество часов (числовое));

– *dep* (название кафедры, на которой ведется преподавание предмета (строковое));

– сохранить структуру таблицы:

кн. Save ► окн. Table `Education`.`Subjects` has been created.

6. Аналогичным образом создать структуру таблицы *Uspev* (успеваемость студентов по предметам) с полями:

– *stud*(идентификатор студента (тип такой же, как в таблице *Студенты*,но не счетчик));

– *predm* (идентификатор предмета (тип такой же, как в таблице *Предметы*,но не счетчик));

– *ocenka* (оценка (символ));

– *data*(дата (дата)).

– ввести составной первичный ключ (поля идентификаторов студента и предмета):

Поле *stud*| Index $\leftarrow$ *Primary*; перейти на след.строку;

Поле *predm*| Index $\leftarrow$ *Primary*;

– сохранить структуру таблицы:

кн. Save ► окн. Table `Education`.`Uspev` has been created.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задание 1, 2, 3, 4, 5, 6) выполнена правильно.

Оценка «4» - Практическая работа (задание (задание 1, 2, 3, 4, 5) выполнена правильно. Имеются ошибки при определении атрибутов или ключей, которые студент может исправить самостоятельно при указании на них преподавателя

Оценка «3» - Практическая работа (задание (задание 1, 2, 3, 4) выполнена правильно. Имеются ошибки при определении атрибутов, студент может исправить ошибки, только после значительного времени выделенного на доработку. Ориентируется в решении при изменении условий преподавателем.

Оценка «2» - Задания с 4 по 6 не выполнены

Работа № 27.

ТЕМА: Создание однотабличной БД средствами PHP MyAdmin

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка программирования PHP, сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (вначале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

1. Запустить СУБД MySQL:

Рабочий стол | ярлык StartDenwer ► окно запуска Denwer; дождаться закрытия окна

2. Открыть среду PHPMyAdmin:

Запустить браузер ► адресная строка ← `http://localhost` ► окн. «Ура, заработало!» | выбр. `http://localhost/Tools/phpMyAdmin` ► окн. phpMyAdmin; Server: localhost

3. Открыть базу данных *Education* (см.Практическую работу №28):

4. Задать ограничения для столбцов таблицы *Students* в форме триггера *students_constraints*, срабатывающего при попытке вставки новой записи в таблицу:

– создать новый SQL-запрос для таблицы *Students*:

пан. phpMyAdmin | Database: ← *Education* (1); выбр. *Students*; вкл. SQL ►

окн. Run SQL query/queries on database Education | удалить текст «SELECT * FROM `Students` WHERE 1»;

– ввести SQL-запрос на удаление триггера *students_constraints_insert* в случае его существования (для возможности редактирования текста триггера):

`DROP TRIGGER IF EXISTS `students_constraints_insert` ;`

– ввести «заголовочный» код создания триггера:

`delimiter ||`

`CREATE TRIGGER `students_constraints_insert` BEFORE INSERT ON `Students``

`FOREACH`

`ROWBEGIN`

– ввести код, проверяющий, что значение поля *kurs* лежит в пределах от 1 до 6 и заменяющий его на 0 в противном случае:

```
IF not(NEW.kurs >= 1 and NEW.kurs <= 6) THEN  
SET NEW.kurs = 0;  
END IF;
```

– ввести код, проверяющий, что значение поля *spec* сводится русскими буквами и заменяющий его на 0 в противном случае:

```
IF not(NEW.spec >= 'А' AND NEW.spec <= 'я') THEN  
SET NEW.spec = 0;  
END IF;
```

– ввести код, проверяющий, что значение поля *gr* лежит в пределах от 1 до 99 и заменяющий его на 0 в противном случае:

```
IF not(NEW.gr >= '01' AND NEW.gr <= '99') THEN  
SET NEW.gr = 0;  
END IF;
```

– ввести завершающий код триггера:

```
END ||  
delimiter ;
```

5. Аналогичным образом запрограммировать триггер *students_constraints_update*, срабатывающий при попытке изменения записи в таблице.

6. Для таблиц *Subjects* и *Uspev* написать по два триггера, проверяющие целостность данных при вставке и изменении данных.

7. Установить связи между таблицами и указать правила ссылочной целостности:

– изменить тип таблицы *Student*:

пан. phpMyAdmin | Database: ← *Education* (1); выбр. *Students*; вкл. Operations ►
StorageEngine ← *InnoDB*;

аналогично для всех остальных таблиц.

– указать поле *stud* в качестве внешнего ключа таблицы *Students*, ссылающегося на первичный ключ таблицы *Students*:

пан. phpMyAdmin | Database: ← *Education* (3); выбр. *Uspev*; вкл. Operations ►
выбр. Relation View ► ф. Links to | стр. *stud*; Foreign key (InnoDB) ←
'Education'. 'Students'; onDelete ← cascade; onUpdate ← cascade;

аналогичным образом задать внешний ключ, ссылающийся на первичный ключ таблицы *Subjects*.

8. Наполнить базу данных сведениями о студентах (не менее 5), предметах (не менее 3) и оценках (не менее 10):

пан. PhpMyAdmin | Database: ← *Education* (3); выбр. таблицу; вкл. Insert ► заполнить столбец *Value*.

Освоить приемы изменения и удаления полей и записей. Проверить работоспособность ограничений значений полей, уникальности и др., предусмотренные при задании структуры базы данных. Проверить работоспособность ссылочной целостности, удаляя, изменяя и вставляя данные.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задание 1, 2, 3, 4, 5, 6, 7, 8) выполнена правильно.

Оценка «4» - Практическая работа (задание (задание 1, 2, 3, 4, 5, 6, 7) выполнена правильно. Имеются ошибки при определении триггера, которые студент может исправить самостоятельно при указании на них преподавателя

Оценка «3» - Практическая работа (задание (задание 1, 2, 3, 4, 5, 6) выполнена правильно. Имеются ошибки при определении триггера, студент может исправить ошибки, только после значительного времени выделенного на доработку. Ориентируется в решении при изменении условий преподавателем.

Оценка «2» - Задания с 4 по 6 не выполнены

Работа № 28.

ТЕМА: Типы данных, привилегии

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка программирования PHP, сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (вначале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

1. Создать просмотр для вывода кратких сведений о студентах (идентификатор, номер зачетки, фамилия и инициалы, идентификатор группы):

– создать новый SQL-запрос для таблицы *Students*:

пан. phpMyAdmin | Database: ← *Education* (3); выбр. *Students*; вкл. SQL ►

окн. Run SQL query/queries on database Education | удалить текст «SELECT * FROM `Students` WHERE 1»;

– ввести программный код запроса на извлечение данных:

```
SELECT `stud_ID`, `no_zk`, CONCAT(fam, ' ', SUBSTRING(im, 1), ' ', SUBSTRING(ot, 1), ' ') AS FIO, CONCAT(spec, ' ', kurs, gr) AS Gruppa FROM `Students` WHERE 1
```

– выполнить запрос

окн. Run SQL query/queries on database Education | кн. Go ► результаты на экране;

– сформировать представление по результатам запроса:

ф. Query results operations | CREATE VIEW | VIEW name ← *Students_info* | кн. Go.

2. Аналогичным образом создать представление для вывода сведений об успеваемости студентов из таблицы успеваемости с указанием сведений о студенте из запроса *Students_info* и сведений о предмете из таблицы предметов. Результат должен содержать следующие поля: ФИО студента, Группа, Предмет, Дата, Оценка. Назвать представление как *Students_uspev*. Программный код запроса:

```
SELECT `no_zk`, `FIO`, `Gruppa`, `ocenka`, `data` FROM `students_info`, `uspev`, `subjects`  
WHERE `students_info`.`stud_ID` = `uspev`.`stud` AND  
`subjects`.`predm_ID` = `uspev`.`predm`
```

3. Создать хранимую процедуру для вывода кратких сведений о студенте (идентификатор, фамилия и инициалы, идентификатор группы) по номеру его зачетной книжки:

– создать новый SQL-запрос для базы данных *Education*:

пан. phpMyAdmin | Database: ← *Education* (3); вкл. SQL ►

окн. Run SQL query/queries on database *Education*;

– ввести программный код хранимой процедуры:

```
DELIMITER $$
```

```
CREATE PROCEDURE studs(IN IN_ZK VARCHAR(6),OUT V_ID INT, OUT V_ZK  
VARCHAR(6),OUT V_FIO VARCHAR(25),OUT V_GR VARCHAR(25))
```

```
BEGIN
```

```
    SELECT stud_id, no_zk, concat(fam,' ',substring(im,1),'',substring(ot,1),''),  
    concat(spec,'',kurs,gr)
```

```
into V_ID, V_ZK, V_FIO, V_GR
```

```
    FROM Students
```

```
    WHERE `no_zk` = IN_ZK;
```

```
END;
```

```
$$
```

```
DELIMITER ;
```

– проверить работоспособность хранимой процедуры:

пан. phpMyAdmin | Database: ← *Education* (3); вкл. SQL ►

окн. Run SQL query/queries on database *Education*; ввести код:

```
CALL studs('номер зачетной книжки',@V_ID,@V_ZK,@V_FIO,@V_GR);
```

```
SELECT @V_ID,@V_ZK,@V_FIO,@V_GR;
```

кн. Go ► Результат на экране.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задание 1, 2, 3) выполнена правильно.

Оценка «4» - Практическая работа (задание 1, 2) выполнена. Имеются ошибки при выполнении задания 3, которые студент может исправить самостоятельно при указании на них преподавателя

Оценка «3» - Практическая работа (задание 1) выполнена правильно. Имеются ошибки при выполнении задания 2,3 которые студент может исправить, после значительного времени выделенного на доработку. Ориентируется в решении при изменении условий преподавателем.

Оценка «2» - Задания с 2,3 не выполнены, самостоятельно выполнить не может.

Работа № 29.

ТЕМА: Подключение к БД с помощью РНР

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка программирования РНР, сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (в начале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

1. Запустить программную оболочку Denwer:

Рабочий стол | ярлык StartDenwer ► окно запуска Denwer; дождаться закрытия окна

2. Проверить корректность работы Denwer:

Запустить браузер ► адресная строка ← http://localhost ► окн. «Ура, заработало!»

3. Создать виртуальную директорию для хранения файлов веб-приложения:

Мой компьютер | Локальный диск D | WebServers | denwer | www | denwer |

создать папку под именем группа (например, lab31_P201).

4. Создать файл index.php начальной страницы веб-приложения:

Директория lab3_группа | Создать текстовый документ |

Переименовать файл как index.php

5. Открыть файл index.php с помощью редактора Notepad++:

Директория lab3_группа | Файл index.php | пр. кн. мыши;

выбр. Edit with Notepad++ ► окноредактора Notepad++

6. Ввести HTML-разметку страницы index.php:

```
<HTML>
  <HEAD><TITLE>Лабораторная работа 3</TITLE></HEAD>
  <BODY>
    <H2>Информация о студентах</H2>
    <FORM id="form" method="POST" action="index.php">
      <TABLE border="1" width="60%">
        <TR>
          <TH width="10%">Код</TH>
          <TH width="20%">Зачетная книжка</TH>
          <TH width="40%">ФИО</TH>
          <TH width="30%">Группа</TH>
        </TR>
        <TR align="center">
          <TD width="10%">Значение кода</TD>
          <TD width="20%">Значение зачетки</TD>
          <TD width="40%">Значение ФИО</TD>
          <TD width="30%">Значение группы</TD>
        </TR>
      </TABLE>
```

```

        <BR/> Номер зачетной книжки: <input name="zk" type="text"/>
    <input type="submit" value="Запрос"/>
</FORM>
</BODY>
</HTML>

```

7. Проверить работоспособность созданной страницы:

Браузер со стартовой страницей Denwer | строка адреса «http://localhost/denwer/» |
 дополнить названием виртуальной директории (см. п. 3),
 например, как http://localhost/denwer/ lab3_asoi338 ► страница на экране

8. Создать PHP-сценарий соединения с базой данных Education:

– создать в виртуальной директории и открыть файл connection.php:

Директория lab3_группа | Создать текстовый документ |
 Переименовать файл как connection.php | пр. кн. мыши;
 выбр. Edit with Notepad++ ► окно редактора Notepad++

– ввести программный код сценария:

– ввести открывающий тег сценария:

```
<?php
```

– ввести программный код соединения с локальным сервером:

```
$link = @mysql_connect("localhost", "root")
or die("Невозможно соединиться с сервером");
```

– ввести программный код соединения с базой данных Education:

```
$db=@mysql_select_db("Education") or die("Нет такой базы данных");
```

– ввести закрывающий тег сценария:

```
?>
```

– сохранить сценарий.

9. Дополнить файл index.php PHP-инструкциями:

– подключить сценарий соединения с базой данных:

окно редактора Notepad++ | вкл. Index.php | поставить курсор
 до первой строки <HTML> | ввести код:

```
<?php
```

```
include("connection.php");
```

```
?>
```

– проверить работоспособность сценария, обновив в браузере страницу index.php.

– выполнить запрос к представлению Students_info базы данных:

окно редактора Notepad++ | вкл. Index.php | поставить курсор
 послестроки include("connection.php"); | ввести код:

```
$sql = "SELECT * FROM `students_info`";
```

```
$query = mysql_query($sql);
```

– организовать цикл по строкам таблицы:

окно редактора Notepad++ | вкл. Index.php | поставить курсор перед второй строкой <TR> | ввести код:

```
<?php
    for($i=0;$i<$count;$i++)
    {
?>
```

поставить курсор после второй строки </TR> | ввести код:

```
<?php
    }
?>
```

– выполнить подстановку результатов запроса в строки таблицы:

окно редактора Notepad++ | вкл. Index.php | поставить курсор

перед второй строкой с текстом «Значение кода» | заменить текст «Значение кода» на:

```
<?php echo mysql_result($query,$i,stud_id);?>
```

аналогичным образом заменить фрагменты текста «Значение зачетки»,

«Значение ФИО» и «Значение группы» на фрагменты кода:

```
<?php echo mysql_result($query,$i,no_zk);?>
```

```
<?php echo mysql_result($query,$i,FIO);?>
```

```
<?php echo mysql_result($query,$i,Gruppa);?>
```

– сохранить файл index.php.

– проверить работоспособность сценария, обновив в браузере страницу index.php.

– добавить инструкции фильтрации данных по номеру зачетной книжки:

– добавить условие, проверяющее, был ли запрос на фильтрацию:

окно редактора Notepad++ | вкл. Index.php | поставить курсор

перед строкой с текстом «\$sql = "SELECT * FROM `students\_info`";» |

ввести код:

```
if(!($_POST['zk']) or $_POST['zk']=="")
{
```

– закрыть условный блок (если запроса не было, то будут показаны все записи):

окно редактора Notepad++ | вкл. Index.php | поставить курсор

после строки с текстом «\$count = mysql_num_rows(\$query);» | ввести «}»

– добавить код, выполняющий запрос на фильтрацию:

окно редактора Notepad++ | вкл. Index.php | поставить курсор

после строки с текстом «};» | ввести код

```
else
```

```
{
```

```
    $sql = "SELECT * FROM `students_info` where
            `no_zk`='".$_POST['zk']."'";
```

```
    $query = mysql_query($sql);
```

```
$count = mysql_num_rows($query);  
}
```

– сохранить файл index.php.

– проверить работоспособность сценария, обновив в браузере страницу index.php.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задание 1-9) выполнена правильно.

Оценка «4» - Практическая работа (задание 1-8) выполнена/ ошибки студент может исправить самостоятельно при указании на них преподавателя

Оценка «3» - Практическая работа (задание 1-7) выполнена правильно. Ошибки студент может исправить, после значительного времени выделенного на доработку. Ориентируется в решении при изменении условий преподавателем.

Оценка «2» - Выполнено менее 7 заданий

Работа № 30.

ТЕМА: Выборка данных - оператор SELECT

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка программирования PHP, сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (вначале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

1. Создать базу данных Forum. В БД Forum создать три таблицы: users (пользователи), topics (темы) и posts (сообщения)

Замечание: структуру таблиц, связи определить самостоятельно.

2. Заполните созданные таблицы (не менее 5 записей в каждой);

3. создайте файл Index.php в котором:

1.1.подключитесь к созданной БД;

1.2.осуществите выборку данных из каждой таблицы;

Замечание: структуру запроса, условия продумайте самостоятельно

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Не переходите к выполнению следующего задания, пока не убедитесь, что задача решена правильно (результат соответствует условию задачи).

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задание 1-3.2) выполнена правильно.

Оценка «4» - Практическая работа (задание 1-3.2) выполнена. Допущенные ошибки студент может исправить самостоятельно при указании на них преподавателя

Оценка «3» - Практическая работа (задание 1-3.2) выполнена правильно. Допущенные ошибки студент может исправить, после значительного времени выделенного на доработку. Ориентируется в решении при изменении условий преподавателем.

Оценка «2» - Выполнено менее 3 заданий

Работа № 31.

ТЕМА: Выборка данных (сложный запрос, ограничение на количество выбранных записей, выборка в алфавитном порядке)

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка программирования PHP, сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (в начале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

1. Открыть базу данных Forum.
2. создайте файл Index.php в котором:
 - 1.1.подключитесь к созданной БД;
 - 1.2.осуществите выборку данных из нескольких таблиц (продемонстрируйте не менее 3 примеров);
 - 1.3.установите параметры выборки (количество записей, сортировка)

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Не переходите к выполнению следующего задания, пока не убедитесь, что задача решена правильно (результат соответствует условию задачи).

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задание 1-2.3) выполнена правильно.

Оценка «4» - Практическая работа (задание 1-2.3) выполнена. Допущенные ошибки студент может исправить самостоятельно при указании на них преподавателя

Оценка «3» - Практическая работа (задание 1-2.2) выполнена правильно. Допущенные ошибки студент может исправить, после значительного времени выделенного на доработку. Ориентируется в решении при изменении условий преподавателем.

Оценка «2» - Выполнено менее 3 заданий

Работа № 32.

ТЕМА: Выборка данных, вывод данных в цикле

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка программирования РНР, сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (вначале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

Задание 1

Спроектировать структуру базы данных согласно варианту:

Вариант 1. Спроектировать структуру базы данных о студентах, для их распределения по местам практики: фамилия, год рождения, пол, группа, факультет, средний балл, место работы, город.

Вариант 2. Спроектировать структуру базы данных об автомобилях: номер, год выпуска, марка, цвет, состояние, фамилия владельца, адрес.

Вариант 3. Спроектировать структуру базы данных о квартирах, предназначенных для продажи: район, этаж, площадь, количество комнат, сведения о владельце, цена.

Вариант 4. Спроектировать структуру базы данных о книгах, купленных библиотекой: название, автор, год издания, адрес автора, адрес издательства, цена, книготорговая фирма.

Вариант 5. Спроектировать структуру базы данных о сотрудниках, имеющих компьютер: фамилия, номер комнаты, название отдела, данные о компьютерах.

Вариант 6. Спроектировать структуру базы данных о заказах, полученных сотрудниками фирмы: фамилия, сумма заказа, наименование товара, название фирмы - клиента, фамилия заказчика.

- Вариант 7. Спроектировать структуру базы данных об оценках, полученных студентами на экзаменах: фамилия, группа, предмет, номер билета, оценка, преподаватель.
- Вариант 8. Спроектировать структуру базы данных о преподавателях кафедры: фамилия, должность, степень, номер комнаты, читаемые курсы.
- Вариант 9. Спроектировать структуру базы данных об авторах web-сайта и их статьях: имя, адрес, учетная запись, пароль, тема, заголовок, текст статьи, иллюстрации.
- Вариант 10. Спроектировать структуру базы данных о списке рассылки и подписчиках: тема и содержание письма, дата отправки, имена и адреса подписчиков, их учетные записи и пароли.

задание 2

создайте файл Index.php в котором:

- 1.1.подключитесь к созданной БД;
- 1.2.осуществите выборку данных из каждой таблицы;
- 1.3.осуществите выборку данных из нескольких таблиц (продемонстрируйте не менее 3 примеров);
- 1.4.установите параметры выборки (количество записей, сортировка)

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Не переходите к выполнению следующего задания, пока не убедитесь, что задача решена правильно (результат соответствует условию задачи).

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задание 1-2.4) выполнена правильно.

Оценка «4» - Практическая работа (задание 1-2.3) выполнена. Допущенные ошибки студент может исправить самостоятельно при указании на них преподавателя

Оценка «3» - Практическая работа (задание 1-2.2) выполнена правильно. Допущенные ошибки студент может исправить, после значительного времени выделенного на доработку. Ориентируется в решении при изменении условий преподавателем.

Оценка «2» - Выполнено менее 2 заданий

Работа № 33.

ТЕМА: Редактирование, обновление и удаление данных

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка программирования PHP, сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (в начале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

Задание 1

Выполните следующие действия:

1. Назначение различных привилегий пользователей MySQL

Для изучения привилегий, назначаемых пользователям сервера MySQL запустить графическую оболочку phpMyAdmin. Для этого запустить пакет Денвер, затем в адресной строке браузера выполнить localhost и выбрать пункт меню «phpMyAdmin — администрирование СУБД MySQL».

В сервере MySQL привилегии назначаются при создании пользователей. При использовании графической оболочки phpMyAdmin доступ к интерфейсу управления привилегиями пользователей осуществляется из главного меню (рис.1).

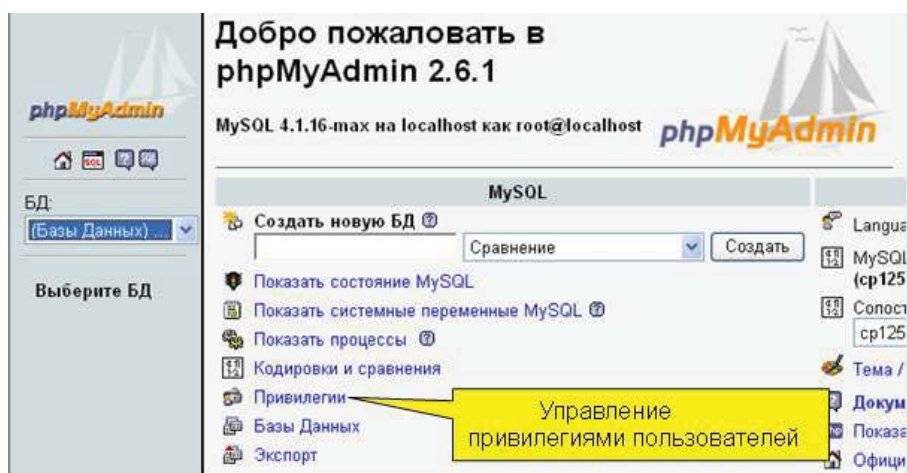


Рис. 1. Главное меню phpMyAdmin

На рис. .2 представлен пример списка пользователей MySQL, вызываемый по гиперссылке **Привилегии**

	Пользователь	Хост	Пароль	Глобальные привилегии	Предоставлять	
☐	admin	localhost	Да	ALL PRIVILEGES	Да	
☐	lwius	localhost	Да	ALL PRIVILEGES	Да	
☐	myportal	localhost	Да	USAGE	Нет	
☐	nirsspb_gost	localhost	Да	SELECT	Нет	
☐	nirsspb_grishin	localhost	Да	ALL PRIVILEGES	Да	
☐	nirsspb_phpadm	%	Да	ALL PRIVILEGES	Да	
☐	nirsspb_vus	localhost	Да	SELECT, INSERT, UPDATE	Нет	
☐	omistru_gost	localhost	Да	SELECT, INSERT, UPDATE, DELETE, CREATE, DROP	Нет	
☐	omistru_omistru	localhost	Да	SELECT, INSERT, UPDATE, DELETE, DROP	Нет	
☐	root	%	Нет	ALL PRIVILEGES	Да	
☐	user_bd	localhost	Да	USAGE	Нет	

Примечание: привилегии MySQL задаются по-английски

Рис. .2. Пример таблицы пользователей сервера MySQL

Назначение полей таблицы пользователей:

Пользователь — имя пользователя сервера MySQL.

Хост — имя хоста, с которого пользователь может зайти на сервер MySQL. Если сервер MySQL и Web-сервер находятся на одном хосте (на одном компьютере), то используется имя **localhost**.

Пароль — есть ли пароль у пользователя.

Глобальные привилегии — привилегии, назначенные данному пользователю на весь сервер MySQL; значение USAGE означает, что у данного пользователя нет глобальных привилегий, но есть привилегии на конкретные базы данных.

Предоставлять — есть ли право предоставлять свои привилегии другим пользователям (при помощи оператора GRANT).

Примечание: привилегии MySQL задаются по-английски
Отметить все Снять отметку со всех

Данные	Структура	Администрирование
☐ SELECT	☐ CREATE	☐ GRANT
☐ INSERT	☐ ALTER	☐ SUPER
☐ UPDATE	☐ INDEX	☐ PROCESS
☐ DELETE	☐ DROP	☐ RELOAD
☐ FILE	☐ CREATE TEMPORARY TABLES	☐ SHUTDOWN
		☐ SHOW DATABASES
		☐ LOCK TABLES
		☐ REFERENCES
		☐ EXECUTE
		☐ REPLICATION CLIENT
		☐ REPLICATION SLAVE

Предел ресурсов
Замечание: Установка этих опций в 0 (ноль) удаляет лимит.

MAX QUERIES PER HOUR	
MAX UPDATES PER HOUR	
MAX CONNECTIONS PER HOUR	

Рис. 3. Форма назначения привилегий пользователя MySQL

Каждому пользователю MySQL можно назначить как глобальные привилегии, распространяющиеся на всю СУБД, в том числе на все базы данных в этой СУБД, так и привилегии специфичные для конкретной базы данных. На рис. 3 показан список возможных глобальных привилегий пользователей MySQL, который вызывается как при редактировании привилегий существующего пользователя, так и при создании нового пользователя MySQL.

MySQL предоставляет различные группы привилегий, перечисленные ниже.

Привилегии на работу с данными:

SELECT — права на извлечение данных из таблиц.

INSERT — права на вставку записей в таблицу.

UPDATE — права на обновление записей в таблице.

DELETE — права на удаление записей из таблицы.

FILE — права на выполнение команд LOAD DATA INFILE и SELECT ... INTO OUTFILE. Такой пользователь может читать и записывать любой файл на сервер MySQL.

Привилегии на работу со структурой баз данных:

CREATE — права на создание баз данных и таблиц.

ALTER — права на модификацию таблиц — добавление, удаление и изменение полей.

INDEX — права на создание и удаление индексов.

DROP — права на удаление баз данных и таблиц.

CREATE TEMPORARY TABLES — права на создание временных таблиц при помощи оператора **CREATE TEMPORARY TABLE**.

Привилегии на администрирование сервера MySQL:

GRANT — позволяет предоставлять другим пользователям привилегии, которыми обладает сам при помощи оператора **GRANT**.

SUPER — права на остановку любого процесса, запущенного на сервере (при помощи команды **kill**).

PROCESS — права на вывод списка процессов на сервере (при помощи команды **processlist**).

RELOAD — права на перезагрузку сервера MySQL.

SHUTDOWN — права на остановку сервера MySQL.

SHOW DATABASES — права на просмотр списка баз данных, имеющихся на сервере MySQL.

LOCK TABLES — права использования оператора **LOCK TABLES** на таблицах, для которых есть привилегия **SELECT**.

REFERENCES — права на создание связей между таблицами при помощи внешних и родительских ключей.

EXECUTE — разрешает пользователю запускать хранимые процедуры (для MySQL 5.0).

REPLICATION CLIENT — предоставляет пользователю право запрашивать местонахождение головного и подчиненных серверов.

REPLICATION SLAVE — необходимо для подчиненных серверов при репликации (для чтения информации из бинарных журналов головного сервера).

Привилегии **CREATETEMPORARYTABLES**, **EXECUTE**, **LOCKTABLES**, **REPLICATION** ..., **SHOWDATABASES** и **SUPER** являются новыми для версии 4.0.2. Чтобы воспользоваться этими новыми привилегиями после обновления до версии 4.0.2, необходимо запустить скрипт `mysql_fix_privilege_tables`.

В более старых версиях MySQL привилегия **PROCESS** предоставляет такие же права, как и новая привилегия **SUPER**.

2. Разграничение привилегий доступа к учебной базе данных различных групп пользователей MySQL

1. В MySQL создать двух пользователей: **admin** и **user**. Ограничить их привилегии только учебной базой данных. С этой целью с главной страницы phpMyAdmin (см. рис. 7.1) следует зайти в раздел Привилегии (рис. 4) и по очереди добавить новых пользователей, рекомендуемый пароль: 1234. Пользователю **admin** дать все привилегии на свою базу данных, а пользователю **user** — только на выборку данных из этой базы (**select**). Глобальные привилегии обоим пользователям не назначать!

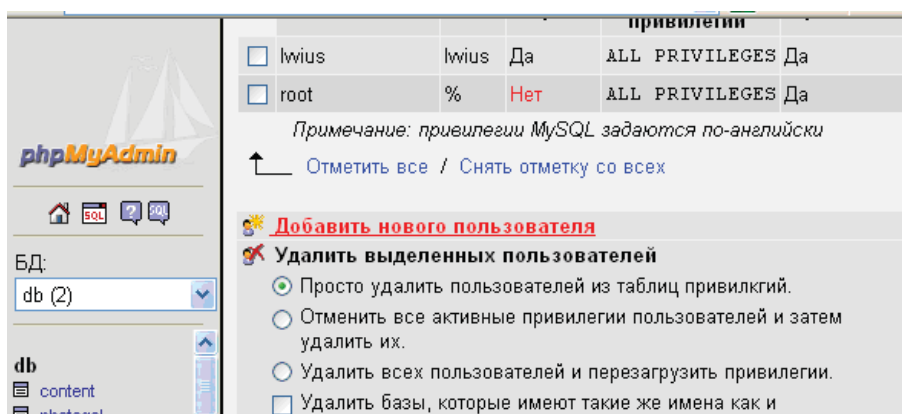


Рис. 4. Интерфейс Привилегии phpMyAdmin

На рис. 5—9 в качестве примера представлен порядок действий при создании пользователя **user_bd**.

Рис. 5. Пример заполнения формы **Информация логина**, глобальные привилегии не задавать — нажать кнопку **Пошел**

Рис. 6. Пример выбора базы данных для назначения привилегий

Рис. 7. Пример заполнения формы привилегий для базы данных **db**

Рис. 8. Пример создания пользователя **user_db** с пользовательскими привилегиями

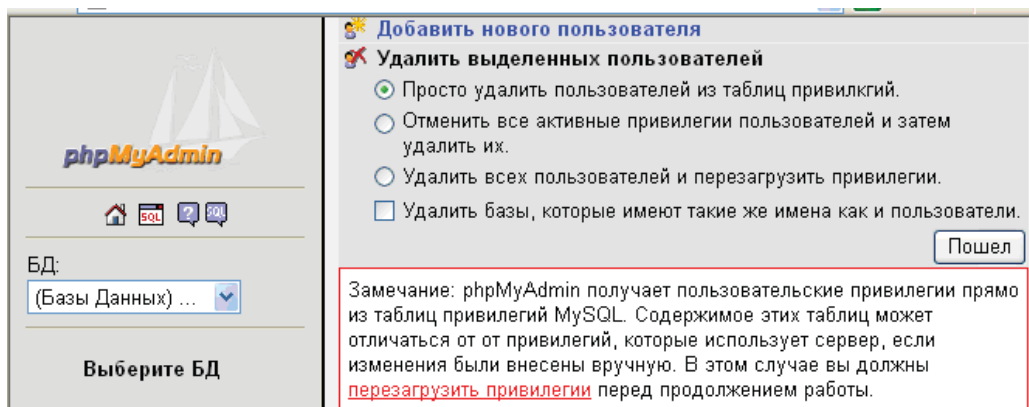


Рис. 9. Предупреждение о необходимости перезагрузки привилегий, чтобы они вступили в силу

2. В PHP-скрипте административного интерфейса, созданного в предыдущей работе изменить параметры подключения к MySQL так, чтобы подключение выполнялось не от имени пользователя **root**, а от имени пользователя **user** (изменения должны быть в подключаемом файле **config.php**). Проверить работу скрипта и убедиться, что действия, связанные с операцией записи новых данных, не выполняются. Изменить пользователя **user** на **admin** убедиться, что теперь все административные действия с базой данных выполняются.
3. При помощи phpMyAdmin создать новую базу данных, например, с именем **bd_2** (рис. 10).

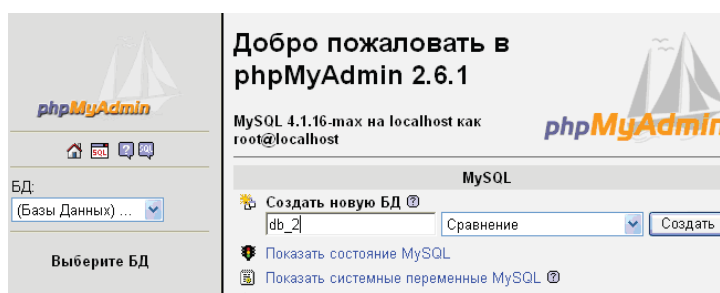


Рис. 10. Форма для создания новой базы данных в главном меню MySQL

4. В PHP-скрипте административного интерфейса изменить параметры подключения к MySQL так, чтобы подключение выполнялось к созданной базе данных от имени пользователя **user** (изменения должны быть в подключаемом файле **config.php**). Проверить работу скрипта и убедиться, что подключение не выполняется. Изменить пользователя **user** на **root** убедиться, что теперь подключение выполняется. Вернуть конфигурационные данные для подключения к базе библиотеки с пользователем **admin**.

3. Ограничение доступа в административный каталог средствами базовой аутентификации Web-сервера

При помощи утилиты **htpasswd.exe** создать файл **.htpasswd** хеш-функцией пароля пользователя **admin** (имя пользователя и пароль в общем случае никак не связаны с пользователями MySQL, но рекомендуется использовать такие же данные учетной записи). Для этого требуется запустить утилиту **htpasswd.exe** в командном режиме (cmd) с ключами, приведенными на рис. 11.

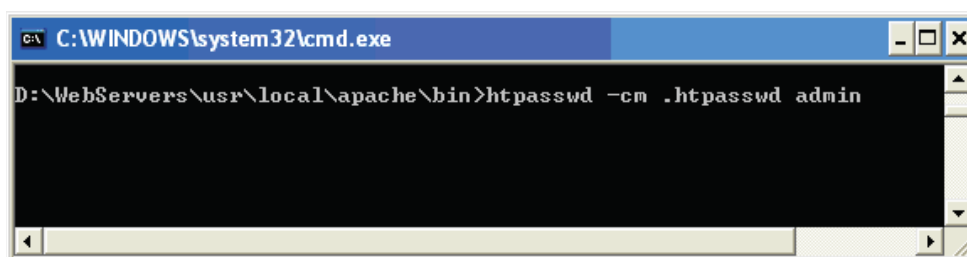


Рис. 11. Путь к утилите htpasswd.exe и ключи ее запуска

1. Для того чтобы не писать длинную строку пути к файлу `htpasswd.exe` рекомендуется открыть требуемый каталог в **Проводнике** (рис. 12) и перетащить файл `htpasswd.exe` в окно программы **cmd** (см. рис. 11). Далее дописать требуемые ключи и выполнить команду. На рис. 13. приведен пример содержания созданного текстового файла с паролем.

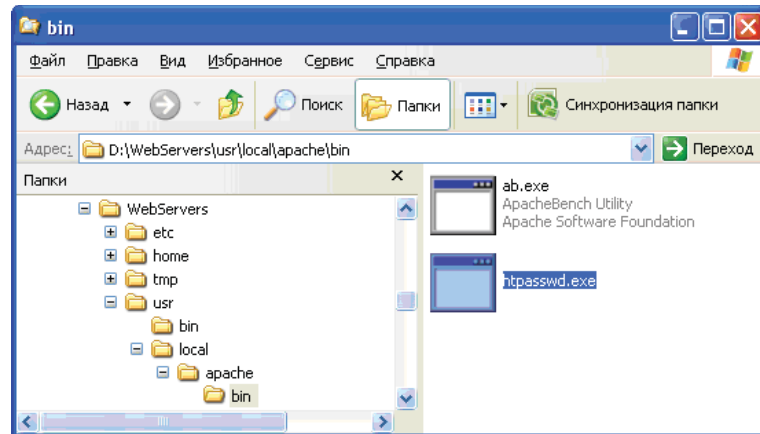


Рис. 12. Окно **Проводника** с файлом `htpasswd.exe`

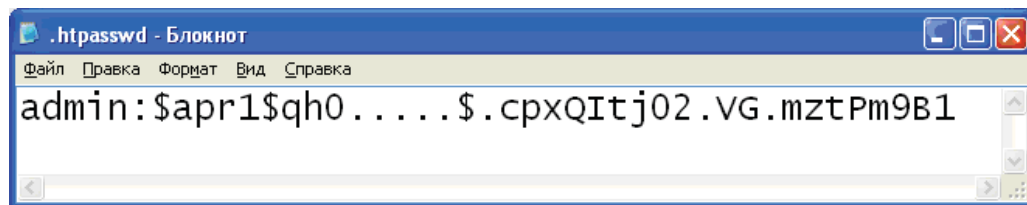


Рис. 13. Пример содержания текстового файла `.htpasswd`

2. Если при создании файла `.htpasswd` не указан путь, то он создается в том каталоге, из которого была выполнена команда запуска утилиты `htpasswd` (создается в каталоге, перед которым стоит символ курсора `>`. В примере рис. 11 — это каталог `bin`).
3. Перенести созданный текстовый файл `.htpasswd` с хеш-функцией пароля в корневой каталог сайта — **proba/**.
4. В каталоге **www/admin/** создать файл **.htaccess** при помощи блокнота, в котором прописать соответствующие директивы:
 - **AuthType** — тип используемой аутентификации: **Basic**.
 - **AuthName** — имя области действия аутентификации, текст, помогающий посетителю понять, куда он пытается получить доступ, например: «Только для администратора».
 - **AuthUserFile** — путь к файлу с паролями (`.htpasswd`), который должен быть прописан от корня Web-сервера, например: `AuthUserFile \home\proba\.htpasswd`.
 - **AuthGroupFile** — путь к файлу групп, если он существует.
 - **Require** — одно или несколько требований, которые должны быть выполнены для получения доступа к закрытой области (**require valid-user** — разрешен доступ всем прошедшим проверку).

На рис. 14 представлен пример текста файла `.htaccess`.

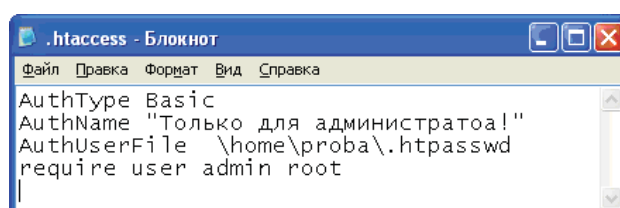


Рис. 14. Пример текста файла `.htaccess`

Обратите внимание, чтобы в свойствах папки (доступных в программах **Проводник** или **Мой компьютер**) было отключено скрывание расширения для зарегистрированных типов файлов (рис. 15), так как файл .htaccess не должен иметь расширения.

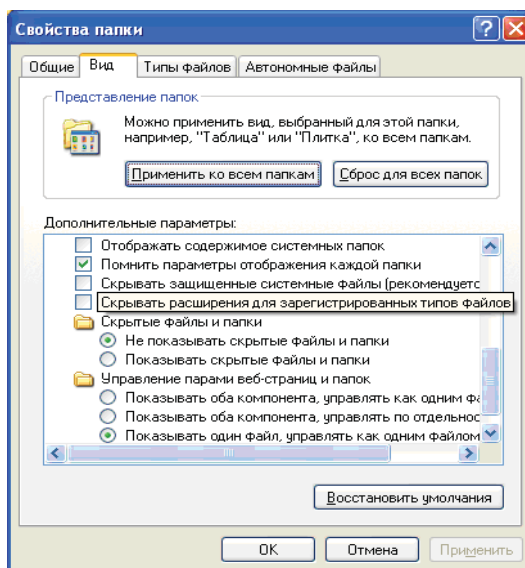


Рис. 15. Вкладка **Вид** в свойствах проводника

Для правильного создания файла .htaccess (без расширения .txt) при его сохранении в блокноте требуется выбрать **Тип файла** «Все файлы» (рис. 16).

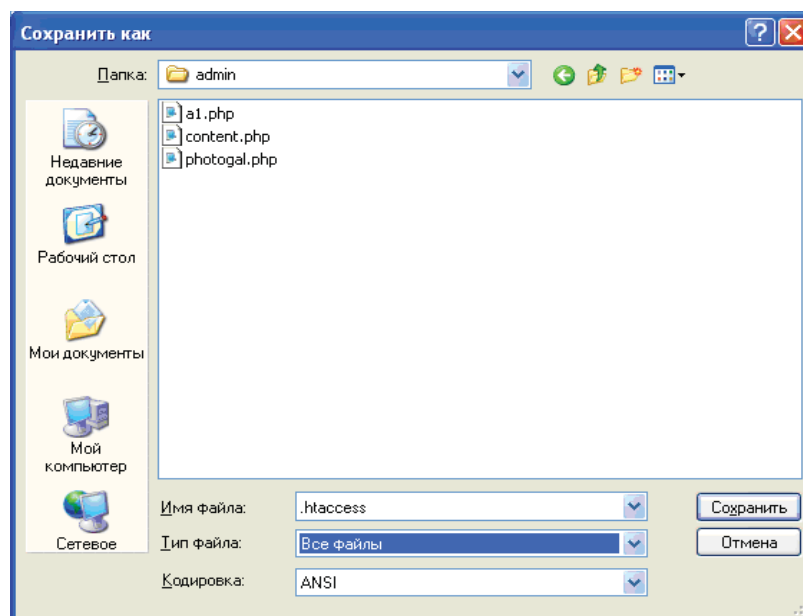


Рис. 16. Окно сохранения файла в блокноте

5. Проверить действие аутентификации при доступе в каталог `www / admin` (рис. 17).

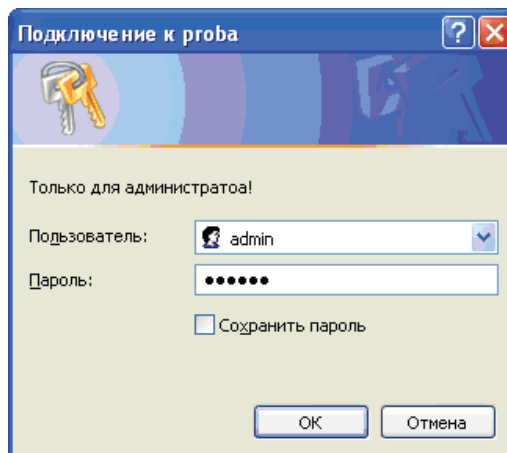


Рис. 17. Окно запроса имени пользователя и пароля при доступе в защищаемую область

4. Задание на самостоятельную работу

Добавить в число пользователей, которым разрешен доступ в каталог **www/admin/**, нового пользователя admin 2 с паролем 1234 и проверить работу системы его аутентификации.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задание 1-4) выполнена правильно. Задание 4 может проверяться отсрочено.выполнение данного задания является критерием получения оценки «отлично» за практическую работу.

Оценка «4» - Практическая работа (задание 1-3) выполнена. Допущенные ошибки студент может исправить самостоятельно при указании на них преподавателя

Оценка «3» - Практическая работа (задание 1-2) выполнена правильно. Допущенные ошибки студент может исправить, после значительного времени выделенного на доработку. Ориентируется в решении при изменении условий преподавателем.

Оценка «2» - Выполнено менее 2 заданий

Работа № 34.

ТЕМА: Совместное использование web- форм и базы данных

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка программирования PHP, web-форм и базы данных сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (вначале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

задание 1. Создать форму для ввода данных в БД в соответствии со своим вариантом:

1. Цветы. Организовать форму для добавления информации о цветке: наименование цветка, изображение цветка, частота цветения - список, способ размножения – список, температура - два числа, описание цветка.
2. Детские игрушки. Организовать форму для добавления информации о детских игрушках: наименование игрушки, Фирма производитель – из списка, две картинки, вид игрушки {для мальчиков, для девочек, для всех}, возраст – два числа.
3. Актеры. Организовать форму для добавления информации об актере: ФИО актера, год рождения, перечень фильмов с его участием, пол {мужской, женский}, жанры – список с множественным выбором, фото, официальный сайт.
4. Писатели. Организовать форму для добавления информации о писателях: ФИО писателя, год рождения, жанры – множественный список, книги – в виде архивных файлов, страна в которой родился – выпадающий список, e-mail адрес.

5. Программы. Организовать форму для добавления информации о программах: название программы, архив с программой, тип программы – из списка, условия распространения - радио кнопки, операционная система – множественный список.
6. Операционные системы. Организовать форму для добавления информации об операционных системах: название, фирма производитель, версия, логотип – файл, требования к оперативной памяти – минимум, объем требуемой оперативной памяти, ссылка на сайт.
7. Мебель. Организовать форму для добавления информации о мебели: название, вид мебели – выпадающий список, коллекция – выпадающий список, фотография, цветовые оттенки – множественный список, ссылка на сайт фирмы производителя.
8. Недвижимость. Организовать форму для добавления информации об недвижимости: вид недвижимости – радио кнопка, этажность, площадь, стоимость, местонахождение, файл с фотографией, ссылка на место на карту.
9. Бытовая техника. Организовать форму для добавления информации о бытовой технике, название, фотография, фирма производитель – список, характеристики техники, год производства.
10. Книги. Организовать форму для добавления информации о книге: название, автор, год издательства, число страниц, жанр – из списка, языки – множественный список, архив с текстом книги, содержание книги.

задание 2. Для разработанной формы в соответствии с заданием 1, создать БД для хранения введенных данных (структуру БД, название таблиц, атрибуты, типы данных продумать самостоятельно)

задание 3. Создать файл обработчик `brabotka.php` который содержит скрипт, для обработки созданной формы:

1.1.Соединяется с БД;

1.2.выполняет ввод данных в соответствующие поля нужной таблицы.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Не переходите к выполнению следующего задания, пока не убедитесь, что задача решена правильно (результат соответствует условию задачи).

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задание 1-3.2) выполнена правильно.

Оценка «4» - Практическая работа (задание 1-3.2) выполнена. Допущенные ошибки студент может исправить самостоятельно при указании на них преподавателя

Оценка «3» - Практическая работа (задание 1-3.2) выполнена правильно. Допущенные ошибки студент может исправить, после значительного времени выделенного на доработку. Ориентируется в решении при изменении условий преподавателем.

Оценка «2» - Выполнено менее 3 заданий

Работа № 35.

ТЕМА: Разработка формы для ввода /вывода данных в БД

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка программирования PHP, web-форм и базы данных сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (в начале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

задание 1. Создать форму для выполнения запроса на выборку данных из БД в соответствии со своим вариантом (практическая работа №36), предусмотреть различные варианты запросов (по разным полям).

задание 2. Создать файл обработчик `brabotka.php`, который содержит скрипт, для обработки формы запроса на выборку.

2.1. Соединяется с БД;

2.2. выполняет ввод данных в соответствующие поля нужной таблицы.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Не переходите к выполнению следующего задания, пока не убедитесь, что задача решена правильно (результат соответствует условию задачи).

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задание 1-2.2) выполнена правильно, разработано 4-5 форм, для осуществления запроса.

Оценка «4» - Практическая работа (задание 1-2.1) выполнена. разработано 3-4 формы, для осуществления запроса. Допущенные ошибки студент может исправить самостоятельно при указании на них преподавателя

Оценка «3» - Практическая работа (задание 1-2.1) выполнена правильно.разработано 2-3 форм, для осуществления запроса. Допущенные ошибки студент может исправить, после значительного времени выделенного на доработку. Ориентируется в решении при изменении условий преподавателем.

Оценка «2» - Выполнено менее 2 заданий

Работа № 36.

ТЕМА: Ввод данных в БД, вывод информации из БД.

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка программирования PHP, web-форм и базы данных сохраняется в личной директории.

Условия контроля: аудиторно на занятии студенту выдается билет, содержащий два вопроса из перечня вопросов к зачету и задача..

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: проверка правильности ответа на вопросы к зачету, отчет по выполнению практических работ

ЗАДАНИЯ:

Вопросы

1. Глобальные компьютерные сети: основные понятия, принципы функционирования. Каталоги ресурсов. Поисковые системы.
2. Язык гипертекстовой разметки страниц HTML: общая структура документа, абзацы, цвета, ссылки.
3. Язык гипертекстовой разметки страниц HTML: списки, графика (графические форматы, графический объект как ссылка), таблицы, фреймы.
4. Общие подходы к дизайну сайта. Разработка макета страницы.
5. Язык гипертекстовой разметки страниц HTML: формы.
6. Использование стиля при оформлении сайта. Возможности CSS.
7. CSS. Свойства текста. Свойства цвета и фона. Свойства шрифта. Свойства блоков.
8. CSS. Свойства списков. Классы. Псевдоклассы.
9. Хостинг. Бесплатный хостинг. FTP. Размещение Интернет-ресурса на сервере провайдера. Регистрация Интернет-ресурса в каталогах и поисковых системах.
10. Преимущества и ограничения программ, работающих на стороне клиента. Язык JavaScript: основы синтаксиса.
11. Объектная модель HTML страницы.
12. Событийная модель DHTML: связывание событий с кодом, всплытие событий, объект Event.
13. Применение DHTML.
14. XML. MathML.
15. Введение в программирование на стороне сервера на примере PHP. Принцип работы.
16. Синтаксис языка программирования PHP.
17. Переменные. Константы. Операторы в PHP. Циклы. Массивы. Работа со строками.
18. Функции в PHP. Встроенные функции.

19. Работа с датой и временем в PHP.
20. Методы передачи параметров между страницами (GET, POST). Обработка действий пользователя при помощи форм.
21. Принципы хранения информации в базах данных MySQL. Архитектура базы данных MySQL (таблицы, связи, триггеры).
22. Механизм работы с базами данных — PhpMyAdmin.
23. Подключение к базе данных из PHP файла. Вывод данных на PHP-страницу, попавших в выборку по SQL запросу. Передача параметров в запрос.
24. Принципы проектирования страниц. Разделение информации по таблицам в базе данных. Вывод группы данных, сортировка данных.
25. Создание HTML-страниц средствами PHP.

Задачи:

в качестве зачетной задачи, студенту предлагается продемонстрировать любую задачу по выбору преподавателя из практических работ. В ходе демонстрации студент должен показать результат работы, продемонстрировать код программы, а также уметь отвечать на дополнительные вопросы по ходу выполнения задачи (назначение тегов, операторов), уметь изменять код решения в зависимости от требования преподавателя.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задание 1-2.2) выполнена правильно, разработано 4-5 форм, для осуществления запроса.

Оценка «4» - Практическая работа (задание 1-2.1) выполнена.разработано 3-4 формы, для осуществления запроса. Допущенные ошибки студент может исправить самостоятельно при указании на них преподавателя

Оценка «3» - Практическая работа (задание 1-2.1) выполнена правильно. разработано 2-3 форм, для осуществления запроса. Допущенные ошибки студент может исправить, после значительного времени выделенного на доработку. Ориентируется в решении при изменении условий преподавателем.

Оценка «2» - Выполнено менее 2 заданий

Работа № 37.

ТЕМА: Проектирование клиентской части

ЦЕЛЬ: Выполнить первоначальный этап разработки web - проекта.

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка гипертекстовой разметки HTMLи сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Ответы на вопросы даются письменно в тетради.

ФОРМА КОНТРОЛЯ: Экспертиза качества выполнения практической работы, Проверка правильности письменных ответов на вопросы.

ЗАДАНИЯ:

1. Составить техническое задание на проектирование web - приложения (техническое задание оформить в виде документа ТЗ.docx и сохранить в свою директорию в папку WEB - программирование);
2. Спроектировать HTML-формы для регистрации и авторизации пользователей сайта.

3. ЗАДАНИЕ К ЗАЩИТЕ ПРАКТИЧЕСКОЙ РАБОТЫ

Письменно ответить на контрольные вопросы:

1. перечислите современные системы коллективной человеческой деятельности;
2. сформулируйте понятие "Клиент-серверная технология";
3. какие существуют подходы для реализации клиент-серверной технологии?
4. что такое web - технология?
5. опишите архитектуру WEB- технологий.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

ДЛЯ ВЫПОЛНЕНИЯ 1-ГО ЗАДАНИЯ

Ознакомиться с материалом учебника:

Серых Ю.А. Современный веб-дизайн. Рисуем сайт который продает. (электронная книга)
<http://www.litmir.net/bd/?b=205941> (дата обращения 10.09.14) стр.43 - 47.

ДЛЯ ВЫПОЛНЕНИЯ 2-ГО ЗАДАНИЯ

Ознакомиться с лекционным материалом, материалом учебника:

Ташков Петр Веб-мастеринг на 100 %: HTML, CSS, JavaScript, PHP, CMS, AJAX, раскрутка.
СПб.: Питер, 2010. стр. 96 - 109.

Стивен Хольцер HTML 5 за 10 минут. М.: ИД Вильямс, 2011. стр. 73 - 97.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - студент выполнил все практические задания, указанные в ходе работы, письменно ответил на вопросы,;

Оценка «4» - студент выполнил все практические задания, указанные в ходе работы, однако теоретическую часть работы в тетради выполнить не успел, но ориентируется в вопросах и может ответить на все вопросы устно;

Оценка «3» - студент выполнил одно задание или два, но с ошибками. Ориентируется в теоретической части работы и может устно отвечать на вопросы;

Оценка «2» - студент справился с менее чем на 50 % работы, не может отвечать на большую часть теоретических вопросов

Работа № 38.

ТЕМА: Проектирование структуры web-приложения

ЦЕЛЬ: Разработать структуру web - приложения в соответствии с выбранной темой проекта.

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка гипертекстовой разметки HTMLи сохраняется в личной директории.

Условия контроля: отсрочено, вначале урока при проведении следующей практической работы.

Работа выполняется в ходе занятия с возможностью доработки самостоятельно (внеаудиторно).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

1. разработать физическую структуру сайта;

2. разработать логическую структуру сайта
3. Письменно ответить на контрольные вопросы:
 1. Перечислите и опишите назначение основных видов web - систем;
 2. каково назначение и основные функции web - браузеров?
 3. каковы основные элементы пользовательского интерфейса web - браузера Opera?
 4. дайте характеристику web-редактора FrontPage?
 5. какова архитектура web - приложения?

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Разработка структуры сайта включает все, что касается его содержимого и информационной стратегии, определяющей, как должна быть организована информация, чтобы пользователи сайта могли найти ее легко и быстро. Создается ли сайт для новой компании или перепроектируется имеющийся, - это неважно: логическая структура необходима любому веб-сайту. Заказчика, прежде всего, заботит внешнее оформление и впечатление, производимое сайтом, но разработка надежной, хорошо продуманной схемы заложит основу для всего остального, включая и визуальный дизайн.

Без хорошего, информативного содержания никакой сайт не станет неотразимым. Контент и структура сайта взаимосвязаны: нельзя создать одно, не продумав другое. Разбивка и классификация страниц напрямую определяются содержанием сайта, а способ, которым организован контент, определяет процесс структурирования.

При разработке сайта нельзя забывать основное - сеть создана для предоставления информации, а также быстрого и легкого ее поиска. Независимо от конкретной задачи пользователям необходимо, чтобы они были ясно и просто направлены к разыскиваемым данным. Хороший информационный дизайн снабжает пользователей «дорожными знаками», которые помогают им ориентироваться и не потеряться.

Структурирование любого сайта - прямо с нуля или при редизайне - включает рассмотрение трех точек зрения: контента, сайта и страницы. Контент, или содержимое - это то, что дает жизнь сайту. Усовершенствование структуры как с точки зрения сайта (схематизация сайта), так и с точки зрения страницы (структурирование) позволяют информационному дизайнеру представить организованное содержимое таким способом, который будет логически понятен пользователю. Структурирование сайта на бумаге перед началом визуального проектирования - очень важный этап правильной адресации к целевой аудитории. Эффективный информационный дизайн незаметен для пользователя. Однако если он плохо продуман, пользователи сразу почувствуют неудобство и даже раздражение.

КРИТЕРИЙ ОЦЕНКИ:

- Оценка «5» - студент выполнил все практические задания, указанные в ходе работы, письменно ответил на вопросы,;
- Оценка «4» - студент выполнил все практические задания, указанные в ходе работы, однако теоретическую часть работы в тетради выполнить не успел, но ориентируется в вопросах и может ответить на все вопросы устно;
- Оценка «3» - студент выполнил одно задание или два, но с ошибками. Ориентируется в теоретической части работы и может устно отвечать на вопросы;
- Оценка «2» - студент справился с менее чем на 50 % работы, не может отвечать на большую часть теоретических вопросов

Работа № 39.

ТЕМА: Разработка клиентской части web - приложения

ЦЕЛЬ: Выполнить первоначальный этап разработки web - проекта.

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка гипертекстовой разметки HTMLи сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: отсрочено, вначале урока при проведении следующей практической работы.

Работа выполняется в ходе занятия с возможностью доработки самостоятельно (внеаудиторно).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

1) Реализовать систему навигации по сайту:

- а) Главное меню сайта;
- б) Динамическое меню подраздела;

2) Письменно ответить на контрольные вопросы:

1. какой тег позволяет создать гиперссылку на языке HTML, какие атрибуты имеет данный тэг и за что отвечают данные атрибуты ;
2. какие атрибуты отвечают за то, чтобы форма реагировала на нажатие клавиши ввода?;

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

ДЛЯ ВЫПОЛНЕНИЯ 1-ГО ЗАДАНИЯ

Повторить теоретический и практический материал по темам прошлого года:

- Разработка структуры титульной страницы тематического сайта
- Разработка системы навигации тематического сайта
- Подготовка изображения для навигационной карты
- Разработка навигационной карты для системы навигации
- Разработка фреймовой структуры web - ресурса

Ознакомиться материалами учебников:

Серых Ю.А. Современный веб-дизайн. Рисуем сайт который продает. (электронная книга)
<http://www.litmir.net/bd/?b=205941> (дата обращения 10.09.14) стр.43 - 47.

ДЛЯ ВЫПОЛНЕНИЯ 2-ГО ЗАДАНИЯ

Ознакомиться с лекционным материалом, материалом представленным в презентации ФОРМЫ.pptx, также материалом учебника:

Ташков Петр Веб-мастеринг на 100 %: HTML, CSS, JavaScript, PHP, CMS, AJAX, раскрутка.
СПб.: Питер, 2010. стр. 96 - 109.

Стивен Хольцер HTML 5 за 10 минут. М.: ИД Вильямс, 2011. стр. 73 - 97.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - студент выполнил все практические задания, указанные в ходе работы, письменно ответил на вопросы,;

Оценка «4» - студент выполнил все практические задания, указанные в ходе работы, однако теоретическую часть работы в тетради выполнить не успел, но ориентируется в вопросах и может ответить на все вопросы устно;

Оценка «3» - студент выполнил одно задание или два, но с ошибками. Ориентируется в теоретической части работы и может устно отвечать на вопросы;

Оценка «2» - студент справился с менее чем на 50 % работы, не может отвечать на большую часть теоретических вопросов

Работа № 40.

ТЕМА: Разработка системы навигации

ЦЕЛЬ: Реализовать систему навигации по сайту средствами HTML.

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка гипертекстовой разметки HTMLи сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: отсрочено, вначале урока при проведении следующей практической работы.

Работа выполняется в ходе занятия с возможностью доработки самостоятельно (внеаудиторно).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

1) Реализовать систему навигации по сайту:

- а) навигационную карту (текстовая или графическая);
- б) перекрестные ссылки внутри страниц, длинна которых превышает один экран;

2) Письменно ответить на контрольные вопросы:

- 1. Как создать ссылку на ФРАГМЕНТ внешнего документа?;
- 2. Какие атрибуты определяют в каком подокне фрейма будет открыт документ по гиперссылке?

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Ознакомиться с материалом учебника:

Борисенко А.А. Web - дизайн. Просто как дважды два. М.: Эксмо, 2008. стр. 99-121.

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - студент выполнил все практические задания, указанные в ходе работы, письменно ответил на вопросы,;

Оценка «4» - студент выполнил все практические задания, указанные в ходе работы, однако теоретическую часть работы в тетради выполнить не успел, но ориентируется в вопросах и может ответить на все вопросы устно;

Оценка «3» - студент выполнил одно задание или два, но с ошибками. Ориентируется в теоретической части работы и может устно отвечать на вопросы;

Оценка «2» - студент справился с менее чем на 50 % работы, не может отвечать на большую часть теоретических вопросов

Работа № 41.

ТЕМА: Обработка контента, ввод и верстка

ЦЕЛЬ: Реализовать наполнение информации разрабатываемого веб-проекта

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка гипертекстовой разметки HTMLи сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: отсрочено, вначале урока при проведении следующей практической работы.

Работа выполняется в ходе занятия с возможностью доработки самостоятельно (внеаудиторно).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

1) заполнить информацией страницы сайта:

- а) 1-й уровень, т.е. все страницы на которые имеются ссылки в главном меню;
- б) 2-й уровень, т.е. все страницы на которые имеются ссылки в динамическом подменю;;

2) Письменно ответить на контрольные вопросы:

- 1. Что такое система управления контентом?;
- 2. Продолжите фразу:

HTML – это

XML – это

DHTML – это

JavaScript – это

PHP – это

MySQL – это

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

- Ознакомиться с материалом учебника
- выполнить ввод текста с клавиатуры, на основании подготовленного дома материала в соответствии с темой проекта.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - студент выполнил все практические задания, указанные в ходе работы, письменно ответил на вопросы,;

Оценка «4» - студент выполнил все практические задания, указанные в ходе работы, однако теоретическую часть работы в тетради выполнить не успел, но ориентируется в вопросах и может ответить на все вопросы устно;

Оценка «3» - студент выполнил одно задание или два, но с ошибками. Ориентируется в теоретической части работы и может устно отвечать на вопросы;

Оценка «2» - студент справился с менее чем на 50 % работы, не может отвечать на большую часть теоретических вопросов

Работа № 42.

ТЕМА: Проектирование стилистических решений

ЦЕЛЬ: Определить какие стилистические решения вы будете использовать при разработке веб-ресурса

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка гипертекстовой разметки HTMLи сохраняется в личной директории.

Условия контроля: отсрочено, вначале урока при проведении следующей практической работы.

Работа выполняется в ходе занятия с возможностью доработки самостоятельно (внеаудиторно).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

1. Заполнить таблицу:

ОБЪЕКТЫ	ПАРАМЕТРЫ
Текст заголовков	
Текст абзацев	
Таблица	
Гиперссылка	
Страница	
Рисунок	

2. Письменно ответить на контрольные вопросы:

1. Что такое система управления контентом?;
2. Какие СМС вы знаете?
3. Проведите анализ возможностей различных СМС, в чем их плюсы и недостатки, сравните и сделайте выводы.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - студент правильно выполнил все практические задания (1-2).

Оценка «4» - студент выполнил практические задания (1-2), однако, имеются незначительные ошибки, которые студент исправил самостоятельно;

Оценка «3» - студент выполнил практические задания (1-2), однако, имеются ошибки, которые студент может исправить при указании на них преподавателя с использованием методического материала (МУ к практическим, справочник, учебник);

Оценка «2» - практическая работа выполнена с большим количеством замечаний, которые студент не может исправить самостоятельно при указании на них преподавателя.

Работа № 43.

ТЕМА: Подключение и создание CSS стилей

ЦЕЛЬ: Подключить каскадную таблицу стилей

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка гипертекстовой разметки HTML и таблицы стилей CSS, сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: отсрочено, в начале урока при проведении следующей практической работы.

Работа выполняется в ходе занятия с возможностью доработки самостоятельно (внеаудиторно).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

1. Создать файл styc.css с теми параметрами, которые были определены в практической работе №44;
2. Подключить этот файл к каждой странице вашего сайта
3. Дайте пояснение следующим фрагментам таблицы стилей:

Фрагмент	Пояснение
<pre>a:link { font-weight: bold; text-decoration: none; color: #3792b3; }</pre>	
<pre>ul.lst li a { display: block; padding: 7px; border-bottom: 1px solid #f3f2e8; }</pre>	
<pre>ul.lst li em { font-style: normal; font-weight: normal; color: #9c836e; }</pre>	
<pre>ul.lst li a { position: relative; display: block; padding: 7px; border-bottom: 1px solid #f3f2e8; }</pre>	

Фрагмент	Пояснение
<pre>ul.lst li em { position: absolute; top: 7px; right: 7px; font-style: normal; font-weight: normal; color: #9c836e; }</pre>	
<pre>ul.lst li a { position: relative; display: block; padding: 7px 50px 7px 7px; border-bottom: 1px solid #f3f2e8; }</pre>	

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Не переходите к выполнению следующего задания, пока не убедитесь, что задача решена правильно (результат соответствует условию задачи).

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задание 1,2, 3) выполнены правильно. Студент ориентируется в CSS коде, может самостоятельно изменить любые стилевые решения, добавлять новые. Задание №4 может проверяться отсрочено, вначале урока следующего занятия.

Оценка «4» - Практическая работа (задание 1, 2) выполнены правильно. Студент ориентируется в CSS коде, может самостоятельно изменить любые стилевые решения, добавлять новые.

Оценка «3» - Практическая работа (задание 1, 2) выполнены правильно. Студент ориентируется в CSS коде, однако для изменения имеющихся стилевых решений нуждается в подсказке (справочная литература, лекционный материал)

Оценка «2» - В практической работе выполнено менее 2 задач. Студент затрудняется в демонстрации возможности изменить стилевые решения при указании преподавателя.

Работа № 44.

ТЕМА: Проектирование серверной части

ЦЕЛЬ: Определить какие функции по обработке данных должен выполнять сервер. Разработать код программы на языке программирования для обработки данных полученных от клиента

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка гипертекстовой разметки HTMLи сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: отсрочено, вначале урока при проведении следующей практической работы.

Работа выполняется в ходе занятия с возможностью доработки самостоятельно (внеаудиторно).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

1. Определить, какие данные (имена переменных, типы переменных) будут получены от пользователя.
2. Для этих данных разработать формы для передачи данных серверу, а также обработчики форм на языке программирования.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Не переходите к выполнению следующего задания, пока не убедитесь, что задача решена правильно (результат соответствует условию задачи).

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задание 1, 2,) выполнены правильно. Студент ориентируется в HTML и CSS коде, может самостоятельно разрабатывать и добавлять формы.

Оценка «4» - Практическая работа (задание 1, 2) выполнены правильно. Студент ориентируется в HTML и CSS коде, может разрабатывать формы, но затрудняется в создании обработчика форм.

Оценка «3» - Практическая работа (задание 1, 2) выполнены правильно. Студент ориентируется в HTML и CSS коде, однако для изменения имеющихся форм нуждается в подсказке (справочная литература, лекционный материал)

Оценка «2» - В практической работе выполнена 1 задача. Студент затрудняется в демонстрации возможности изменить стилевые решения при указании преподавателя.

Работа № 45.

ТЕМА: Проектирование реляционной БД

ЦЕЛЬ: Описать объекты информации о которых будет храниться в базе данных. Свойства объектов в соответствии с полями форм

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере в текстовом (графическом) редакторе и сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (в начале урока).

Работа выполняется в ходе занятия с возможностью доработки самостоятельно (внеаудиторно).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

1. Определить набор сущностей достаточный для хранения информации в базе данных при решении поставленной задачи
2. Для каждой выделенной сущности перечислить набор атрибутов, который необходимо хранить в БД
3. Для каждой сущности, выделить первичные ключи
4. Спроектировать базу данных, опираясь на правило «Каждому информационному объекту отдельную таблицу»
5. Установить связи между таблицами

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Не переходите к выполнению следующего задания, пока не убедитесь, что задача решена правильно (результат соответствует условию задачи).

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задания 1-5) выполнены правильно.

Оценка «4» - Практическая работа (задание 1-5) выполнены правильно. Имеются ошибки при определении атрибутов или ключей, которые студент может исправить самостоятельно при указании на них преподавателя

Оценка «3» - Практическая работа (задание 1-4) выполнены правильно. Имеются ошибки при определении атрибутов, студент может исправить ошибки, только после значительного времени выделенного на доработку. Ориентируется в решении при изменении условий преподавателем.

Оценка «2» - В практической работе выполнено менее 2 задач. Студент испытывает затруднения и не может исправить ошибки даже при указании на них преподавателя.

Работа № 46.

ТЕМА: Разработка реляционной базы данных

ЦЕЛЬ: Разработать БД средствами RНРМуAdmin в соответствии с результатами выполнения практической работ №8,9

Работа выполняется на компьютере с использованием языка программирования РНР, сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (вначале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

Реализовать Таблицы базы данных в соответствии с результатами проектирования в практических работах №8,9.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа выполнена правильно.

Оценка «4» - Практическая работа выполнена правильно. Имеются небольшие ошибки при определении триггера, которые студент может исправить самостоятельно при указании на них преподавателя

Оценка «3» - Практическая работа выполнена. Имеются ошибки при определении триггера, студент может исправить ошибки, только после значительного времени выделенного на доработку. Ориентируется в решении при изменении условий преподавателем.

Оценка «2» - Задание не выполнено

Работа № 47.

ТЕМА: Ввод данных в БД

ЦЕЛЬ: Осуществить ввод данных в разработанные таблицы БД

Работа выполняется на компьютере с использованием языка программирования РНР, сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (вначале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

Ввести исходные данные в разработанные таблицы. В каждую таблицу не менее 10 записей.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа выполнена правильно.

Оценка «4» - Практическая работа выполнена правильно. Имеются небольшие ошибки, которые студент может исправить самостоятельно при указании на них преподавателя

Оценка «3» - Практическая работа выполнена. Имеются ошибки, студент может исправить ошибки, только после значительного времени выделенного на доработку. Ориентируется в решении при изменении условий преподавателем.

Оценка «2» - Задание не выполнено

Работа № 48.

ТЕМА: Обеспечение взаимодействия БД и web - приложения

ЦЕЛЬ: Протестировать правильность работы HTML форм и обработчиков форм для организации ввода данных из БД в окно программы браузера.

Работа выполняется на компьютере с использованием языка программирования РНР, сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (вначале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза подготовленных отчетов. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

1. Проверить корректность ввода данных в БД посредством разработанных форм.
2. Подготовить отчет по проведенной проверке (в текстовом редакторе)

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа выполнена правильно.

Оценка «4» - Практическая работа выполнена правильно. Имеются небольшие ошибки, которые студент может исправить самостоятельно при указании на них преподавателя

Оценка «3» - Практическая работа выполнена. Имеются ошибки, студент может исправить ошибки, только после значительного времени выделенного на доработку. Ориентируется в решении при изменении условий преподавателем.

Оценка «2» - Задание не выполнено

Работа № 49.

ТЕМА: Организация вывода данных из БД в окно программы браузера

ЦЕЛЬ: Осуществить ввод данных в разработанные таблицы БД

Работа выполняется на компьютере с использованием языка программирования PHP, сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (в начале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

1. Проверьте корректность ввода данных в БД посредством разработанных форм.
2. Подключитесь к созданной БД;
3. Осуществите выборку данных из нескольких таблиц (продемонстрируйте не менее 3 примеров);
4. Установите параметры выборки (количество записей, сортировка)

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Не переходите к выполнению следующего задания, пока не убедитесь, что задача решена правильно (результат соответствует условию задачи).

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задание 1-4) выполнена правильно.

Оценка «4» - Практическая работа (задание 1-4) выполнена. Допущенные ошибки студент может исправить самостоятельно при указании на них преподавателя

Оценка «3» - Практическая работа (задание 1-3) выполнена правильно. Допущенные ошибки студент может исправить, после значительного времени выделенного на доработку. Ориентируется в решении при изменении условий преподавателем.

Оценка «2» - Выполнено менее 3 заданий

Работа № 50.

ТЕМА: Разработка web-форм (формы HTML)

ЦЕЛЬ: Разработать web-формы для реализации поставленной задачи.

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка программирования PHP, web-форм и базы данных сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (вначале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

1. Создать форму для выполнения запроса на выборку данных из БД в соответствии со своим вариантом (практическая работа №36), предусмотреть различные варианты запросов (по разным полям).

2. Создать файл обработчик `obrabotka.php`, который содержит скрипт, для обработки формы запроса на выборку.

2.1. Соединяется с БД;

2.2. Выполняет ввод данных в соответствующие поля нужной таблицы.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Не переходите к выполнению следующего задания, пока не убедитесь, что задача решена правильно (результат соответствует условию задачи).

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задание 1-2.2) выполнена правильно, разработано 4-5 форм, для осуществления запроса.

Оценка «4» - Практическая работа (задание 1-2.1) выполнена.разработано 3-4 формы, для осуществления запроса. Допущенные ошибки студент может исправить самостоятельно при указании на них преподавателя

Оценка «3» - Практическая работа (задание 1-2.1) выполнена правильно.разработано 2-3 форм, для осуществления запроса. Допущенные ошибки студент может исправить, после значительного времени выделенного на доработку. Ориентируется в решении при изменении условий преподавателем.

Оценка «2» - Выполнено менее 2 заданий

Работа № 51.

ТЕМА: Ввод данных в БД с помощью передачи простой формы

ЦЕЛЬ: Реализовать наполнение информации разрабатываемой базы данных

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка программирования PHP, web-форм и базы данных сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (в начале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

Осуществить ввод данных в БД, используя формы, разработанные в работе 52.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа выполнена правильно.

Оценка «4» - Практическая работа выполнена правильно. Имеются небольшие ошибки, которые студент может исправить самостоятельно при указании на них преподавателя

Оценка «3» - Практическая работа выполнена. Имеются ошибки, студент может исправить ошибки, только после значительного времени выделенного на доработку. Ориентируется в решении при изменении условий преподавателем.

Работа № 52.

ТЕМА: Извлечение содержимого таблиц

ЦЕЛЬ: Реализовать выборку данных из нескольких таблиц.

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка программирования PHP, web-форм и базы данных сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (вначале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

1. В БД создать три таблицы: users (пользователи), topics (темы) и posts (сообщения). Структуру таблиц, связи определить самостоятельно.
2. Заполните созданные таблицы (не менее 5 записей в каждой);
3. Создайте файл Index.php в котором:
 - 3.1. Подключитесь к созданной БД;
 - 3.2. Осуществите выборку данных из каждой таблицы;

Замечание: структуру запроса, условия продумайте самостоятельно

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задание 1-3.2) выполнена правильно.

Оценка «4» - Практическая работа (задание 1-3.2) выполнена. Допущенные ошибки студент может исправить самостоятельно при указании на них преподавателя

Оценка «3» - Практическая работа (задание 1-3.2) выполнена правильно. Допущенные ошибки студент может исправить, после значительного времени выделенного на доработку. Ориентируется в решении при изменении условий преподавателем.

Оценка «2» - Выполнено менее 3 заданий

Работа № 53.

ТЕМА: Определение количества записей в результирующей таблицы

ЦЕЛЬ: Устанавливать параметры выборки. Осуществить выборку данных из нескольких таблиц.

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка программирования PHP, web-форм и базы данных, сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (в начале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

1. Открыть базу банных Forum.
2. Создайте файл Index.php в котором:
 - 2.1. Подключитесь к созданной БД;
 - 2.2. Осуществите выборку данных из нескольких таблиц (продемонстрируйте не менее 3 примеров);
 - 2.3. Установите параметры выборки (количество записей, сортировка)

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

Коротко зафиксируйте в тетради положения, теги (скрипты, команды, алгоритмы...), необходимые для выполнения задания и отсутствующие в существующем лекционном материале (эта информация может еще не раз пригодиться вам в дальнейшем, а «записывание» поможет вам лучше и быстрее запомнить материал). Не забудьте указать источник информации, чтобы облегчить поиск информации, необходимой для выполнения последующих заданий.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задание 1-2.3) выполнена правильно.

Оценка «4» - Практическая работа (задание 1-2.3) выполнена. Допущенные ошибки студент может исправить самостоятельно при указании на них преподавателя

Оценка «3» - Практическая работа (задание 1-2.2) выполнена правильно. Допущенные ошибки студент может исправить, после значительного времени выделенного на доработку. Ориентируется в решении при изменении условий преподавателем.

Оценка «2» - Выполнено менее 2 заданий

Работа № 54.

ТЕМА: Система регистрации

ЦЕЛЬ: Разработать и создать систему регистрации разрабатываемого web-ресурса
ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка гипертекстовой разметки HTML, PHP, web- форм и базы данных, сохраняется в личной директории.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (в начале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

1. Разработать проект системы регистрации:

1.1. Макет формы

1.2. Таблица для БД

2. Создать таблицу БД.

3. Подключить БД к Web-приложению

4. Предусмотреть возможность изменения и удаления полей и записей

5. Проверить работоспособность ограничений значений полей, уникальности и др., предусмотренные при задании структуры базы данных

6. Проверить работоспособность ссылочной целостности, удаляя, изменяя и вставляя данные

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задание 1, 2, 3, 4, 5, 6) выполнена правильно.

Оценка «4» - Практическая работа (задание (задание 1, 2, 3, 4, 5) выполнена правильно. Имеются ошибки при определении триггера, которые студент может исправить самостоятельно при указании на них преподавателя

Оценка «3» - Практическая работа (задание (задание 1, 2, 3, 4) выполнена правильно. Имеются ошибки при определении триггера, студент может исправить ошибки, только после значительного времени выделенного на доработку. Ориентируется в решении при изменении условий преподавателем.

Оценка «2» - Задания с 4 по 6 не выполнены

Работа № 55.

ТЕМА: Система авторизации

ЦЕЛЬ: Разработать и создать систему регистрации разрабатываемого web-ресурса

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка гипертекстовой разметки HTML, PHP, web- форм и базы данных, сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (в начале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

1. Разработать проект системы авторизации:
 - 1.1. Макет формы
 - 1.2. Выборка данных из БД
2. Создать форму
3. Создать систему сложных запросов к БД, предусмотрев возможность орфографических и стилистических ошибок, исключение повторов
4. Предусмотреть возможность изменения и удаления полей и записей
5. Проверить работоспособность ограничений значений полей, уникальности и др., предусмотренные при задании структуры базы данных
6. Проверить работоспособность ссылочной целостности, удаляя, изменяя и вставляя данные

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задание 1, 2, 3, 4, 5, 6) выполнена правильно.

Оценка «4» - Практическая работа (задание (задание 1, 2, 3, 4, 5) выполнена правильно. Имеются ошибки при определении триггера, которые студент может исправить самостоятельно при указании на них преподавателя

Оценка «3» - Практическая работа (задание (задание 1, 2, 3, 4) выполнена правильно. Имеются ошибки при определении триггера, студент может исправить ошибки, только после значительного времени выделенного на доработку. Ориентируется в решении при изменении условий преподавателем.

Оценка «2» - Задания с 4 по 6 не выполнены

Работа № 56.

ТЕМА: Базовая НТТР - авторизация

ЦЕЛЬ: Разработать систему базовой НТТР-авторизации

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка гипертекстовой разметки HTML, PHP, web- форм и базы данных, сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (в начале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

В разработанную систему регистрации-авторизации внести следующие дополнения:

1. Защите от «робота»;
2. Разграничение прав разных категорий пользователей;
3. Возможность смены пароля;
4. Возможность восстановления пароля.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам, представленным ниже.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задание 1-4) выполнена правильно.

Оценка «4» - Практическая работа (задание 1-4) выполнена. Допущенные ошибки студент может исправить самостоятельно при указании на них преподавателя

Оценка «3» - Практическая работа (задание 1-3) выполнена правильно. Допущенные ошибки студент может исправить, после значительного времени выделенного на доработку. Ориентируется в решении при изменении условий преподавателем.

Оценка «2» - Выполнено менее 3 заданий

Работа № 57.

ТЕМА: Пользователи on-line

ЦЕЛЬ: Разработать и создать систему для on-line диалога.

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОБЪЕМУ ВЫПОЛНЯЕМЫХ ЗАДАНИЙ:

Работа выполняется на компьютере с использованием языка гипертекстовой разметки HTML, PHP, web- форм и базы данных, сохраняется в личной директории.

Задания выполняются последовательно, без пропусков.

Условия контроля: аудиторно на занятии (5-10 минут в конце урока) или отсрочено на следующем занятии 5-10 минут (в начале урока).

Работа выполняется в ходе занятия (2 часа).

ФОРМА КОНТРОЛЯ: Экспертиза правильности решения задач. Устные ответы на вопросы преподавателя.

ЗАДАНИЯ:

1. Разработать и создать систему для on-line диалога пользователя с администратором Web-ресурса
2. Реализовать возможность комментирования, публикации статей, фотографий и т.п.

3. Реализовать возможность предоставления пользователю дополнительных возможностей после регистрации и авторизации на сайте

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЗАДАНИЯ:

Внимательно прочитайте задание.

Если не помните (не знаете), как его выполнить, обратитесь к материалам лекции и презентационному материалу.

При недостатке знаний, полученных на уроках-лекциях и предыдущих занятиях-практикумах, обратитесь к информационным источникам.

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Практическая работа (задание 1-3) выполнена правильно.

Оценка «4» - Практическая работа (задание 1-3) выполнена. Допущенные ошибки студент может исправить самостоятельно при указании на них преподавателя

Оценка «3» - Практическая работа (задание 1-2) выполнена правильно. Допущенные ошибки студент может исправить, после значительного времени выделенного на доработку. Ориентируется в решении при изменении условий преподавателем.

Оценка «2» - Выполнено менее 2 заданий

Работа № 58.

ТЕМА: **Дифференцированный зачет в форме защиты проекта**

ФОРМА КОНТРОЛЯ: Защита проекта. Устные ответы на вопросы преподавателя.

Зачет проводится в форме защиты проекта.

Оценка складывается суммарно из набранных при защите баллов.каждый балл определяется степенью реализации компонентов сайта в соответствии с критериями к разработке, оценка каждого критерия варьируется в диапазоне от 0 до 1 балла:

Критерий	Максимальный балл
1. Внешний вид (характеристика художественной оценки)	
Стилистическая выдержанность дизайна сайта - единый стиль цветов, шрифтов и графики. Сбалансированность цветового исполнения дизайна страниц. Уровень восприятия представленной информации. Читабельность текста. Доступность информации об авторских правах и обратной связи.	1 балл
2. Структура и навигация (характеристика эргономической оценки)	
Логичность организации содержания сайта. Место расположения навигационных элементов на разных страницах. Работоспособность ссылок. Простота использования навигации. Отсутствие тупиковых страниц. Наличие текстовых пояснений или всплывающих подсказок при использовании в навигации иконок. Возврат на предыдущие подуровни.	1 балл

3. Контент (характеристика информационного наполнения ресурса) Значимость. Актуальность. Полнота (информационная насыщенность сайта). Достоверность информации. Художественная ценность и стилистика. Отсутствие грамматических или синтаксических ошибок. Наличие системы поиска информации.	1 балл
4. Юзабилити (характеристика технологичности сайта) Многоплатформенность и возможность просмотра различными браузерами. Удобство поиска и доступа к информации. Использование форм.	1 балл
5. Интерактивность (реализация взаимодействия с пользователем) Реализации системы регистрации, авторизации пользователя на сайте. Реализация комментирования, публикации статей, фотографий и т.п. предоставление пользователю дополнительных возможностей после регистрации и авторизации на сайте.	1 балл

КРИТЕРИЙ ОЦЕНКИ:

Оценка «5» - Студент правильно ответил на теоретический вопрос. По всем практическим работам имеет положительные оценки. Демонстрирует отличное владение технологией решения задачи в соответствии с билетом (ориентируется в коде, может изменить код по требованию преподавателя).

Оценка «4» - Студент правильно ответил на теоретический вопрос. По всем практическим работам имеет положительные оценки. Демонстрирует хорошее владение технологией решения задачи в соответствии с билетом (ориентируется в коде, может изменить код по требованию преподавателя с использованием рабочей тетради).

Оценка «3» - Студент правильно ответил на теоретический вопрос (ответ краткий, не развернутый, при возникновении дополнительных вопросов в рамках теоретической части билета испытывает затруднения при ответе). По всем практическим работам имеет положительные оценки. Демонстрирует удовлетворительное владение технологией решения задачи в соответствии с билетом (плохо ориентируется в коде, может изменить код по требованию преподавателя только с использованием справочников, учебников).

Оценка «2» - Студент не может ответить на теоретический вопрос. Имеет задолженности по практическим работам.

ИНФОРМАЦИОННЫЕ ИСТОЧНИКИ

Основные источники:

1. Жадаев А.Г. PHP для начинающих СПб.: Питер, 2017. 288 с.
2. Зандстра М. PHP: объекты, шаблоны и методики программирования М.: Вильямс, 2016. 560с.
3. Исаченко О.В. Программное обеспечение компьютерных сетей. М.: Инфра-М, 2018. 117с.
4. Уэнц К. PHP и MySQL. Карманный справочник: Необходимый код и команды. М.: Вильямс, 2016. 256с.
5. Хоган Б. HTML 5 CSS3 веб-разработка по стандартам нового поколения СПб.: Питер, 2018. 272 с.
6. Шмитт К., Симпсон К. HTML5. Рецепты программирования. СПб.: Питер, 2018. 288 с.

Дополнительные источники, периодические профессиональные издания:

1. Гончаров А. Ю. Web-дизайн: HTML, JavaScript и CSS. Карманный справочник. М.: КУДИЦ-ПРЕСС, 2017. 320с.
2. Кухарчик А. PHP: обучение на примерах. М: Новое знание, 2014. 237с.
3. Мазуркевич А. Настольная книга программиста. М.: Новое знание, 2014. 480с.
4. Печников В.Н. Создание web-страниц и web-сайтов. Самоучитель: [учебное пособие]. М.: Триумф, 2016. 264с.
5. Томсон Л., Веллинг Л.. Разработка web-приложений на PHP и MySQL. СПб.: ДИА СофтЮП, 2013. 672с.
6. Ульман Л. Основы программирования на PHP. М.: ДМК Пресс, 2015. 288с.
7. Якоб Нильсен. Веб-дизайн. СПб: Символ-Плюс, 2015. 512 с.

Интернет-ресурсы:

1. Сидерхолм Дэн CSS3 для веб-дизайнеров (электронная книга) URL: <http://www.litmir.net/bd/?b=189371>
2. Джереми Кит HTML5 для веб-дизайнеров (электронная книга) URL: <http://www.litmir.net/bd/?b=189366>
3. Монтейро Майк Дизайн – это работа (электронная книга) URL: <http://www.litmir.net/bd/?b=189365>
4. Киссейн Эрин Основы контентной стратегии (электронная книга) URL: <http://www.litmir.net/bd/?b=189368>
5. Серых Ю.А. Современный веб-дизайн. Рисуем сайт который продает. (электронная книга) <http://www.litmir.net/bd/?b=205941>
6. Глазычев В. Дизайн как он есть (электронная книга) URL: <http://www.litmir.net/bd/?b=130486>
7. Иттан Маркотт Отзывчивый веб-дизайн (электронная книга) URL: <http://www.litmir.net/bd/?b=189371>
8. Аарон Уолтер Эмоциональный веб-дизайн (электронная книга) URL: <http://www.litmir.net/bd/?b=189370>
9. Методические материалы по дисциплине URL: ptb-online.ru (дата обращения 10.01.14)
10. Основы CSS и HTML. Блочная верстка. [веб - уроки] URL: <http://www.calameo.com/link?id=41512044>
11. Онлайн учебник по JavaScript [веб - уроки] URL: <http://www.webzz.ru/>
12. Перспективные технологии и языки веб-разработки [учебный курс] URL: <http://www.intuit.ru/department/internet/atlwebdev/class/free/status/>

13. Введение в технологии создания Интернет-узлов [учебный курс] URL: <http://www.intuit.ru/department/internet/inwwtech/class/free/status/>
14. Современные веб-технологии [учебный курс] URL: <http://www.intuit.ru/department/internet/mwebtech/class/free/status/>
15. Язык программирования PHP [учебный курс] URL: <http://www.intuit.ru/department/pl/plphp/class/free/status/>
16. Учебник HTML [урок on-line] URL: <http://webremeslo.ru/html/glava0.html>