Федеральное государственное бюджетное образовательное учреждение высшего образования
«РОССИЙСКАЯ АКАДЕМИЯ НАРОДНОГО ХОЗЯЙСТВА И ГОСУДАРСТВЕННОЙ СЛУЖБЫ ПРИ ПРЕЗИДЕНТЕ РОССИЙСКОЙ ФЕДЕРАЦИИ»

Нижегородский институт управления – филиал РАНХиГС
(наименование института (факультета))
Кафедра информатики и информационных технологий
(наименование кафедры)

УТВЕРЖДЕНО
Директор Нижегородского института управления – филиала РАНХиГС

ПРОГРАММА БАКАЛАВРИАТА

Корпоративные информационные системы управления

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
реализуемой с применением электронного (онлайн) курса

Б1.В.12.03 РАЗРАБОТКА WEB-ПРИЛОЖЕНИЙ
(индекс и наименование дисциплины (модуля), в соответствии с учебным планом)

направление подготовки
09.03.03 Прикладная информатика
(код и наименование направления подготовки)

очная, заочная
(форма(ы) обучения)

2024
(год набора)

Нижний Новгород, 2024 г.

Автор(ы)–составитель(и):

Доцент кафедры информатики и информационных технологий, кандидат физико-математических наук, доцент *(ученая степень и(или) ученое звание, должность, наименование кафедры)*	В.И. Окулич *(Ф.И.О.)*
Старший преподаватель кафедры информатики и информационных технологий *(ученая степень и(или) ученое звание, должность, наименование кафедры)*	Н.М. Трубилов *(Ф.И.О.)*
Заведующий кафедрой информатики и информационных технологий, кандидат технических наук, доцент *(ученая степень и(или) ученое звание, должность, наименование кафедры)*	И.И. Гребенюк *(Ф.И.О.)*

Изменения РПД Б1.В.12.03 «Разработка web-приложений» одобрены на заседании кафедры информатики и информационных технологий

Протокол от 20 февраля 2024 г. № 12.

Содержание

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения программы 4

2. Объем и место дисциплины в структуре ОП ВО....6

4. Материалы текущего контроля успеваемости обучающихся....12

5. Оценочные материалы промежуточной аттестации по дисциплине....24

6. Методические материалы по освоению дисциплины с применением ЭК....29

7. Учебная литература и ресурсы информационно-телекоммуникационной сети Интернет 32

7.1. Основная литература. 32

7.2. Дополнительная литература. 33

7.3 Учебно-методическое обеспечение самостоятельной работы. 33

7.4. Нормативные правовые документы. 33

7.5. Интернет-ресурсы. 33

8. Материально-техническая база, информационные технологии, программное обеспечение и информационные справочные системы 34

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения программы

1.1. Дисциплина «Разработка web-приложений» обеспечивает овладение следующими компетенциями:

Таблица 1

Код компетенции	Наименование компетенции	Код компонента компетенции	Наименование компонента освоения компетенции
ПКс-2	Способен разрабатывать и адаптировать прикладное программное обеспечение	ПКс-2.3	Использует приемы разработки и адаптации прикладных программных решений
ПКс-3	Способен осуществлять способен проектировать ИС по видам обеспечения	ПКс-3.2	Способен владеть приемами описания и разработки проекта информационной системы по видам обеспечения

1.2. В результате освоения дисциплины у студентов должны быть сформированы:

Таблица 2

ОТФ/ТФ/ трудовые или профессиональные действия	Код компонента компетенции	Результаты обучения
ОТФ: выполнение работ и управление работами по созданию (модификации) и сопровождению ИС, автоматизирующих задачи организационного управления и бизнес-процессы. ТФ: • определение первоначальных требований заказчика к ИС и возможности их реализации в ИС на этапе предконтрактных работ (С/01.6); • документирование существующих бизнес-процессов организации заказчика (реверс-инжиниринг бизнес-процессов организации) в рамках проекта создания (модификации) ИС (С/07.6); • разработка модели бизнес-процессов заказчика в рамках проекта создания (модификации) ИС (С/08.6); • адаптация бизнес-процессов заказчика ИС к возможностям ИС в рамках проекта создания (модификации) ИС (С/09.6); • выявление требований к ИС в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС (С/11.6); • разработка архитектуры ИС в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС (С/14.6); • разработка прототипов ИС в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС (С/15.6);	ПКс-2.3	На уровне знаний: На уровне знаний: −предметная область автоматизации; −программные средства и платформы инфраструктуры информационных технологий организаций; −современные подходы и стандарты автоматизации организации; −возможности типовой ИС; −предметная область автоматизации; −инструменты и методы моделирования бизнес-процессов; −устройство и функционирование современных ИС; −технологии межличностной и групповой коммуникации в деловом взаимодействии, основы конфликтологии; −современные стандарты информационного взаимодействия систем; −программные средства и платформы инфраструктуры информационных технологий организаций; −современные объектно-ориентированные языки программирования; −современные структурные языки программирования; −языки современных бизнес-приложений. На уровне умений: −разрабатывать документы в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС; −кодировать на языках программирования в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС.

ОТФ/ТФ/ трудовые или профессиональные действия	Код компонента компетенции	Результаты обучения
• разработка баз данных ИС в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС (С/17.6); • оптимизация работы ИС заказчика в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС (С/26.6). ТД: • сбор исходных данных у заказчика ИС о его бизнес-процессах в рамках проекта создания (модификации) ИС (С/09.6); • анализ функциональных разрывов и корректировка на его основе существующей модели бизнес-процессов заказчика ИС в рамках проекта создания (модификации) ИС (С/09.6); • утверждение у заказчика предлагаемых изменений его бизнес-процессов для реализации их автоматизации в типовой ИС (С/09.6); разработка прототипа ИС в соответствии с требованиями заказчика к ИС в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС (С/15.6).		На уровне навыков: • анализировать исходную документацию.
ОТФ: выполнение работ и управление работами по созданию (модификации) и сопровождению ИС, автоматизирующих задачи организационного управления и бизнес-процессы. ТФ: • определение первоначальных требований заказчика к ИС и возможности их реализации в ИС на этапе предконтрактных работ (С/01.6); • документирование существующих бизнес-процессов организации заказчика (реверс-инжиниринг бизнес-процессов организации) в рамках проекта создания (модификации) ИС (С/07.6); • разработка модели бизнес-процессов заказчика в рамках проекта создания (модификации) ИС (С/08.6); • адаптация бизнес-процессов заказчика ИС к возможностям ИС в рамках проекта создания (модификации) ИС (С/09.6); • выявление требований к ИС в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС (С/11.6); • разработка архитектуры ИС в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС (С/14.6);	ПКс-3.2	На уровне знаний: - возможности типовой ИС; - устройство и функционирование современных ИС; - инструменты и методы моделирования бизнес-процессов организации; - инструменты и методы выявления требований; - инструменты и методы оптимизации ИС. На уровне умений: - проектировать архитектуру ИС в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС. На уровне навыков: - тестировать результаты прототипирования.

ОТФ/ТФ/ трудовые или профессиональные действия	Код компонента компетенции	Результаты обучения
• оптимизация работы ИС заказчика в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС (С/26.6). ТД: • сбор исходных данных у заказчика ИС о его бизнес-процессах в рамках проекта создания (модификации) ИС (С/07.6, С/08.6); • описание бизнес-процессов заказчика ИС на основе полученных исходных данных в рамках проекта создания (модификации) ИС (С/07.6); • согласование с заказчиком ИС описания его бизнес-процессов в рамках проекта создания (модификации) ИС (С/07.6, С/08.6); • утверждение у заказчика ИС описания его бизнес-процессов в рамках проекта создания (модификации) ИС (С/07.6, С/08.6); • разработка модели бизнес-процессов заказчика ИС в рамках проекта создания (модификации) ИС (С/08.6); • утверждение у заказчика ИС модели его бизнес-процессов в рамках проекта создания (модификации) ИС (С/08.6); • моделирование бизнес-процессов заказчика в ИС в рамках проекта создания (модификации) ИС (С/09.6); • создание вариантов архитектурных спецификаций ИС в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС (С/14.6); • выбор и согласование с заинтересованными сторонами оптимальной архитектурной спецификации ИС в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС(С/14.6).		

1.3. Студенты также должны овладеть навыками:

- анализа исходной документации;
- тестирования результатов прототипирования;
- разработки и адаптации прикладных программных решений.

2. Объем и место дисциплины в структуре ОП ВО

Индекс дисциплины Б1.В.12.03. «Разработка web-приложений» является составной частью модуля «Создание сервисов информационных систем» части, формируемой участниками образовательных отношений..

Общая трудоемкость дисциплины составляет 6 зачетных единиц, 288 академических часов / 216 астрономических часов.

По очной форме обучения: количество академических часов, выделенных на контактную работу с преподавателем составляет 114 часов, из них, лекции - 48 часов, практические занятия - 64 часа, консультация – 2 часа. Самостоятельная работа составляет - 134 часов, контроль – 40 часов.

По заочной форме обучения: количество академических часов, выделенных на контактную работу с преподавателем составляет 22 часов, из них, лекции – 8 часов, практические занятия – 12 часов, консультация – 2 часа. Самостоятельная работа составляет 257 часа, контроль – 9 часов.

Форма промежуточной аттестации в соответствии с учебным планом: зачет, экзамен.

3. Содержание и структура дисциплины

3.1 Структура дисциплины

Таблица 3.

№ п/п	Наименование тем (разделов)	Объем дисциплины, час.						Форма текущего контроля успеваемости, промежуточной аттестации
		Всего	Контактная работа обучающихся с преподавателем по видам учебных занятий				СР	
			Л/ЭО, ДОТ*	ЛР/ ЭО, ДОТ*	ПЗ/ ЭО, ДОТ*	КСР		
Очная форма обучения								
Тема 1	Структура Интернета и всемирной веб-сети. Семейство протоколов TCP/IP.	12	2		2		8	О
Тема 2	Основные характеристики протокола HTTP.	12	2		2		8	О
Тема 3	Основы языка PHP и его основополагающие особенности.	14	2		2		10	О, Пз
Тема 4	Синтаксис и семантика операторов PHP.	14	2		2		10	О, Пз
Тема 5	Основы работы с массивами.	14	2		2		10	О, Пз
Тема 6	Функции в PHP. Создание и использование пользовательских функций в PHP	14	2		2		10	О, Пз
Тема 7	Работа с файлами	14	2		2		10	О, Пз
Тема 8	Взаимодействие PHP, HTML, CSS. HTML-формы и их использование в PHP.	14	2		2		10	О, Пз
Промежуточная аттестация								**зачет**
Тема 9	Использование MySQL для создания веб-приложений.	20	4		6		10	О, Пз
Тема 10	Безопасность веб-приложений.	14	2		6		6	О, Пз
Тема 11	Работа с датой и временем, генерация изображений.	18	4		6		8	О, Пз

№ п/п	Наименование тем (разделов)	Объем дисциплины, час.						Форма текущего контроля успеваемости, промежуточной аттестации
		Всего	Контактная работа обучающихся с преподавателем по видам учебных занятий				СР	
			Л/ЭО, ДОТ*	ЛР/ ЭО, ДОТ*	ПЗ/ ЭО, ДОТ*	КСР		
Тема 12	Управление сеансами в PHP.	20	4		6		10	О, Пз
Тема 13	Системы управления содержанием сайта (CMS). CMS Joomla: основные понятия и установка.	20	6		6		8	О, Пз
Тема 14	Создание сайта с помощью CMS Joomla.	22	6		8		8	О, Пз
Тема 15	Введение в создание расширений Joomla.	24	6		10		8	О, Пз
Консультация		**2**						
Промежуточная аттестация		**40**						**Зачет, экзамен**
Всего (в з.е./акад.часах/астр.часах):		**8/288/216**	**48/36**		**64/48**		**134/ 100,5**	**40/30**
Заочная форма обучения								
Тема 1	Структура Интернета и всемирной веб-сети. Семейство протоколов TCP/IP.	10,5	0,5				10	О
Тема 2	Основные характеристики протокола HTTP.	10,5	0,5				10	О
Тема 3	Основы языка PHP и его основополагающие особенности.	17,5	0,5				17	О, Пз
Тема 4	Синтаксис и семантика операторов PHP.	18,5	0,5		1		17	О, Пз
Тема 5	Основы работы с массивами.	18,5	0,5		1		17	О, Пз
Тема 6	Функции в PHP. Создание и использование пользовательских функций в PHP	18,5	0,5		1		17	О, Пз
Тема 7	Работа с файлами	18,5	0,5		1		17	О, Пз
Тема 8	Взаимодействие PHP, HTML, CSS. HTML-формы и их использование в PHP.	18,5	0,5		1		17	О, Пз
Тема 9	Использование MySQL для создания веб-приложений.	25,5	0,5		1		24	О, Пз
Тема 10	Безопасность веб-приложений.	18,5	0,5		1		17	О, Пз
Тема 11	Работа с датой и временем, генерация изображений.	18,5	0,5		1		17	О, Пз
Тема 12	Управление сеансами в PHP.	11,5	0,5		1		10	О, Пз

№ п/п	Наименование тем (разделов)	Объем дисциплины, час.						Форма текущего контроля успеваемости, промежуточной аттестации
		Всего	Контактная работа обучающихся с преподавателем по видам учебных занятий				СР	
			Л/ЭО, ДОТ*	ЛР/ ЭО, ДОТ*	ПЗ/ ЭО, ДОТ*	КСР		
Тема 13	Системы управления содержанием сайта (CMS). CMS Joomla: основные понятия и установка.	18,5	0,5		1		17	О, Пз
Тема 14	Создание сайта с помощью CMS Joomla.	27,5	0,5		1		26	О, Пз
Тема 15	Введение в создание расширений Joomla.	26	1		1		24	О, Пз
Консультация		**2**						
Промежуточная аттестация		**9**						**Экзамен**
Всего (в з.е./акад.часах/астр.часах):		**8/288/216**	**8/6**		**12/9**	**4/3**	**257/ 192,75**	**9/6,75**

Примечание:

** – при применении электронного обучения, дистанционных образовательных технологий в соответствии с учебным планом*

*** – формы текущего контроля успеваемости: опрос (О), тестирование (Т),), оценка доклада (Д), оценка работы с документами в микрогруппе (М), оценка презентации(Пр), отчет по практическим занятиям (Пз), отчет по лабораторным работам (Лб) .*

3.2 Содержание дисциплины

Тема 1 Структура Интернета и всемирной веб-сети.

Семейство протоколов TCP/IP. Развитие интернет-сети от веб-страниц к веб-сайтам и веб-приложениям. Понятие веб-сервера. Клиентские и серверные веб-приложения. Статические и динамические веб-страницы. Основные протоколы веб-сети. Парадигма и структура протокола TCP/IP. Линейка протоколов – сервисных приложений TCP/IP.

Тема 2 Основные характеристики протокола HTTP

Базовые компоненты веб-технологии: HTML, URI и HTTP. Основные положения протокола HTTP. Структура HTTP-сообщений. Методы запроса к веб-серверу и их сравнение. HTTP-заголовки как источники дополнительной информации. Классификация заголовков. Структура HTTP-ответа сервера. Решение вопросов безопасности передачи сообщений в HTTP-протоколе. Понятие сессий и cookie: назначение, использование и опасности. Эволюция протокола HTTP.

Тема 3. Основы языка PHP и его основополагающие особенности.

Особенности разработки web приложений. Взаимодействие web-сервера с клиентской программой (браузером). Особенности языка PHP как основного языка разработки web приложений, история его развития. Логика работы обработчика PHP (обработчик PHP – интерпретатор или транслятор?). Понятие скрипта PHP. Совместное использование тегов HTML и кода PHP.

Среды разработки и выполнения проектов PHP: настройка и особенности использования.

Начальные сведения о синтаксисе PHP: идентифицирующие теги, регистрозависимость идентификаторов, разделение инструкций, использование

кириллицы, отображение текста, оформление комментариев, использование escape-последовательностей (управляющих символов).

Основы понятия «тип данного» и классификация типов данных в PHP.

Интервалы значений числовых данных в PHP.

Строки в PHP и особенности работы с ними. Особенности работы с переменными и константами в PHP.

Начальное введение в тему функции в PHP. Правила использования. Стандартные функции PHP. Определение и роль строковых функций в PHP.

Понятие выражения в PHP, типы выражений и их использование. Манипуляции с типами выражений в PHP.

Тема 4. Синтаксис и семантика операторов PHP.

Классификация операторов PHP и порядок их выполнения. Математические операторы и функции. Операторы присваивания. Операции декремента и инкремента. Строковые операторы. Семейство побитовых операторов. Операторы сравнений и логических операций. Понятие логического выражения.

Операторы управления выполнением скрипта: условного перехода, переключения, цикла – их синтаксис, семантика и особенности выполнения.

Тема 5. Основы работы с массивами.

Общие вопросы использования массивов в PHP: определение, типы, использование в выражениях и размерность. Создание одномерных массивов и особенности их использования в PHP. Понятие индекса и ключа массива. Определение и особенности использования ассоциативных массивов в PHP.

Операции с массивами и вывод их значений на экран. Действия над массивами и их операторы. Изменение, перебор и удаление элементов массива. Использование указателя элемента массива.

Базовые функции по работе с массивами: сортировка, преобразование строк в массивы и массива в строки.

Специальные функции обработки массивов.

Многомерные массивы: создание и доступ к их элементам. Решение задачи перебора элементов многомерного массива.

Тема 6. Функции в PHP. Создание и использование пользовательских функций в PHP

Определение функции в PHP. Синтаксис и семантика описания и вызова функции. Способы передачи данных (по ссылке и значению) в функцию и возврат результата её работы. Передача массивов в функцию. Возврат массивов из функции. Использование списков. Возврат ссылок.

Понятия областей действия и видимости, а также времени жизни переменной. Глобальные, суперглобальные и статические переменные.

Функции с переменным числом аргументов.

Архитектура проекта PHP, включающего использование пользовательских функций. Обеспечение подключения функций к проекту.

Дополнительные возможности по работе с функциями PHP:

- переменные, ссылающиеся на функции,
- условное (динамическое) описание функций;
- вложенные файлы и функции;

Обработка ошибок в функциях пользователя.

Рекурсивные функции: определение и механизм функционирования.

Тема 7. Работа с файлами

Понятие файла и файловой системы. Свойства файла. Операции и сущности, связанные с файлами. Классификация файлов и операций над ними.

Возможности работы с файлами в PHP. Расположение файлов по отношению к скрипту PHP. Понятия имени файла и пути до него в скриптах PHP. Классификация путей до файла и их использование в функциях по работе с файлами в PHP. Понятие унифицированного идентификатора ресурса.

Синтаксис и семантика базовой функции по работе с файлами в PHP - открытия файла. Режимы открытия файла. Использование служебных массивов PHP при работе с файлами.

Основные функции по работе с файлами PHP, обеспечивающие базовые операции с ними:

- Запись в файл и запись файла целиком.
- Чтение из файла.
- Чтение строк текста.
- Проверка существования и размера файла.
- Закрытие файла.
- Посимвольное чтение файла.
- Чтение файла в индексированный массив.
- Чтение файла целиком в строку.
- Чтение файла с синтаксическим разбором.

Обеспечение прав доступа пользователей по доступу к файлам: понятие, классификация, реализация.

Тема 8. Взаимодействие PHP, HTML, CSS. HTML-формы и их использование в PHP. .

Взаимодействие PHP и HTML. Взаимодействие PHP и CSS.

Переменные сервера и их использования. Отображение данных формы. Использование переменных сервера. Работа с заголовками HTML-формы. Реализация перенаправления браузера на нужную страницу. Приложение из одной страницы. Организация проверки данных числовых и строковых полей. Сохранение данных. Понятие аутентификации пользователя и её реализация в протоколе HTTP.

Тема 9. Использование MySQL для создания веб-приложений.

Установка и настройка СУБД MySQL для работы с PHP в ОС Windows и IDE Net Beans. Кроссплатформенная сборка XAMPP – назначение, настройка, использование для работы с СУБД MySQL. Обзор инструментов для работы с СУБД MySQL. Веб-доступ к базе данных MySQL с помощью PHP: соединение с сервером, создание базы данных и таблицы, добавление, удаление, отображение и сортировка данных. Отсоединение от базы данных. Процедурный и объектно-ориентированный интерфейс взаимодействия PHP с СУБД MySQL.

Тема 10. Безопасность веб-приложений.

Классификация и характеристики угроз безопасности веб-приложению. Администрирование баз данных MySQL. Понятие и принципы аутентификации пользователя. Её реализация с помощью PHP и MySQL. Защита сервера баз данных MySQL. Методы фильтрации пользовательского кода и литерализации выходных данных. Защита веб-браузера и PHP. Настройка безопасности веб-серверов. Понятие DoS и DDoS-атак.

Тема 11. Работа с датой и временем, генерация изображений.

Получение даты и времени средствами PHP. Генерация изображений в PHP: настройка поддержки изображений, Форматы и создание изображений. Использование сгенерированных изображений на других страницах.

Тема 12. Управление сеансами в PHP.

Определение сеанса связи пользователя с веб-сервером пользователя. Ограничения протокола HTTP в аутентификации автора запроса. Понятие переменных сеанса. Функции PHP для управления сеансами. Использование сеанса и его реализация. Назначение, определение и использование Cookie-набора. Установка, чтение, срок действия и удаление cookie.

Тема 13. Системы управления содержанием сайта (CMS).

CMS Joomla: основные понятия и установка. История развития, типы версий и архитектура Joomla. Типы сайтов, созданных на основе Joomla. Этапы установки CMS. Клиентская часть сайта. Административная часть сайта. Компонентная структура Joomla. Расширения Joomla: компоненты, модули, плагины, шаблоны. Проблема совместимости расширений. Файловая структура Joomla. Структура базы данных. Основные источники сведений по Joomla в Internet: joomla.org, joomla.ru, joomlaportal.ru, joomlaforum.ru.

Тема 14. Создание сайта с помощью CMS Joomla.

Настройки Joomla: сайт, система, сервер, права, фильтры текста. Шаблоны. Страницы. Навигация. Управление пользователями и правами. Управление расширениями: установка, настройка прав, возможность модификации. Сайт – визитка. Корпоративный сайт. Интернет магазин. Блог. SEO оптимизация и продвижение сайта.

Тема 15. Введение в создание расширений Joomla.

Платформа Joomla: особенности API. Компонент: файловая структура и особенности создания. Модуль: файловая структура и особенности создания. Плагин: файловая структура и особенности создания. Шаблон: файловая структура и особенности создания.

4. Материалы текущего контроля успеваемости обучающихся

4.1. В ходе реализации дисциплины «Разработка web-приложений» используются следующие методы текущего контроля успеваемости обучающихся:

Таблица 4.

№	Тема (раздел)	Методы текущего контроля успеваемости
Тема 1	Структура Интернета и всемирной веб-сети. Семейство протоколов TCP/IP.	Опрос
Тема 2	Основные характеристики протокола HTTP.	Опрос
Тема 3	Основы языка PHP и его основополагающие особенности.	Опрос, отчет по практическим занятиям
Тема 4	Синтаксис и семантика операторов PHP.	Опрос, отчет по практическим занятиям
Тема 5	Основы работы с массивами.	Опрос, отчет по практическим занятиям
Тема 6	Функции в PHP. Создание и использование пользовательских функций в PHP	Опрос, отчет по практическим занятиям

№	Тема (раздел)	Методы текущего контроля успеваемости
Тема 7	Работа с файлами	Опрос, отчет по практическим занятиям
Тема 8	Взаимодействие PHP, HTML, CSS. HTML-формы и их использование в PHP.	Опрос, отчет по практическим занятиям
Тема 9	Использование MySQL для создания веб-приложений.	Опрос, отчет по практическим занятиям
Тема 10	Безопасность веб-приложений.	Опрос, отчет по практическим занятиям
Тема 11	Работа с датой и временем, генерация изображений.	Опрос, отчет по практическим занятиям
Тема 12	Управление сеансами в PHP.	Опрос, отчет по практическим занятиям
Тема 13	Системы управления содержанием сайта (CMS). CMS Joomla: основные понятия и установка.	Опрос, отчет по практическим занятиям
Тема 14	Создание сайта с помощью CMS Joomla.	Опрос, отчет по практическим занятиям
Тема 15	Введение в создание расширений Joomla.	Опрос, отчет по практическим занятиям

4.2. Типовые материалы текущего контроля успеваемости обучающихся

Материалы текущего контроля успеваемости по теме 1.

Вопросы для контроля успеваемости в форме опроса

1. Чем отличаются друг от друга веб-страницы, веб-сайты и веб-приложения?
2. Что такое «веб-приложения»? Назовите выполняемые ими задачи.
3. Какую программу использует веб-приложение в качестве клиентской?
4. Чем отличается статическая веб-страница от динамической?
5. Чем обеспечивается динамичность веб-страницы?
6. В чём заключается радикальное отличие технологии AJAX от ей предшествующих?
7. Чем является TCP/IP – протоколом или их семейством?
8. Назовите наиболее важные протоколы, реализующие сервисные приложения TCP/IP, их уровень и назначение.

Материалы текущего контроля успеваемости по теме 2.

Вопросы для контроля успеваемости в форме опроса

1. Какое место занимает HTTP-протокол в семействе TCP/IP?
2. Что скрывается под аббревиатурами URL, URN и URI?
3. Назовите составляющие URL. Какие из них являются обязательными?
4. Место прокси-серверов HTTP в виртуальной цепи «запрос-ответ». Могут ли они работать в качестве HTTP-клиентов?
5. Чем отличаются протоколы передачи сообщений с запоминанием состояния и без запоминания состояния? К какому из этих классов относится протокол HTTP?
6. Какова структура HTTP-сообщений? Кто или что читает их?
7. Как отделяется заголовок HTTP-сообщения от тела сообщения?
8. Перечислите основные методы запросов, возможнее в HTTP-сообщении. В чём их различие и как они идентифицируются сервером?

Материалы текущего контроля успеваемости по теме 3.

Вопросы для контроля успеваемости в форме опроса

1. Что такое скрипт? На каком процессоре выполняется скрипт PHP?

2. Что означает понятие «кроссплатформенность» языка или технологии? Является ли технология использования PHP кроссплатформенной?

3. Нарисуйте схему работы скрипта на PHP и опишите её. Как интерпретатор и компилятор распознают части кода PHP?

4. Назовите и опишите базовые объекты программы на языке PHP.

5. Понятие имени переменной и правила его использования в PHP.

6. Перечислите типы и категории данных языка PHP и охарактеризовать их особенности.

7. Что означает термин «пустая переменная»?

8. С какими целями вводятся константы? Понятие области видимости в языках программирования. Какова область видимости констант в PHP? Какой тип данных может «содержаться» в константах? Дайте полное описание функции задания константы в PHP.

9. Дайте определение функции в PHP. Что означает выражение «Функция возвращает значение»? Как классифицируются функции в PHP? Как определить функцию в PHP? Перечислите правила использования функций в скриптах PHP.

10. Дайте своё понимание понятию «выражение» в PHP. Является ли функция PHP выражением? Что такое «тип выражения», и по каким правилам он определяется в PHP?

Практические задания:

1. Рассмотреть работу функции date(…) (найти в интернете) и написать скрипт, вычисляющий с её помощью текущие дату в формате «день.месяц.год» и время в формате «h:i:s», а также выводящую эти значения в виде двух строк: Текущая дата: … года и Текущее время: … .

2. Присвоить двум переменным значение двух чисел и вывести результаты их сложения и конкатенации. Далее скрипт должен, удаляя по одной переменной и проверяя наличие каждой из них, проверить результат сложения и конкатенации этих переменных.

3. Разработать алгоритм вычисления факториала от числа n. Создать функцию, вычисляющую значения факториала n!, и скрипт, вычисляющий значение факториала с помощью этой функции.

4. Написать и выполнить скрипт, в котором применялись бы различные стандартные функции по работе с текстом.

5. Проиллюстрируйте скриптом правила преобразования типов переменных.

Материалы текущего контроля успеваемости по теме 4.
Вопросы для контроля успеваемости в форме опроса

1. Дайте обобщённое определение оператора языка программирования. Можно ли сказать, что функция является оператором языка?

2.Классифицируйте операторы PHP. Что определяет приоритет операторов? Чему соответствует понятие «ассоциативность» оператора?

3. Является ли оно эквивалентом понятию «приоритет» оператора?

4. Являются ли предопределённые константы, определяющие значения общеупотребительных математических констант функциями?

5. Зачем в современных языках программирования введены операторы декремента и инкремента? Как работают операторы инкремента и декремента? Какие вы знаете типы этих операторов? Приведите примеры использования всех известных вам строковых операторов языка PHP. Какие побитовые операторы PHP вы знаете, как и с какими объектами они работают, для чего могут применяться?

6. Посредством каких операторов программист может управлять ходом выполнения операторов (решением задачи) в программе?

7. Перечислите операторы сравнения PHP и результаты их работы. Какие объекты языка PHP нельзя сравнивать между собою? Можно ли в PHP сравнивать между собой строку с вещественным или целым числом? А две числовые строки?

8. Как определяется логическое выражение в PHP? Какой вид элементарных логических выражений в PHP? Приведите примеры. Нарисовать таблицу истинности для всех известных вам логических операторов PHP. Есть ли у логических операторов приоритет выполнения? Всегда ли логические выражения выполняются «до конца»? Дайте пример использования логических переменных с операцией декремента или инкремента.

9. Сколько форм имеет оператор условного перехода? Опишите работу условного оператора перехода с помощью блок-схемы. Опишите применение конструкции **elseif** в операторе условного перехода. Области применения и семантика оператора **switch**. Отобразите в том числе с помощью блок-схемы. Чем должен заканчиваться оператор переключения (выбора)? С помощью какого оператора в PHP реализуется многонаправленное ветвление?

10. Чем операторы цикла отличаются от операторов ветвления? Выразить семантику операторов семейства while блок-схемами. Сколько форм имеют операторы семейства while?

11. В чём принципиальное различие операторов цикла с пост- и предварительным условием?

12. Подробно описать работу традиционной формы оператора **for**. Как называются структурные части этого оператора?

13. Как работает цикл **for** с инкрементом?

14. Для чего счётчик цикла оператора цикла может использоваться в теле цикла? Приведите логическое обоснование использования вложенных циклов **for**. Правила построения вложенных циклов. Подробно опишите формы универсальной формы for. Синтаксис и семантика всех форм оператора foreach.

Практические задания:

1. С помощью соответствующего скрипта проверить, как интерпретатор PHP производит арифметические действия над операндами смешанного типа – числа и строки.

2.Провести те же операции, что и в предыдущем скрипте, применив перед выполнением операций необходимое преобразование операндов. Использовать для этого как функцию **settype( )**, так и стандартное приведение типов.

3.Написать скрипт, в котором определяются или, как второй вариант, не определяются имя, фамилия и отчество человека. В первом случае скрипт должен сообщить, что все данные записаны и поблагодарить. Во втором – сообщить, что данные не определены, и попросить их ввода. В скрипте использовать логический оператор ИЛИ.

4. Распечатать случайное число раз (но не более 19) любой символ (звездочку, вопросительный знак), используя оператор «если». Необходимое число символов определить, используя функцию выработки случайного числа в интервале (1-10) **rand( )**.

5.Ввести три логических переменных: $l1, $l2 и $l3. Построить логические выражения, в которых эти три величины образовывали логическое выражение, соединяясь знаками логических операций в различных сочетаниях. Каждое логическое выражение проверить на истинность и вывести на экран текстовое заключение об этом.

6. Используя оператор **elseif**, написать скрипт, который выдавал бы значение налога в зависимости от полученной прибыли , а именно: если прибыль <= величины P_{min}, используется минимальная процентная ставка налога stavkaMin, если прибыль > P_{max}, ставка налога равна stavkaMax, противном случае используется промежуточная ставка налога stavkaSrdn.

7. Написать скрипты, вычисляющие сумму членов арифметической прогрессии. Использовать различные известные вам операторы цикла. Сумма должна ограничиваться некоторым максимальным или минимальным значением.

8. Пять матрёшек вставлены друг в друга. Написать скрипт, отображающий процесс открывания матрёшек. Скрипт должен оповещать, сколько матрёшек осталось открыть и факт завершения раскрытий.

9. Написать скрипт, реализующий известную сказку о богатыре, стоящем на распутье перед тремя дорогами. Если богатырь пойдёт налево, о скрипт выдаст информацию, о том, что он получит полцарства. Если пойдёт направо, то получит коня, а если прямо – найдёт принцессу. Если параметр, определяющий выбор, будет не соответствовать ни одному из предложенных вариантов, скрипт должен выдать сообщение «Трусливым ничего не светит!».

10. Предприятие имело начальный капитал в размере ***A*** тыс.руб. В первый месяц его прибыль составила ***m*** процентов от начального капитала. Каждый последующий месяц его прибыль составляла ***n*** процентов от прибыли предыдущего месяца. Найти общий капитал предприятия за ***k*** месяцев работы. Найти номер месяца, в котором общий капитал предприятия превысит ***B*** тыс.руб.

Примечание: общий капитал предприятия ***S*** после ***k*** месяцев работы вычисляется по формуле:

$$S = A\left(1 + \frac{m}{100}\left(1 + \frac{n}{100} + \left(\frac{n}{100}\right)^2 + \ldots + \left(\frac{n}{100}\right)^{k-1}\right)\right)$$

Материалы текущего контроля успеваемости по теме 5.

Вопросы для контроля успеваемости в форме опроса

1. Дайте определение массива, элемента массива и ключа элемента массива. Что такое численно-индексированный массив?
2. Дайте определение и особенности использования многомерных массивов? Что такое ассоциативный массив?
3. Опишите синтаксис использования массива – создания и вставки в выражении.
4. Как в PHP разрешается ситуация, когда программист забудет указать индекс массива?
5. Может ли массив иметь одновременно численные и строковые индексы?
6. Вы в скрипте случайно реализовали вариант использования элементов одного массива, но разного типа. Какова будет реакция интерпретатора PHP на эту ситуацию?
7. Опишите синтаксис создания элемента ассоциативного массива.
8. Как создаются массивы, элементы которых будут автоматически принимать значения в некотором интервале?
9. Какие функции удобно применять для вывода элементов массива.
10. Приведите пример использования дескриптора **<pre>** при выводе массивов на экран браузера.
11. Как вывести элемент массива в строке, обрамлённой двойными кавычками?
12. Правильно ли утверждение «массивы в PHP не являются ненаправленными», и что это означает?
13. Перечислите функции работы с указателем элемента массива в PHP, опишите их синтаксис и расшифруйте семантику.
14. Как в PHP используется функция **unset()**.
15. Какие возможности сортировки массивов существуют в PHP, и с чем это связано?

16. Перечислите простейшие функции сортировки массивов разного типа в PHP.

17. Опишите возможности взаимного преобразования строк и массивов в PHP.

18. Как можно интерпретировать многомерные массивы в PHP?

19. Как создать многомерный массив в PHP? Приведите пример создания такого массива и присвоения значения элементу такого массива.

20. Приведите примеры ссылок на элементы многомерных массивов разными способами.

21. Как осуществляется перебор элементов многомерных массивов в PHP? Зависит ли способ перебора от особенностей массива? Если зависит, то как?

Практические задания:

1. Создать скрипт, описывающий содержание цифрового семейного фотоальбома, в котором находятся несколько фотографий (например 5) ваших родственников.
2. Проанализировать и выполнить приведённый ниже скрипт, предварительно предсказав результат его работы:

```
<?php
$flowers = array (
  array('name'=>'magnolia', 'лепестки'=>'8', 'тычинки'=>'5',
       'пестики'=>'2', 'цвет'=>'белый'),
  array('name'=>'primula', 'лепестки'=>'10', 'тычинки'=>'5',
       'пестики'=>'5', 'цвет'=>'алый'),
  array('name'=>'campanula latifolia', 'лепестки'=>'10',
       'тычинки'=>'5', 'пестики'=>'3', 'цвет'=>'фиолетовый')
);
for ($row=0; $row<3; $row++)
{
  echo "|". $flowers[$row]["name"]. "|". $flowers[$row]["лепестки"]. "|".
     $flowers[$row]["тычинки"]."| <br>";
}
?>
```

3. Изучить работу функции **shuffle( )**. Написать скрипт, реализующей следующую задачу:

а. Турфирма имеет набор файлов (например, формата .jpg) с видами 10 российских городов;

б. При входе на сайт этой фирмы посетитель видит таблицу с видами трёх из этих городов, выбранных случайным образом.

Указание: таблицу отображать средствами HTML.

4. Создать двумерный массив, отображающий меню ресторана из, где на первое – одно блюдо, на второе – одно блюдо, а на третье (десерт)– три возможных варианта. Распечатать меню с помощью функции **print_r()**. Сделать то же самое для меню, где первые, вторые и третьи блюда имеют по 2ва варианта.

Материалы текущего контроля успеваемости по теме 6.

Вопросы для контроля успеваемости в форме опроса:

1. Что такое модульное программирование и структурирование программ? Причины перехода широкому использование модулей при программировании.

2. Дайте определение функции и её основных характеристик. Что такое «описание функции», и где оно может размещаться? Представьте синтаксис описания функции в PHP. Может ли функция не иметь параметров?

3. Что такое аргументы функции? Можно ли сконструировать (создать) функцию с неизвестным числом аргументов? Какие способы передачи аргументов вы знаете, в чём они различаются?

4. Расскажите о назначении использования жёстких и символических ссылок в PHP. Приведите примеры использования жёстких и символических ссылок в PHP. Разъясните механизм использование ссылки при передаче значения в функцию (опишите синтаксис и семантику этого способа).

5. В каких случаях применяется передача значения аргумента в функцию по умолчанию? Как организовать передачу массива в функцию? Приведите примеры. Какой способ передачи данных в функцию называется передачей по значению и чем он отличается от передачи данных по ссылке (имени)? Как организовать возврат значения, вычисленного в функции по ссылке, и что это значит?

6. Какой оператор обеспечивает возврат в скрипт единичного значения? Опишите его синтаксис. Может ли в функции использоваться несколько операторов возврата значения?

7. Опишите имеющиеся в PHP способы, обеспечивающие возврат в скрипт новых массивов, полученных в результате работы функции.

8. Дайте определение понятия «область действия» переменной, «область видимости» переменной. К каким объектам PHP относятся эти понятия? Как сделать переменные скрипта глобальными для подсоединяемых к нему функций?

9. Дайте определение суперглобальных массивов в PHP, к какому типу они относятся. Приведите наименования таких массивов и примеры их использования.

10. Определите семантику понятия и использования «предопределённых переменных». Раскройте необходимость введения статических переменных, опишите семантику использования и синтаксис определения этих переменных.

11. Можно ли в PHP вызвать функцию через имя переменной? Если можно, то как?

12. В чём заключается смысл динамического описания функции? В чём разница между динамическим и условным описание функции?

13. Как использовать условный оператор **if** для условного описания функции?

14. Можно ли вызывать неописанную функцию? Можно ли описывать функцию дважды?

15. Как использовать вложенные функции и файлы? Какова цель их использования? Где хранить вложенные файлы? Какую команду надо использовать для вставки файла в нужное место? Опишите синтаксис и расширение вложенных файлов. Составьте пример их использования.

16. Как в PHP обеспечить обработку вычислительных ошибок, если они возникнут при работе функции?

17. Перечислите важнейшие положения по работе с функциями PHP в части передачи данных в них и возврата результата из функции. Перечислите важнейшие положения по работе с функциями PHP в части реализации различных вариантов «видимости» и сохранения переменных скриптов и функций. В чём разница между внутренней для скрипта и внешней для него функции? Как организован их вызов?

Практические задания:

1. Реализовать расчёт факториала от целого числа в виде итеративной функции и использовать её в скрипте. Применить эту функцию как внутреннюю, так и как внешнюю по отношению к скрипту.

2. Выполнить пункт 1, но функция, вычисляющая значение факториала должна быть рекурсивной.

3. Придумать и реализовать использование рекурсивной функции для такого вычисления, в котором заранее неизвестно количество повторяющихся вычислений.

Материалы текущего контроля успеваемости по теме 7.

Вопросы для контроля успеваемости в форме опроса:

1. Дайте определение сущности файл. Перечислите основные свойства файлов. Перечислите основные типы файлов. Дайте развёрнутое толкование для каждого из следующих типов файла: исполняемых, каталогов и обычных. Сформулируйте, в чём разница между бинарными и текстовыми файлами. Что означает термин «обычный файл»? Где могут располагаться файлы, к которым обращаются скрипты PHP?
2. Состав и классификация операций с файлами. Дайте определение и опишите назначение дескриптора и указателя файла. Как используется дескриптор файла в скриптах PHP? Опишите синтаксис функции открытия файла и особенности её использования. Перечислите операции с файлами, требующими его открытия. Какие операции с файлами не требуют его открытия?
3. Классификация путей до файла и их использование. Приведите примеры возможных путей до файлов со скриптами. Понятие каталога веб-сервера. Отличие корня веб-сервера от корня файловой системы.
4. В чём разница отработки системы дорожек до файлов для URI, начинающихся с префиксов ftp: и http: ?
5. Перечислите варианты режимов работы с открываемым файлом, обеспечиваемые параметром mode при вызове функции открытия файла. Как шифруются эти режимы?
6. Какие значения возвращает функция открытия файлов в зависимости от результата открытия?
7. Дайте анализ возможных ситуаций с абсолютными и относительными путями до файлов. Чем различается соответствующие решения для ОС Windows и Unix?
8. Перечислите функции PHP по работе с файлами и сформируйте классификатор этих функций.
9. Для каких операционных систем имеет смысл цифровое обозначение прав (к примеру – 777)
10. Опишите в понятиях rwx для различных категорий пользователей права, имеющие следующие цифровые обозначения:
    a. 777
    b. 644
    c. 770

    В чем будет разница при работе web-сервера с файлами, для которых прописаны эти права при запросе, отправленному с помощью браузера.

Практические задания:

1. Самостоятельно выведите содержимое произвольного каталога с помощью функции scandir().
2. Написать скрипт, выводящий список файлов и подкаталогов, содержащихся в текущем каталоге.
3. Написать скрипт, подсчитывающий число файлов в каталоге.
4. Реализовать скрипт, осуществляющий копирование одного каталога (с вложенными каталогами произвольной глубины) в другой. Указание: использовать функцию copy() и mkdir().
5. Разобрать представленный ниже скрипт. Написать краткий отчёт по нему, включающий:
    - Определение функциональности скрипта – то есть, описать, что делает скрипт;
    - Указать функции, применённые скриптом, описать их синтаксис и семантику;

- Описать назначение переменных, используемых в скрипте;

```
<?php
function files_names( $nach_dir )
{
/* Открывается директория, имя которой было сообщено в начале выполнения функции */
 $directoriya = opendir( $nach_dir );
/*Запускается цикл while, который будет выполняться до тех пор,
пока не будут считаны все файлы из папки
*/
while( ( $uroven = readdir( $directoriya ) ) !== false )
 {
   if ( $uroven != "." && $uroven != ".." )
/*
 Выше - условие, позволяющее исключить из дальнейшего рассмотрения условные обозначения папок, находящихся на один и два уровня выше текущей
*/
 {
   $path = $nach_dir . "/" . $uroven;
//условие, формирующее полное имя проверяемого объекта

 if(!is_dir( $path ))//Условие, проверяющее является ли проверяемый объект файлом
   {
      $file_name=$path;//переменная $file_name, значение которой представляет собой значение переменной
//$path, урезанное на 2 символа (./) в начале
      $extention = pathinfo($file_name, PATHINFO_EXTENSION);
//определение расширения файла
         $date_file=date(filemtime($file_name));//определение даты создания (изменения) файла, которая будет отображаться
//в формате "d.m.Y"
            {
              $date_file=date('d.m.Y',filemtime($file_name));
              $filename="otchet_files.txt";
              $file_name=str_replace('////', '/', $file_name);
              $file_name=str_replace('///', '/', $file_name);
              $file_name=str_replace('//', '/', $file_name);
              $file_name=str_replace('\//', '/', $file_name);
              $file_name=str_replace('\/', '/', $file_name);
              $text= $file_name.";". $date_file.";\n";
              $file=fopen($filename,'a+');
              fwrite($file, $text);
              fclose($file);
            }
      }
    }
    else
     {
      files_names($path);
/*повторный запуск функции, который осуществляется с начальным значением аргумента равным значению $patch, если найденный объект каталог*/
      }
    }
  }
 closedir($directoriya);
}
$root= '/var/www/html/';
files_names($root); //Запуск функции на выполнение с начальным значением аргумента "./"
?>
```

6. Задание на управление правами доступа к файлам на web-сервере:

6.1. Используя панель управления XAMPP, запустите службу ftp-сервер FileZilla.

6.2. Используя панель администратора ftp-сервера, создайте группу пользователей Users

6.3. Предоставьте всем пользователям, входящим в состав указанной группы следующие права для работы с папкой XAMPP_HOME\htdocs:

a.Чтение файлов

b.Создание каталогов

c.Чтение содержания каталога

6.4. Используя панель администратора ftp-сервера, создайте пользователя User1 и User2, являющихся членами созданной ранее группы.

Предоставьте пользователю User1 следующие права для работы с папкой XAMPP_HOME\htdocs:

a.Чтение файлов

b.Запись файлов

c.Удаление файлов

d.Дописывание файлов

e.Создание каталогов

f.Удаление каталогов

g.Чтение содержания каталога

Последние 3 категории прав распространите и на подкаталоги папки.

6.5. Установите на свой домашний компьютер ftp-клиент FileZilla (http://filezilla.ru/).

6.6. Используя ftp-клиент Filezilla, подключитесь к серверу localhost с правами пользователя User1 и User2.

6.7. Экспериментальным путем определите разницу в правах этих пользователей. Какие права (для пользователя или группы) имеют приоритет?

6.8. Используя ftp-клиент FileZilla, попробуйте, используя числовую систему обозначения прав, задать права для папок и файлов, находящихся внутри папки XAMPP_HOME\htdocs. Результат поясните.

6.9. С помощью менеджера файлов, создайте внутри папки XAMPP_HOME\htdocs папку test.

6.10. Используя ftp-клиент FileZilla, с помощью прав пользователя User1 загрузите в эту папку несколько ранее созданных php-файлов. Какие права, на ваш взгляд, назначены для этих файлов? Проверьте корректность интерпретации кода web-сервером, используя в качестве url-адреса следующий адрес: http://localhost/test/file_name

Материалы текущего контроля успеваемости по теме 8.

Вопросы для контроля успеваемости в форме опроса

1. Перечислите основные элементы управления HTML-формы. Какую роль они играют в работе пользователя Интернетом?
2. Какие тэги или атрибуты задают URL скрипта-обработчика и метод обработки?
3. С помощью какого элемента управления HTML-формы инициализируется отправка её данных на HTTP-сервер?
4. Для чего могу использоваться скрытые поля формы?
5. Как организуется аутентификация пользователя в скрипте PHP?

Практические задания:

- Создать веб-форму, определяющую тип браузера;

- Разработать веб-форму, реализующие варианты перенаправления браузера указанную страницу;
- Реализовать организацию данных, введённых в различные поля, в массивы;
- Ознакомиться с методами проверки числовых и строковых данных, вводимых в поля на соответствие определённым критериям и реализовать их для веб-форм;
- Разработать веб-форму, осуществляющую как удаление тегов HTML, так и их кодирование;

Материалы текущего контроля успеваемости по теме 9.

Вопросы для контроля успеваемости в форме опроса

1. Опишите процедуру установки и настройки СУБД MySQL.
2. Как настроить интерпретатор PHP на работу с базами данных СУБД MySQL?
3. Какие кроме XAMPP существуют кроссплатформенные сборки для создания и тестирования полноценных веб-приложений на PHP?
4. Состав сборки XAMPP.
5. Опишите программные инструменты, обеспечивающие автоматизацию работы с базами данных в СУБД MySQL.
6. Сформулируйте принцип процедурного подхода к работе с базами данных в скриптах PHP.
7. Чем объектно-ориентированный подход к работе с базами данных в PHP отличается от процедурного?

Практические задания:

- Реализовать объектно-ориентированный интерфейс работы с базой данных в проекте PHP.

Материалы текущего контроля успеваемости по теме 10.

Вопросы для контроля успеваемости в форме опроса

1. Перечислите наиболее серьёзные угрозы безопасности веб-приложению.
2. Характеристики пользователя «root». Каким требованиям должен удовлетворять пароль для этого пользователя?
3. Как реализовать простой контроль доступа к веб-приложению с помощью HTML-формы и сценария PHP?
4. Каким должен быть файл, используемый для аутентификации пользователя? Какой недостаток присущ этому способу?
5. Как использовать базу данных для аутентификации пользователя веб-приложения?
6. Как проводится шифрование паролей в PHP?
7. Что такое базовая аутентификация протокола HTTP? Как её использовать в PHP?

Практические задания:

Ознакомится с конфигурационными настройками (параметрами) Apache сервера в файле httpd.conf и материалами по обеспечению его безопасности по адресу ://httpd.appache.org/docs-project и составить отчёт по их применению в виде презентации.

Материалы текущего контроля успеваемости по теме 11.

Вопросы для контроля успеваемости в форме опроса

5. Какие функции используются для работы с датами в PHP?

6. Что такое «метка времени»? Как можно с её помощью получить дату и время в требуемом виде?

7. Как можно определить время работы скрипта PHP?

8. Какая библиотека необходима для реализации функций для работы с изображениями в PHP?

9. Являются ли изображения, создаваемые в скрипте PHP статическими?

10. Дайте характеристику и особенности использования форматов JPEG, PNG и GIF.

11. Перечислите совокупность и последовательность шагов создания изображения в скрипте PHP.

12. Куда выводится изображение, созданное в скрипте PHP?

13. Как вставлять изображение PHP в различные HTML-страницы?

Практические задания:

Разработать скрипт, позволяющий отобразить в виде прямоугольных диаграмм количество проголосовавших за трёх кандидатов на выборах, хранящейся в соответствующей базе данных

Материалы текущего контроля успеваемости по теме 12.

Вопросы для контроля успеваемости в форме опроса

1. Почему протокол HTTP называют «протоколом без состояния»?
2. Можно ли в течение одного сеанса связи в интернете с веб-сайтом установить, что запросы на сервер поступают от одного пользователя?
3. В чём заключается смысл (идея) управления сеансами связи с веб-сайтом?
4. Для чего используются идентификатор сеанса и cookie-набор? В чём разница в их функциональности?
5. Перечислите этапы использования сеанса.
6. Функции PHP по управлению идентификатором сеансов?
7. Функции и методы работы с cookie в PHP?

Практические задания:

Разработать скрипт, осуществляющий аутентификацию пользователя средствами управления сеансами.

Материалы текущего контроля успеваемости по теме 13.

Вопросы для контроля успеваемости в форме опроса

1. Что Вы понимаете под термином CMS? Перечислите и охарактеризуйте основные CMS, существующие в настоящее время на рынке.
2. Опишите требования к системному программному обеспечению, необходимому для установки Joomla 3.4
3. Опишите основные этапы установки Joomla 3.4.
4. Что Вы понимаете под терминами «Клиентская часть» и «Административная часть» сайта?
5. Понятие расширения в CMS Joomla. В чем отличие между компонентом, модулем и плагином. Проблема совместимости расширений.
6. Охарактеризуйте файловую структуру CMS Joomla.
7. Охарактеризуйте структуру базы данных CMS Joomla

Практические задания:

- Выбрать систему бесплатного хостинга и установить на него CMS Joomla;

Материалы текущего контроля успеваемости по теме 14.

Вопросы для контроля успеваемости в форме опроса

1. Опишите процесс создания сайта в административной панели CMS Joomla.
2. Что Вы понимаете под термином «компонент»? Опишите процесс работы с компонентом Joomla: установку, настройку прав, подключение к сайту, удаление.
3. Что Вы понимаете под термином «модуль»? Опишите процесс работы с модулем Joomla: установку, настройку прав, подключение к сайту, удаление.
4. Что Вы понимаете под термином «плагин»? Опишите процесс работы с плагином Joomla: установку, настройку прав, подключение к сайту, удаление.
5. Что такое «сайт визитка». Опишите расширения, которые целесообразно использовать при создании сайта визитки.
6. Что такое «корпоративный сайт». Опишите расширения, которые целесообразно использовать при создании корпоративного сайта.
7. Что такое «интернет магазин». Опишите расширения, которые целесообразно использовать при создании сайта интернет магазина.
8. Что Вы понимаете под термином «SEO оптимизация»? Опишите основные требования к сайту, облегчающие процесс его индексации поисковыми системам**.**

Практические задания:

- Составить структурированный список параметров конфигураций сайта (на XML);
- Создать на локальном веб-сервере сайт на основе шаблонов Joomla (по указанию преподавателя), реализовав соответствующие параметры по пользователям, дизайну и структуре;
- Разместить на сайте соответствующий выбранному шаблону вариант контента;
- Скопировать разработанный сайт на хостинг.

Материалы текущего контроля успеваемости по теме 15.

Вопросы для контроля успеваемости в форме опроса

1. Понятие API. Особенности API Joomla: основные классы и методы.
2. Что такое компонент? Опишите файловую структуру пакета установки типового компонента Joomla.
3. Что такое модуль? Опишите файловую структуру пакета установки типового модуля Joomla.
4. Что такое плагин? Опишите файловую структуру пакета установки типового плагина Joomla.
5. Что такое шаблон? Опишите файловую структуру пакета установки типового шаблона Joomla.

Практические задания:

- Создать расширение Joomla с функциональностью, заданной преподавателем.
- Установить созданное расширение на сайты локального и удалённого веб серверов.

5. Оценочные материалы промежуточной аттестации по дисциплине

5.1. Зачет и экзамен проводится с применением следующих методов (средств): проведение письменного зачета и устного экзамена, включающего теоретическую часть и

практико-ориентированное задание, требующее демонстрации уровня владения умениями/навыками.

5.2. Оценочные материалы промежуточной аттестации

Таблица 5.

Показатели и критерии оценивания

Компонент компетенции	Промежуточный индикатор оценивания	Критерий оценивания
ПКс-2.3 Использует приемы разработки и адаптации прикладных программных решений	ИПКс-2.3 Владеет приемами разработки и адаптации прикладных программных решений	Осуществляет разработку и адаптацию прикладных программных решений
ПКс-3.2 Способен владеть приемами описания и разработки проекта информационной системы по видам обеспечения	ИПКс-3.2 Владеет приемами описания и разработки проекта информационной системы по видам обеспечения	Обладает приемами описания и разработки проекта информационной системы по видам обеспечения

Перечень теоретических вопросов к зачету

1. Отличительные признаки веб-страниц, сайтов и веб-приложений;
2. Характеристики статических и динамических веб-страниц;
3. Механизм взаимодействия Web-сервера, браузера, многостраничного сайта и базы данных (необходимые определения данной предметной области, CGI интерфейс, протокол HTTP).
4. Программное обеспечение реализации разработки и функционирования web-приложения.
5. Взаимодействие PHP-кода и тегов HTML. Методы внедрения PHP-кода в HTML-документ.
6. Функции PHP: организация, назначение, классификация и примеры использования. Создание пользовательских функций. Организация хранения функций. Использование операторов require и include.
7. Операции сравнения и логические операции в языке PHP. Ссылки и их использование в PHP. Конкатенация строк.
8. Массивы в PHP. Особенности создания и описания. Ассоциативные массивы. Многомерные массивы. Автоглобальные массивы. Функции для работы с массивами.
9. Особенности процедуры вывода в PHP. Использование escape-последовательностей. Форматирование строк для печати (использование функций).
10. Строковые функции и регулярные выражения: описание и примеры использования.
11. Управляющие конструкции. Особенности PHP.
12. Типы данных в PHP. Преобразование типов данных. Специальные символы. Функции для работы с символами. Функции преобразования чисел. Функции для работы с датой и временем.
13. Создание и использование пользовательских функций в PHP.
14. Классификация ошибок и их обработка в PHP. Понятие переменных окружения и их назначение. Структура и методы передачи. Часто используемые переменные окружения.
15. Структура заголовка HTTP, общее назначение заголовка. Использование заголовка для перенаправления на другой URL - адрес. Понятие кэширования страниц и его запрет в заголовке HTTP.
16. Работа с файлами и каталогами. Области и границы применения файлов при создании динамических WEB – страниц. Функции для работы с файлами. Пример

использования файла: создание списка рассылки с возможностью добавления, изменения и удаления E-mail адресов. Организация загрузки файлов на сервер с помощью формы.

17. Работа с файлами и каталогами. Области и границы применения файлов при создании динамических WEB – страниц. Функции для работы с каталогами. Создание программы для просмотра каталога и файлов на диске.

18. Передача данных в PHP с помощью форм HTML и их обработка. Роль метода передачи данных на сервер и использование глобальных переменных в этом процессе. Кнопка Submit. Примеры для следующих элементов формы: текстовое поле, поле ввода пароля, скрытое поле, поле для ввода многостраничного текста, раскрывающийся список.

19. Передача данных в PHP с помощью форм HTML и их обработка. Роль метода передачи данных на сервер и использование глобальных переменных в этом процессе. Кнопка Submit. Примеры для следующих элементов формы: текстовое поле, флажок, переключатель.

Перечень практико-ориентированных (практических) заданий к зачету

ПКс-2.3 *- использует приемы разработки и адаптации прикладных программных решений:*

Разработать проект web-приложения автоматизирующей ведение файловой БД.

ПКс-3.2 - способен владеть приемами описания и разработки проекта информационной системы по видам обеспечения:

Разработать web-приложение, позволяющее провести опрос посетителей сайта.

Перечень теоретических вопросов к экзамену

1. История, современное состояние и перспективы развития веб технологий;
2. Состав и характеристика аппаратно-программных составляющих веб-технологий;
3. Протокол TCP/IP: назначение, особенности и сервисные приложения;
4. Протокол HTTP: назначение, структура, методы обеспечения взаимодействия клиента и сервера;
5. Основные инструментальные средства работы с СУБД MySQL;
6. Задачи веб-доступа к базе данных и способы их реализации;
7. Особенности объектно-ориентированного подхода с базами данных в PHP;
8. Базовые методы взаимодействия HTML-формы со скриптом PHP;
9. Угрозы безопасности веб-приложения и способы борьбы с ними;
10. Способы создания изображений на веб-страницах в PHP;
11. Понятие, назначение и управление сеансами в PHP;
12. Работа с cookie-набором в PHP;
13. Назначение и особенности использования скриптов JavaScript;
14. Особенности реализации объектно-ориентированной парадигмы в языке JavaScript;
15. Структура и основные управляющие конструкции JavaScript;
16. Особенности работы с массивами в JavaScript;
17. Объектные модели веб-документа, браузера и JavaScript;
18. Управление текстом, графикой и стилями в JavaScript.
19. Классификация задач взаимодействия базы данных MySQL со скриптом PHP и способы их реализации;
20. Особенности процедурного и объектно-ориентированного интерфейсов взаимодействия СУБД MySQL и PHP;
21. Проблемы безопасности веб-приложений и способы их решения;

22. Особенности форматов изображений в веб приложениях. Логика и реализация алгоритма представления их в PHP;

23. Задачи, решаемые сеансами и Cookie-наборами при работе веб приложений, методы их реализации в PHP;

24. Системы управления содержимым сайта: классификация, особенности области использования и современное состояние рынка;

25. CMS Joomla: история развития, архитектура и компоненты;

26. Задачи администрирования в CMS Joomla;

27. Понятие расширений в CMS Joomla: особенности создания, состав и использование шаблонов;

28. Задачи процесса установки и настройки сайта на основе Joomla.

Перечень практико-ориентированных (практических) заданий к экзамену

ПКс-2.3 *- использует приемы разработки и адаптации прикладных программных решений:*

Разработать проект web-приложения со следующим функционалом:

Наименование проекта	***Описание функциональности проекта***	***Минимально необходимый набор форм***
Гостевая книга	Для каждой записи (сообщения) вашей гостевой книги храните в базе данных ее код, имя автора записи, e-mail автора записи, ICQ автора записи, дату добавления записи, текст записи. Для каждого ответа на запись храните его код, текст и код записи, к которой относится этот ответ. Должна быть страница, на которой выводятся все сообщения. Обеспечьте возможность просмотра одного сообщения со всеми ответами на него, добавления сообщения, удаления сообщения, ответа на сообщение.	Формы: добавление сообщения, добавление ответа на сообщение. Обратите внимание, что на странице для ответа на сообщение выводится поле для ответа как textarea, в которое можно ввести текст, а остальные поля записи выводятся как текст (не для редактирования)

ПКс-3.2 - способен владеть приемами описания и разработки проекта информационной системы по видам обеспечения:

Разработать web-приложение со следующим функционалом:

Наименование проекта	***Описание функциональности проекта***	***Минимально необходимый набор форм***
Доска объявлений с категориями	Для каждого объявления храните в базе данных его код, автора, текст, дату добавления и код категории. Для каждой категории - ее код и название. Обеспечьте возможность добавления объявления, добавления категории, удаления объявления, удаления категории, просмотра списка всех категорий, просмотра списка всех объявлений в конкретной категории, просмотра конкретного объявления.	Формы: добавление объявления, добавление категории.

Шкала оценивания

Промежуточная аттестация проводится в форме экзамена, зачета без применения балльно-рейтинговой системы оценивания - посредством аттестационных испытаний в форме традиционного экзамена, традиционного зачета.

Результаты промежуточной аттестации оцениваются: экзамена - по традиционной балльной системе оценивания («отлично», «хорошо», «удовлетворительно», «неудовлетворительно»); зачета — по бинарной системе оценивания («зачтено», «не зачтено»), или традиционной балльной системе оценивания.

Наряду с оцениванием учебных достижений обучающихся по принятой образовательной организации системе оценок и использованием ее при оформлении официальных дипломов Академии о высшем образовании, вводят повышенный

(«Excellent» и «Good») и базовый («Very good» и «Satisfactory») уровень оценок «отлично» и «хорошо» для использования при оформлении ЕПД, согласно Положению об оформлении и выдаче ЕПД.

Таблица 6.

Соответствие академической оценке

Академическая оценка		Оценка по шкале ESTS	
зачтено	отлично	Passed	A (Excellent)
			B (Very Good)
	хорошо		C (Good)
			D (Satisfactory)
	удовлетворительно		E (Sufficient)
не зачтено	неудовлетворительно	-	-

Критерии оценивания в ходе промежуточной аттестации

Оценка **«отлично»** выставляется обучающемуся:

– если он глубоко и прочно усвоил программный материал, исчерпывающе, последовательно, четко и логически стройно излагает его на экзамене, умеет тесно увязывать теорию с практикой, свободно справляется с задачами, вопросами и другими видами знаний, причем не затрудняется с ответом при видоизменении заданий, правильно обосновывает принятое решение; учебные достижения и результаты текущего контроля успеваемости демонстрируют высокую степень сформированности соответствующих компетенций.

Оценка **«хорошо»** выставляется обучающемуся:

– если он твердо знает материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, правильно применяет теоретические положения при решении практических вопросов и задач, владеет необходимыми навыками и приемами их выполнения; учебные достижения и результаты текущего контроля успеваемости демонстрируют хорошую степень сформированности соответствующих компетенций.

Оценка **«удовлетворительно»** выставляется обучающемуся:

– если он имеет знания только основного материала, но не усвоил его деталей, допускает неточности, недостаточно правильные формулировки, нарушения логической последовательности в изложении программного материала, испытывает затруднения при выполнении практических работ; учебные достижения и результаты текущего контроля успеваемости демонстрируют удовлетворительную степень сформированности соответствующих компетенций.

Оценка **«неудовлетворительно»** выставляется обучающемуся:

– если он не знает значительной части программного материала, допускает существенные ошибки, неуверенно, с большими затруднениями выполняет практические работы; учебные достижения и результатами текущего контроля успеваемости демонстрируют низкую степень сформированности соответствующих компетенций.

Оценка «**зачтено**» выставляется обучающемуся:

– если он глубоко и прочно усвоил программный материал, исчерпывающе, последовательно, четко и логически стройно излагает его на зачете, умеет увязывать теорию с практикой, свободно справляется с задачами, вопросами и другими видами заданий, не затрудняется с ответом при видоизменении заданий, правильно обосновывает принятое решение; в ходе обучения демонстрирует полное понимание и использует полученные знания и умения;

– твердо знает материал, грамотно и по существу излагает его, правильно применяет теоретические положения при решении практических вопросов и задач; в ходе обучения демонстрирует понимание и использует полученные знания и умения;

– знает материал, излагает его, но допускает несущественные неточности в ответе на вопрос, в применении теоретических положений при решении практических вопросов и задач; в ходе обучения проявляет понимание и использует полученные знания и умения в выстраивании и реализации своего развития;

Оценка «**не зачтено**» выставляется обучающемуся:

– который не знает значительной части программного материала, допускает существенные ошибки, неуверенно, с большими затруднениями выполняет практические задания; учебные достижения демонстрируют низкую степень овладения программным материалом.

6. Методические материалы по освоению дисциплины с применением ЭК

Обучение по дисциплине предполагает изучение курса на аудиторных занятиях (лекции, практические) и самостоятельной работы студентов. Практические занятия предполагают их проведение в различных формах с целью выявления полученных знаний, умений, навыков и компетенций.

Подготовка к лекции студентами заключается в следующем:

– повторить материал предыдущей лекции, прочитав его повторно;

– ознакомиться с темой предстоящей лекции (в рабочей программе учебной дисциплины);

– ознакомиться с учебными материалами по данной теме в соответствии с предложенным списком литературы в рабочей программе учебной дисциплины или с электронными материалами, предложенными лектором;

– записать возможные вопросы, которые можно будет задать лектору.

Подготовка к практическим (семинарским) занятиям и лабораторным работам*:*

– внимательно прочитать материал лекций относящихся к данному занятию, ознакомиться с учебными материалами, включая электронные в соответствии с предложенным списком литературы в рабочей программе учебной дисциплины;

– выполнить задания, если они предусмотрены в письменной форме;

– понять, что для вас осталось неясными и постараться получить на них ответ заранее;

– готовиться к практическим/семинарским занятиям можно как индивидуально, так и в составе малой группы;

– рабочую программу учебной дисциплины необходимо использовать в качестве основного ориентира в организации обучения;

– электронная версия рабочей программы по дисциплине размещена на сайте Нижегородского института управления, к ней предоставлен авторизованный доступ.

Самостоятельная работа обучающегося:

Самостоятельная работа призвана закрепить теоретические знания и практические навыки, полученные на лекциях, практических и семинарских занятиях.

Самостоятельная работа проводится с целью:

- систематизации и закрепления полученных теоретических знаний и практических умений обучающихся;
- углубления и расширения теоретических знаний;
- формирования умений использовать справочную документацию и специальную литературу;

- развития познавательных способностей и активности: творческой инициативы, самостоятельности, ответственности и организованности;
- формирования самостоятельности мышления, способностей к саморазвитию, самосовершенствованию и самореализации;
- развития исследовательских умений.

Эффективность лекционных, семинарских и практических занятий по дисциплине во многом зависит от качества самостоятельной работы обучающихся, от их самоподготовки.

Часть времени, отведенного на самостоятельную работу должна использоваться на подготовку к аудиторным занятиям, другая часть на выполнение домашней работы, осмысление и оформление результатов практических занятий.

При подготовке к занятиям студенту полезно:

- изучить теоретический материал по данной теме (конспект занятия);
- ознакомиться с литературой, рекомендованной преподавателем;
- выполнить задания, предложенные преподавателем, к занятию;
- составить перечень вопросов, вызывающих затруднения, неясности или сомнения, обсудить их с преподавателем или на занятии;
- заниматься самостоятельным поиском дополнительной литературы по изучаемой теме.

Подготовка к экзамену (зачету). К экзамену (зачету) необходимо готовиться целенаправленно, регулярно, систематически и с первых дней обучения по данной дисциплине. В самом начале учебного курса познакомьтесь со следующей учебно-методической документацией:

- программой дисциплины;
- перечнем знаний и умений, которыми студент должен владеть;
- тематическими планами лекций, семинарских занятий;
- учебником, учебными пособиями по дисциплине, а также электронными ресурсами;
- перечнем и тематикой письменных работ, а также методическими рекомендациями по их выполнению;
- перечнем вопросов.

Систематическое выполнение всех видов заданий на лекциях, практических занятиях, а также самостоятельная работа позволит успешно освоить дисциплину и создать хорошую базу для сдачи экзамена (зачета)**.**

Вопросы для самостоятельной подготовки к занятиям

1. Отличительные признаки веб-страниц, сайтов и веб-приложений;
2. Характеристики статических и динамических веб-страниц;
3. Механизм взаимодействия Web-сервера, браузера, многостраничного сайта и базы данных (необходимые определения данной предметной области, CGI интерфейс, протокол HTTP).
4. Программное обеспечение реализации разработки и функционирования web-приложения.
5. Взаимодействие PHP-кода и тегов HTML. Методы внедрения PHP-кода в HTML-документ.
6. Функции PHP: организация, назначение, классификация и примеры использования. Создание пользовательских функций. Организация хранения функций. Использование операторов require и include.

7. Операции сравнения и логические операции в языке PHP. Ссылки и их использование в PHP. Конкатенация строк.

8. Массивы в PHP. Особенности создания и описания. Ассоциативные массивы. Многомерные массивы. Автоглобальные массивы. Функции для работы с массивами.

9. Особенности процедуры вывода в PHP. Использование escape-последовательностей. Форматирование строк для печати (использование функций).

10. Строковые функции и регулярные выражения: описание и примеры использования.

11. Управляющие конструкции. Особенности PHP.

12. Типы данных в PHP. Преобразование типов данных. Специальные символы. Функции для работы с символами. Функции преобразования чисел. Функции для работы с датой и временем.

13. Создание и использование пользовательских функций в PHP.

14. Классификация ошибок и их обработка в PHP. Понятие переменных окружения и их назначение. Структура и методы передачи. Часто используемые переменные окружения.

15. Структура заголовка HTTP, общее назначение заголовка. Использование заголовка для перенаправления на другой URL - адрес. Понятие кэширования страниц и его запрет в заголовке HTTP.

16. Работа с файлами и каталогами. Области и границы применения файлов при создании динамических WEB – страниц. Функции для работы с файлами. Пример использования файла: создание списка рассылки с возможностью добавления, изменения и удаления E-mail адресов. Организация загрузки файлов на сервер с помощью формы.

17. Работа с файлами и каталогами. Области и границы применения файлов при создании динамических WEB – страниц. Функции для работы с каталогами. Создание программы для просмотра каталога и файлов на диске.

18. Передача данных в PHP с помощью форм HTML и их обработка. Роль метода передачи данных на сервер и использование глобальных переменных в этом процессе. Кнопка Submit. Примеры для следующих элементов формы: текстовое поле, поле ввода пароля, скрытое поле, поле для ввода многостраничного текста, раскрывающийся список.

19. Передача данных в PHP с помощью форм HTML и их обработка. Роль метода передачи данных на сервер и использование глобальных переменных в этом процессе. Кнопка Submit. Примеры для следующих элементов формы: текстовое поле, флажок, переключатель.

20. История, современное состояние и перспективы развития веб технологий;

21. Состав и характеристика аппаратно-программных составляющих веб-технологий;

22. Протокол TCP/IP: назначение, особенности и сервисные приложения;

23. Протокол HTTP: назначение, структура, методы обеспечения взаимодействия клиента и сервера;

24. Основные инструментальные средства работы с СУБД MySQL;

25. Задачи веб-доступа к базе данных и способы их реализации;

26. Особенности объектно-ориентированного подхода с базами данных в PHP;

27. Базовые методы взаимодействия HTML-формы со скриптом PHP;

28. Угрозы безопасности веб-приложения и способы борьбы с ними;

29. Способы создания изображений на веб-страницах в PHP;

30. Понятие, назначение и управление сеансами в PHP;

31. Работа с cookie-набором в PHP;

32. Назначение и особенности использования скриптов JavaScript;

33. Особенности реализации объектно-ориентированной парадигмы в языке JavaScript;
34. Структура и основные управляющие конструкции JavaScript;
35. Особенности работы с массивами в JavaScript;
36. Объектные модели веб-документа, браузера и JavaScript;
37. Управление текстом, графикой и стилями в JavaScript.
38. Классификация задач взаимодействия базы данных MySQL со скриптом PHP и способы их реализации;
39. Особенности процедурного и объектно-ориентированного интерфейсов взаимодействия СУБД MySQL и PHP;
40. Проблемы безопасности веб-приложений и способы их решения;
41. Особенности форматов изображений в веб приложениях. Логика и реализация алгоритма представления их в PHP;
42. Задачи, решаемые сеансами и Cookie-наборами при работе веб приложений, методы их реализации в PHP;
43. Системы управления содержимым сайта: классификация, особенности области использования и современное состояние рынка;
44. CMS Joomla: история развития, архитектура и компоненты;
45. Задачи администрирования в CMS Joomla;
46. Понятие расширений в CMS Joomla: особенности создания, состав и использование шаблонов;
47. Задачи процесса установки и настройки сайта на основе Joomla.

7. Учебная литература и ресурсы информационно-телекоммуникационной сети Интернет

7.1. Основная литература.

1. Тузовский, А. Ф. Проектирование и разработка web-приложений : учебное пособие для вузов / А. Ф. Тузовский. — Москва : Издательство Юрайт, 2023. — 218 с. — (Высшее образование). — ISBN 978-5-534-00515-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/512113 (дата обращения: 18.03.2024). — Режим доступа: по подписке.

2. Кисленко Н.П. Интернет-программирование на PHP : учебное пособие / Кисленко Н.П.. — Новосибирск : Новосибирский государственный архитектурно-строительный университет (Сибстрин), ЭБС АСВ, 2015. — 177 с. — ISBN 978-5-7795-0745-5. — Текст : электронный // IPR SMART : [сайт]. — URL: https://www.iprbookshop.ru/68769.html (дата обращения: 18.03.2024). — Режим доступа: для авторизир. пользователей. - DOI: https://doi.org/10.23682/68769

3. Седова, Я. А. Разработка расширений для CMS Joomla : учебное пособие / Я. А. Седова. — 2-е изд. — Москва : ИНТУИТ, 2016. — 250 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/100471 (дата обращения: 18.03.2024). — Режим доступа: для авториз. пользователей.

4. Соколова, В. В. Вычислительная техника и информационные технологии. Разработка мобильных приложений : учебное пособие для вузов / В. В. Соколова. — Москва : Издательство Юрайт, 2024. — 160 с. — (Высшее образование). — ISBN 978-5-534-16302-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/537272 (дата обращения: 26.04.2024). — Режим доступа: для авториз. пользователей.

7.2. Дополнительная литература.

1. Баранов Р.Д. Практические аспекты разработки веб-ресурсов : учебное пособие / Баранов Р.Д., Иноземцева С.А., Рябова А.А.. — Саратов : Вузовское образование, 2018. — 121 c. — ISBN 978-5-4487-0263-1. — Текст : электронный // IPR SMART : [сайт]. — URL: https://www.iprbookshop.ru/75692.html (дата обращения: 18.03.2023). — Режим доступа: для авторизир. пользователей. - DOI: https://doi.org/10.23682/75692

2. Сычев А.В. Перспективные технологии и языки веб-разработки : практикум / Сычев А.В.. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Эр Медиа, 2019. — 493 c. — ISBN 978-5-4486-0507-9. — Текст : электронный // IPR SMART : [сайт]. — URL: https://www.iprbookshop.ru/79730.html (дата обращения: 18.03.2023). — Режим доступа: для авторизир. пользователей.

3. Полуэктова, Н. Р. Разработка веб-приложений : учебное пособие для вузов / Н. Р. Полуэктова. — 2-е изд. — Москва : Издательство Юрайт, 2024. — 204 с. — (Высшее образование). — ISBN 978-5-534-18645-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/545238 (дата обращения: 26.04.2024). — Режим доступа: для авторизир. пользователей.

4. Сысолетин, Е. Г. Разработка интернет-приложений : учебное пособие для вузов / Е. Г. Сысолетин, С. Д. Ростунцев ; под научной редакцией Л. Г. Доросинского. — Москва : Издательство Юрайт, 2023. — 90 с. — (Высшее образование). — ISBN 978-5-9916-9975-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/514303 (дата обращения: 26.04.2024). — Режим доступа: для авторизир. пользователей.

7.3 Учебно-методическое обеспечение самостоятельной работы.

1. Электронные информационно-образовательные ресурсы Нижегородского института управления – филиала РАНХиГС - http://sdo.niu.ranepa.ru/

2. Тузовский, А. Ф. Проектирование и разработка web-приложений : учебное пособие для среднего профессионального образования / А. Ф. Тузовский. — Москва : Издательство Юрайт, 2022. — 218 с. — (Профессиональное образование). — ISBN 978-5-534-10017-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/495109 (дата обращения: 26.04.2024). — Режим доступа: по подписке.

7.4. Нормативные правовые документы.

1. Федеральный закон от 29 декабря 2012 г. № 273-ФЗ "Об образовании в Российской Федерации"
2. Федеральный закон от 27 июля 2006 г. N 149-ФЗ "Об информации, информационных технологиях и о защите информации" (с изменениями и дополнениями)
3. Федеральный закон от 27 июля 2006 г. N 152-ФЗ "О персональных данных" (с изменениями и дополнениями)
4. Приказ Министерства образования и науки РФ от 05.04.2017 №301 «Об утверждении Порядка организации и осуществления образовательной деятельности по образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры»
5. Образовательный стандарт ФГБОУ ВО «Российской академии народного хозяйства и государственной службы при Президенте Российской Федерации», утвержденный приказом ректора Академии от 07.09.2023 № 01-24607 и разработанного на основе федерального государственного образовательного стандарта высшего образования

– бакалавриат по направлению подготовки 09.03.03 Прикладная информатика, утвержденного приказом Министерства образования и науки Российской Федерации от 19.09.2017 №922 (зарегистрировано в Министерстве юстиции Российской Федерации 12.10.2017, регистрационный номер № 48531) и приказа Минобрнауки России от 26 ноября 2020 г. № 1456 «О внесении изменений в федеральные государственные образовательные стандарты высшего образования» (зарегистрировано в Минюсте России 27.05.2021 г., регистрационный № 63650).

7.5. Интернет-ресурсы.

1. Joomla.ru - Все о Joomla! на русском - просто и понятно - https://joomla.ru/
2. Joomla! The CMS Trusted By Millions for their Websites - https://www.joomla.org/
3. Joomla! по-русски - https://joomlaportal.ru/
4. MySQL - http://www.mysql.ru/docs/mysql-man-4.0-ru/introduction.html
5. PHP, MySQL и другие веб-технологии http://www.php.su
6. PHP.RU – Сообщество PHP-Программистов - https://php.ru/
7. XAMPP Apache + MySQL + PHP + Perl - https://www.thegeekstuff.com/2011/01/xampp-install/
8. Интегрированная среда разработки NetBeans https://netbeans.apache.org/
9. Форум русской поддержки Joomla! CMS - https://joomlaforum.ru/
10. Основы работы в OpenOffice. Web: http://www.intuit.ru/department/office/openofficebasics .
11. Основы работы с Яндекс. Web: http://www.intuit.ru/department/internet/bwyandex/.
12. Работа в Интернет с Microsoft Internet Explorer. Web: http://www.intuit.ru/department/internet/internetcert .
13. Современные офисные приложения. Web: https://trashexpert.ru/software/ms-office/the-best-office-suites/ .

8. Материально-техническая база, информационные технологии, программное обеспечение и информационные справочные системы

8.1. Материально-техническая база

Перечень материально-технического обеспечения:

1. Учебные аудитории, оборудованные для проведения учебных занятий лекционного и семинарского типа, групповых и индивидуальных консультаций, коллоквиумов, мероприятий текущего контроля и промежуточной аттестации, в том числе мультимедийным оборудованием для демонстрации электронных презентаций и аудио- и видеоматериалов.
2. Компьютерные классы для выполнения групповых тестовых и иных заданий, а также для самостоятельной работы обучающихся оснащенные компьютерной техникой и обеспечением доступа к сети «Интернет» и доступа в электронную информационно-образовательную среду организации.
3. Специализированные аудитории и лаборатории.
4. Библиотека с обеспечением печатными изданиями или электронно-библиотечная система обеспечивающая доступ к электронным изданиям (электронная библиотека).
5. Читальный зал.
6. Технические средства обучения: персональные компьютеры; компьютерные проекторы; звуковые динамики; программные средства, обеспечивающие просмотр

видеофайлов в форматах AVI, MPEG-4, DivX, RMVB, WMV и др.

8.2. Информационные технологии, программное обеспечение:

При осуществлении образовательного процесса по дисциплине используется информационные технологии и программное обеспечение:

1. Современная операционная система.
2. Kaspersky Endpoint Security (или аналог).
3. Средство просмотра файлов формата pdf.
4. Современные офисные средства (текстовые и табличные редакторы, средства работы с презентационными материалами и т.д.).
5. Архиватор 7-Zip.
6. Система дистанционного обучения.
7. Автоматизированная библиотечная система.
8. СУБД PostgreSQL.
9. Lucidchart.
10. Eclipse
11. Система ADVANTA для управления проектами.
12. 1С: Предприятие 8.2

8.3. Информационные справочные системы:

1. https://www.urait.ru –Электронно-библиотечная система [ЭБС] Юрайт;
2. http://www.iprbookshop.ru – Электронно-библиотечная система [ЭБС] «IPR SMART» (ранее – IPRBooks)
3. https://e.lanbook.com - Электронно-библиотечная система [ЭБС] «Лань».
4. https://znanium.com - Электронно-библиотечная система [ЭБС] «Znanium.com».
5. https://www.book.ru - Электронно-библиотечная система [ЭБС] «Book.ru».
6. https://ibooks.ru - Электронно-библиотечная система [ЭБС] «ibooks.ru».
7. https://www.ezproxy.ranepa.ru:2443/login?url=https://dlib.eastview.com/ – East View. Полные тексты российских научных и практических журналов, а так же газет центральной прессы России.
8. https://www.ezproxy.ranepa.ru:2443/login?url=https://grebennikon.ru/ - Полные тексты 38 научно-практических журналов по маркетингу, менеджменту, финансам и управлению персоналом ИД «Гребенников»
9. http://www.consultant.ru/ - Справочно-правовая система «Консультант».
10. http://www.garant.ru/ Справочно-правовая система «Гарант».