	Федеральное государственное бюджетное образовательное учреждение высшего образования «Башкирский государственный аграрный университет»	Приложение к ОПОП ВО
		Рабочая программа дисциплины


РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
Б1.В.ДВ.06.01 WEB-ПРОГРАММИРОВАНИЕ

Направление подготовки
09.03.03 Прикладная информатика

Профиль подготовки
Прикладная информатика цифровой экономики


Квалификация (степень) выпускника
Бакалавр

Уфа 2021


Составитель:  ст. преподаватель Прокофьева С.В.

Программа составлена в соответствии с Федеральным государственным образовательным стандартом высшего образования по направлению подготовки 09.03.03 Прикладная информатика, утвержден приказом Министерства образования и науки Российской Федерации от 19.09.2017 № 922.

Рабочая программа обсуждена и одобрена на заседании кафедры информатики и информационных технологий «25» марта 2021 г. (протокол № 9).

Зав. кафедрой информатики и информационных технологий доцент, д-р техн. наук  Беляева А.С.

Рассмотрена и одобрена на заседании методической комиссии экономического факультета «25» марта 2021 г. (протокол № 8).

Председатель методической комиссии экономического факультета доцент, канд. экон. наук  Тукаева Ф.А.

Согласовано:
Руководитель ОПОП ВО  доцент, канд. физ.-мат. наук Шамсутдинова Т. М.

1 Перечень планируемых результатов обучения по дисциплине Web-программирование, соотнесенных с планируемыми результатами освоения образовательной программы

В результате освоения ОПОП ВО бакалавриата обучающийся должен овладеть следующими результатами обучения по дисциплине (модулю):

<i>Код и наименование компетенции</i>	<i>Код и наименование индикаторов достижения компетенции</i>	<i>Планируемые результаты обучения</i>
ПК-6 Выполнение работ по созданию (модификации) и сопровождению ИС, автоматизирующих задачи организационного управления и бизнес-процессы	ПК-6.1 Разработка прототипов ИС на базе типовой ИС	Знания: ПК-6.1/Зн1 языков программирования и работы с базами данных; инструменты и методов модульного тестирования, инструментов и методов тестирования нефункциональных и функциональных характеристик ИС; инструментов и методов прототипирования пользовательского интерфейса; Умения: ПК-6.1/Ум1 кодировать на языках программирования в процессе разработки прототипов ИС на базе типовой ИС; тестировать результаты прототипирования; проводить презентации, проводить переговоры в ходе согласования пользовательского интерфейса с заказчиком; Навыки: ПК-6.1/Нв1 разработки прототипа ИС на базе типовой ИС в соответствии с требованиями; тестирования прототипа ИС на корректность архитектурных решений, проведения анализа результатов тестирования; принятия решения о пригодности архитектуры, согласования пользовательского интерфейса с заказчиком.
	ПК-6.2 Установка и настройка системного и прикладного ПО, необходимого для функционирования ИС	Знания: ПК-6.2/Зн1 основ современных операционных систем; основ современных систем управления базами данных; основ администрирования СУБД; устройства и функционирования современных ИС, коммуникационного оборудования, сетевых протоколов, основ системного администрирования Умения: ПК-6.2/Ум1 Устанавливать и настраивать операционные системы, СУБД, прикладное ПО Навыки: ПК-6.2/Нв1 установки операционных систем, СУБД, прикладного ПО, необходимых для функционирования ИС; настройки операционных систем, СУБД, прикладного ПО для оптимального функционирования ИС.

2 Место дисциплины в структуре ОПОП ВО

Дисциплина «Web-программирование» относится к дисциплинам по выбору блока «Б1. Дисциплины (модули)», части, формируемой участниками образовательных отношений учебного плана направления 09.03.03 Прикладная информатика. Базируется на компетенциях, полученных при изучении дисциплин Б1.В.ДВ.04.01 Управление данными, Б1.В.ДВ.04.02 Базы данных корпоративных информационных систем, Б1.В.ДВ.05.01 Корпоративные информационные системы, Б1.В.ДВ.05.02 Комплексная автоматизация предприятий.

Для изучения дисциплины обучающиеся должны владеть такими понятиями как компьютерные сети, телекоммуникации, сетевые протоколы, базы данных; должны иметь представление об основных средствах и языках разработки интернет-ресурсов.

Дисциплина изучается на 4 курсе в 7 семестре на очном и заочном обучении.

Последующие дисциплины (практики) по связям компетенций Б1.В.09 Администрирование корпоративных информационных систем, Б1.В.ДВ.07.01 Обмен данными в корпоративных информационных системах, Б1.В.ДВ.07.02 Основы клиент-серверного программирования,

3 Объем дисциплины и виды учебной работы

Общая трудоемкость (объем) дисциплины составляет 4 зачетные единицы (ЗЕ)

3.1 Очное обучение (срок обучения: 4 года)

Вид учебной работы	Всего часов	Распределение по семестрам
		7
Контактная работа, всего	58	58
в т.ч.: занятия лекционного типа (лекции) (Л)	24	24
занятия семинарского типа (лабораторные работы) (ЛР)	34	34
в т.ч. направленные на практическую подготовку (ПРП)	6	6
Самостоятельная работа обучающегося (СРО), всего	86	86
в т.ч.: подготовка к лабораторным работам (ЛР)	34	34
самостоятельное изучение теоретического материала (СИТМ)	52	52
Вид промежуточной аттестации (зачет, экзамен)	зачет	зачет
Общая трудоемкость дисциплины	часы	144
	зачетные единицы	4

3.2 Заочное обучение (срок обучения: 4 года 6 месяцев)

Вид учебной работы	Всего часов	Распределение по семестрам
		7
Контактная работа, всего	14	14
в т.ч.: занятия лекционного типа (лекции) (Л)	4	4
занятия семинарского типа (лабораторные работы) (ЛР)	10	10
в т.ч. направленные на практическую подготовку (ПРП)	-	-
Самостоятельная работа обучающегося (СРО), всего	130	130
в т.ч.: подготовка к лабораторным работам (ЛР)	10	10
самостоятельное изучение теоретического материала (СИТМ)	120	120
Вид промежуточной аттестации (зачет, экзамен)	зачет	зачет
Общая трудоемкость дисциплины	часы	144
	зачетные единицы	4

4 Содержание дисциплины**4.1 Разделы дисциплины и виды занятий**

№ п/п	Наименование модуля дисциплины	Очное обучение			Заочное обучение		
		Л	ЛР/ПРП	СРО	Л	ЛР/ПРП	СРО
1	Создание и публикация информационных ресурсов WWW в Internet	16	16	52	2	2	66
2	Взаимодействие серверных сценариев с СУБД	8	18/6	34	2	8	64
Итого:		24	34/6	86	4	10	130

4.2 Содержание разделов дисциплины

№ п/п	Наименование модуля (раздела) дисциплины	Содержание раздела
1	Модуль 1 Создание и публикация информационных ресурсов WWW в Internet	<p>Компьютерные телекоммуникационные сети. Глобальные гипертекстовые структуры: WWW. Клиент-серверные технологии web. Обзор средств создания web-приложений. Рабочая среда разработки.</p> <p>Особенности структурированного языка гипертекстовой разметки HTML как основы создания WEB – страниц. Создание интерактивных WEB-страниц для передачи информации на сервер (формы): параметры формы, элементы формы.</p> <p>Программирование на стороне сервера. Формирование ответов на запрос пользователя. Обзор языков сценариев (JavaScript, PHP, Perl и др.).</p> <p>Основы PHP. Основные элементы языка. Основные операторы языка PHP: оператор присваивания, операторы вывода информации на web-страницу, оператор ветвления, операторы цикла. Работа с массивами. Обработка строк. Создание и обработка форм. Типы запросов POST и GET. Использование параметров URL. Сессии и cookie. Работа с файлами.</p>
2	Модуль 2 Взаимодействие серверных сценариев с СУБД	<p>Разработка WEB - интерфейса доступа к серверным базам данных, серверная обработка информации с выдачей результатов пользователю. Обзор серверных СУБД.</p> <p>Серверная СУБД MySQL. Язык обработки данных SQL. Функции PHP для работы с базами данных MySQL. Создание базы данных и таблиц. Выборка данных из таблиц MySQL. Добавление, удаление, редактирование данных в таблицах MySQL. Использование программных средств идентификации пользователей при работе с базами данных.</p>

5 Тематика контактной работы

5.1 Занятия лекционного типа (лекции)

№ п/п	№ модуля	Наименование лекционных занятий	Объем, часы	
			Очное обучение	Заочное обучение
1	1	Характеристики и структуры сетевых технологий. Средства создания web-приложений.	4	
2	1	Основы языка PHP. Основные элементы языка.	4	1
3	1	Основные операторы языка PHP.	4	1
4	1	Функции PHP. Работа с массивами в PHP.	4	
5	2	Серверные СУБД. Язык обработки данных SQL.	4	1
6	2	Функции PHP для работы с базами данных	4	1
Итого:			24	4

5.2 Занятия семинарского типа (практические занятия) не предусмотрены.

5.3 Занятия семинарского типа (лабораторные работы)

№	№ модуля	Наименование лабораторных работ	Объем, часы
---	----------	---------------------------------	-------------

п/п	ля		Очное обучение	Заочное обучение
1	1	Основы PHP. Создание простых сценариев.	4	1
2	1	Основы PHP. Обработка форм. Формирование ответа клиенту.	4	1
3	1	Основы PHP. Работа с массивами.	4	-
4	1	Основы PHP. Работа с файлами	4	-
5	2	Основы PHP. Работа с СУБД MySQL. Создание базы данных и таблиц.	4	2
6	2	Основы PHP. Выборка данных из таблиц MySQL.	4	2
7	2	Основы PHP. Добавление, удаление, редактирование данных в таблицах MySQL (ПП).	6	4
8	2	Основы PHP. Авторизация пользователя.	4	-
Итого:			34	10

6 Самостоятельная работа обучающихся

6.1 Очное обучение

№ п/п	№ модуля (раздела)	Виды самостоятельной работы	Название (содержание) работы	Объем, часы
1	2	3	4	5
1	1	СИТМ	Характеристики и структуры сетевых технологий. Средства создания web-приложений.	12
2	1	СИТМ	Основы языка PHP. Основные элементы языка.	8
3	1	СИТМ	Основные операторы языка PHP.	8
4	1	СИТМ	Функции PHP. Работа с массивами в PHP.	8
5	1	Подготовка к лабораторным работам	Основы PHP. Создание простых сценариев.	4
6	1	Подготовка к лабораторным работам	Язык PHP. Обработка форм. Формирование ответа клиенту.	4
7	1	Подготовка к лабораторным работам	Основы PHP. Работа с массивами.	4
8	1	Подготовка к лабораторным работам	Основы PHP. Работа с файлами	4
9	2	СИТМ	Серверные СУБД. Язык обработки данных SQL.	8
10	2	СИТМ	Функции PHP для работы с базами данных.	8
11	2	Подготовка к лабораторным работам	Язык PHP. Работа с базой данных MySQL. Создание базы данных и таблиц.	4
12	2	Подготовка к лабораторным работам	Основы PHP. Выборка данных из таблиц MySQL.	4
13	2	Подготовка к лабораторным работам	Добавление, удаление, редактирование	6

		рабочим работам	данных в таблицах MySQL.	
14	2	Подготовка к лабораторным работам	Авторизация пользователя.	4
		Всего:		86

6.2 Заочное обучение

№ п/п	№ модуля (раздела)	Виды самостоятельной работы	Название (содержание) работы	Объем, Часы
1	2	3	4	5
1	1	СИТМ	Характеристики и структуры сетевых технологий. Средства создания web-приложений.	20
2	1	СИТМ	Основы языка PHP. Основные элементы языка.	20
3	1	СИТМ	Основные операторы языка PHP.	20
4	1	СИТМ	Функции PHP. Работа с массивами в PHP.	20
5	1	Подготовка к лабораторным работам	Основы PHP. Создание простых сценариев.	1
6	1	Подготовка к лабораторным работам	Язык PHP. Обработка форм. Формирование ответа клиенту.	1
7	2	СИТМ	Серверные СУБД. Язык обработки данных SQL.	20
8	2	СИТМ	Функции PHP для работы с базами данных.	20
9	2	Подготовка к лабораторным работам	Язык PHP. Работа с базой данных MySQL. Создание базы данных и таблиц.	2
10	2	Подготовка к лабораторным работам	Основы PHP. Выборка данных из таблиц MySQL.	2
11	2	Подготовка к лабораторным работам	Добавление, удаление, редактирование данных в таблицах MySQL.	4
		Всего:		130

7 Образовательные технологии

Реализация у обучающихся навыков командной работы, межличностной коммуникации, принятия решений, лидерских качеств предусмотрено использование в учебном процессе проведение занятий в виде анализа ситуации и групповых дискуссий.

№ п/п	№ модуля (раздела)	Наименование темы	Вид учебного занятия	Активные и интерактивные формы проведения обучения
1	Модуль 1	Основы PHP. Создание простых сценариев.	Лабораторные работы	Проведение лабораторных занятий с элементами групповой дискуссии
2	Модуль 2	Основы PHP. Работа с СУБД MySQL. Создание базы данных и таблиц.	Лабораторные работы	Анализ ситуации

8 Оценочные материалы для проведения промежуточной аттестации обучающихся по дисциплине

Перечень компетенций с указанием этапов их формирования; описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания; типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы; методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций) представлены в **Приложение 1** к рабочей программе дисциплины (модуля) оценочные материалы по учебной дисциплине в виде «**Фонда оценочных средств**».

9 Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины (модуля)

а) Основная литература

1 Лисьев Г.А. Программное обеспечение компьютерных сетей и web-серверов [Электронный ресурс] : учебное пособие / Г.А. Лисьев, П.Ю. Романов, Ю.И. Аскерко. - М.: Инфра-М, 2021. - 145 с. – Режим доступа: <https://znanium.com/read?id=365037>

2 Евсеев, Д. А. Web-дизайн в примерах и задачах [Текст] : учебное пособие для студентов высш. учеб. заведений, обуч. по специальности "Прикладная информатика" и др. эконом. специальностям / Д. А. Евсеев, В. В. Трофимов ; под ред. В. В. Трофимова ; Санкт-Петербургский гос. ун-т экономики и финансов. - Москва : Кнорус, 2014. - 263 с.

б) Дополнительная литература

1 Винарский Я.С. Web-аппликации в интернет-маркетинге: проектирование, создание и применение [Электронный ресурс] : практическое пособие / Я.С. Винарский, Р.Д. Гутгарц Г.А. - М.: Инфра-М, 2019. - 269 с. – Режим доступа: <https://znanium.com/read?id=359301>

2 Емельянова, Н. З. Основы построения автоматизированных информационных систем [Текст] : учеб. пособие / Н. З. Емельянова, Т. Л. Партыка, И. И. Попов. - М. : ФОРУМ : ИНФРА-М, 2007. - 415 с.

3 Нельзина, О. Г. Информационные системы для электронной коммерции [Текст] : учеб. пособие / О. Г. Нельзина. - Ростов н/Д: Феникс, 2008. – 271 с.

4 Хорошилов, А. В. Управление информационными ресурсами [Текст] : учебник для студ. вузов, обуч. по напр. подготовки и спец. "Прикладная информатика (по областям)" / А. В. Хорошилов, С. Н. Селетков, Н. В. Днепровская ; под ред. А. В. Хорошилова. - М.: Финансы и статистика, 2007. – 270 с.

5 Бройдо, В. Л. Вычислительные системы, сети и телекоммуникации [Текст] : учеб. пособие для студентов вузов, обуч. по направлениям "Прикладная информатика" и "Информационные системы в экономике" / В. Л. Бройдо. - 2-е изд. - М. [и др.] : Питер, 2006. - 702 с.

6 Васильев, Г. А. Электронный бизнес и реклама в Интернете [Текст] : учебное пособие для студ. вузов / Г. А. Васильев, Д. А. Забегалин. - М. : ЮНИТИ-ДАНА, 2008. - 181 с.

7 Вычислительные системы, сети и телекоммуникации: учеб. пособие. / А.П. Пятибратов, Л.П. Гудыно, А.А. Киреченко; Под ред. А.П. Пятибратов. - М.: Финансы и статистика, 2005. – 560 с.

8 Журнал «Системы управления»

10 Профессиональные базы данных и ресурсы «Интернет», к которым обеспечивается доступ обучающихся

Профессиональные базы данных:

1. <http://biblio.bsau.ru> - Электронная библиотека Башкирского ГАУ;
2. <http://znanium.com/> - Электронная библиотечная система;
3. <http://elibrary.ru> – Электронно-библиотечная система elibrary.

Ресурсы «Интернет»:

1. <https://edu.bsau.ru/> - Система управления обучением Башкирского ГАУ;
2. <http://window.edu.ru/> - "Единое окно": доступ к образовательным ресурсам;
3. <http://www.gks.ru/> - Федеральная служба государственной статистики.

Перечень информационно-справочных систем:

1. <http://biblio.bsau.ru> - Электронная библиотека Башкирского ГАУ;
2. <http://www.consultant.ru> – Справочная правовая система Консультант плюс;
3. <http://garant.ru> - Информационно-правовое обеспечение «Система ГАРАНТ».

11 Методические указания для обучающихся по освоению дисциплины

Изучаемая дисциплина поделена на 2 модуля. При изучении дисциплины «Web-программирование» предусматриваются: лекционное изложение курса, работа с учебниками и учебными пособиями, лабораторные работы, просмотры видеофильмов, обсуждение просмотренного материала, консультации по курсу. На лабораторных занятиях предусмотрены опросы, выполнение заданий, сдача оформленных лабораторных работ и рубежный тестовый контроль после завершения изучения каждого модуля.

Эффективность СРО проверяется преподавателем. Для этого при изучении каждого модуля студенты сдают отчеты о проработке каждой темы, предусмотренной в данном модуле. Заочное обучение предусматривает больший удельный вес самостоятельной работы.

Вид учебных занятий	Организация деятельности обучающегося
Занятия лекционного типа (лекция)	Написание конспекта лекций: кратко, схематично, последовательно фиксировать основные положения, выводы, формулировки, обобщения; помечать важные мысли, выделять ключевые слова, термины. Проверка терминов, понятий с помощью энциклопедий, словарей, справочников с выписыванием толкований в тетрадь. Обозначить вопросы, термины, материал, который вызывает трудности, пометить и попытаться найти ответ в рекомендуемой литературе. Если самостоятельно не удастся разобраться в материале, необходимо сформулировать вопрос и задать преподавателю на консультации, на лабораторной работе.
Занятия семинарского типа (лабораторные Работы)	Проработка методических указаний к выполнению лабораторных работ, уделяя особое внимание целям и задачам работы. Конспектирование источников. Работа с конспектом лекций, подготовка ответов к контрольным вопросам, просмотр рекомендуемой литературы.
Самостоятельная работа	Подготовка к занятиям лекционного и семинарского типа. Самостоятельное изучение теоретического материала, основной и дополнительной литературы, включая справочные издания, зарубежные источники и т.д. по разделам (модулям) дисциплины. Работа с конспектом лекций, подготовка ответов к контрольным вопросам, просмотр рекомендуемой литературы, работа с текстом, решение расчетно-графических заданий, решение задач по алгоритму и др.
Подготовка к зачету	При подготовке к зачету необходимо ориентироваться на конспекты лекций, рекомендуемую литературу и др.

Методические указания для обучающихся по освоению дисциплины

№	Наименование методических указаний,	Назначение
---	-------------------------------------	------------

п/п	тестов по дисциплине	(виды занятий, № тем и т.д.)
1	2	3
1	Web-программирование: методические указания к лабораторным работам и самостоятельной работе. / [сост. С. В. Прокофьева]. - Уфа: БГАУ, 2021. – 28 с.	Лабораторные работы № 1-8

12 Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю)

№ п/п	Наименование методических указаний, тестов по дисциплине	Назначение (виды занятий, № тем и т.д.)
1	Web-программирование: Методические указания к лабораторным работам и самостоятельной работе. / [сост. С. В. Прокофьева]. - Уфа: БГАУ, 2021. – 28 с.	Подготовка к ЛР СИТМ

13 Программное обеспечение, используемое при осуществлении образовательного процесса по дисциплине

Внеаудиторное контактное взаимодействие с обучающимися по самостоятельному изучению теоретического материала, выполнению контролируемых и /или неконтролируемых видов СРО осуществляется в системе управления обучением электронной информационной образовательной среды университета <https://edu.bsau.ru>.

Перечень программного обеспечения:

1. Microsoft Windows
2. Microsoft Office 2010 Standard
3. Антивирус Касперского
4. СПС Гарант

14 Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)

Для проведения лекционных занятий по данной дисциплине используются аудитории, оснащенные мультимедийным оборудованием.

Лабораторные работы проводятся в лабораториях, оснащенных необходимым оборудованием, обеспечивающих получение знаний по дисциплине.

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду университета.

Материально-техническое обеспечение дисциплины

№ п/п	Наименование	Назначение (виды занятий)
1	2	3
1	Аудитория для занятий лекционного типа	Лекции
2	Аудитория для занятий семинарского типа	Лабораторные работы
3	Аудитория для групповых и индивидуальных консультаций	Консультации
4	Аудитория для самостоятельной работы обучающегося	Самостоятельная работа обучающихся

Перечень лабораторного оборудования

№	Наименование	Кол-во, шт.
---	--------------	-------------

п/п		
1	2	3
1	Интерактив.доска SMARTBoard 680(диагональ 77"/195.6)	1
2	Коммутатор Allied Telesyn AT-FS7024 24*10/100Mbps	1
3	Компьютер Depo Neos	12
4	Компьютер(компл.)	1
5	Проектор BenQ Multimedia Projector MH FullMW	1
6	Стойка для интерактив.доски SMARTBoard 660/680	1

15 Особенности организации обучения по дисциплине для инвалидов и лиц с ограниченными возможностями здоровья

Организация обучения инвалидов и лиц с ограниченными возможностями здоровья (далее ОВЗ) осуществляется на основе адаптированной образовательной программы с использованием специальных методов обучения и дидактических материалов, составленных с учетом особенностей психофизического развития, индивидуальных возможностей и состояния здоровья таких обучающихся (обучающегося).

Образование инвалидов и лиц с ОВЗ может быть организовано как совместно с другими обучающимися, так и в отдельных группах или индивидуально.

Учебно-методические материалы для самостоятельной работы обучающихся из числа инвалидов и лиц с ОВЗ предоставляются в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Категория обучающихся	Формы предоставления материалов
С нарушением слуха	- в печатной форме; - в форме электронного документа.
С нарушением зрения	- в печатной форме увеличенным шрифтом; - в форме электронного документа; - в форме аудиофайла.
С нарушением опорно-двигательного аппарата	- в печатной форме увеличенным шрифтом; - в форме электронного документа; - в форме аудиофайла.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

Для обучающихся инвалидов и лиц с ОВЗ предусмотрены следующие оценочные средства:

Категория обучающихся	Виды оценочных средств	Формы контроля и оценки результатов обучения
С нарушением слуха	тест	преимущественно письменная проверка
С нарушением зрения	собеседование	преимущественно устная проверка (индивидуально)
С нарушением опорно-двигательного аппарата	решение дистанционных тестов, контрольные вопросы	организация контроля с помощью LMS Башкирского ГАУ, письменная проверка.

Обучающимся инвалидам и лицам с ОВЗ увеличивается время на подготовку ответов к зачёту, допускается готовить ответы с использованием дистанционных образовательных технологий.

При проведении процедуры оценивания результатов обучения инвалидов и лиц с ОВЗ предусматривается использование технических средств, необходимых им в связи с их индивиду-

альными особенностями. Эти средства предоставляются ФГБОУ ВО Башкирский ГАУ или могут использоваться собственные технические средства обучающихся.

Процедура оценивания результатов обучения инвалидов и лиц с ОВЗ по дисциплине предусматривает предоставление информации в формах, адаптированных к ограничениям их здоровья и восприятия информации.

Так для лиц с нарушениями зрения:

- в печатной форме увеличенным шрифтом;
- в форме электронного документа;
- в форме аудиофайла.

Для лиц с нарушениями слуха:

- в печатной форме;
- в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

- в печатной форме;
- в форме электронного документа;
- в форме аудиофайла.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

При проведении процедуры оценивания результатов обучения инвалидов и лиц с ОВЗ по дисциплине (модулю) обеспечивается выполнение следующих дополнительных требований в зависимости от индивидуальных особенностей обучающихся:

1. Инструкция по порядку проведения процедуры оценивания предоставляется в доступной форме (устно, в письменной форме, устно с использованием услуг сурдопереводчика).

2. Доступная форма предоставления заданий оценочных средств (в печатной форме, в печатной форме увеличенным шрифтом, в форме электронного документа, задания зачитываются ассистентом, задания предоставляются с использованием сурдоперевода).

3. Доступная форма предоставления ответов на задания (письменно на бумаге, набор ответов на компьютере, с использованием услуг ассистента, устно).

При необходимости для инвалидов и обучающихся с ОВЗ процедура оценивания результатов обучения по дисциплине (модулю) может проводиться в несколько этапов. Проведение процедуры оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья допускается с использованием дистанционных образовательных технологий.

Для освоения дисциплины инвалидами и лицами с ОВЗ предоставляются основная и дополнительная учебная литература в виде электронного документа в фонде библиотеки и / или в электронно-библиотечных системах. А также предоставляются бесплатно специальные учебники и учебные пособия, иная учебная литература и специальные технические средства обучения коллективного и индивидуального пользования, а также услуги сурдопереводчиков и тифлосурдопереводчиков.

В зависимости от нозологии для пользователей с ОВЗ организован доступ к электронным информационным и образовательным ресурсам библиотеки университета из любой точки с доступом к «Интернет». Заключен договор о сотрудничестве с Башкирской республиканской специальной библиотекой для слепых. Предоставляется возможность аудио прослушивания и сохранения файла электронных изданий ЭБС «Консультант студента. Электронная библиотека технического вуза» (полные тексты изданий доступны пользователям ФГБОУ ВО Башкирский ГАУ, после самостоятельной регистрации в Электронной библиотечной системе Университета). Предоставляется возможность пользоваться бесплатным мобильным приложением для операционных систем IOS и Android ЭБС издательства «Лань», с синтезатором речи (возможность использования книг в учебном процессе для незрячих и слабовидящих обучающихся).

В освоении дисциплины инвалидами и лицами с ОВЗ большое значение имеет индивидуальная работа. Под индивидуальной работой подразумевается две формы взаимодействия с преподавателем: индивидуальная учебная работа (консультации), т.е. дополнительное разъяснение учебного материала и углубленное изучение материала с теми обучающимися, которые в этом за-

интересованы, и индивидуальная воспитательная работа. Индивидуальные консультации по предмету являются важным фактором, способствующим индивидуализации обучения и установлению воспитательного контакта между преподавателем и обучающимся инвалидом или обучающимся с ОВЗ.

Освоение дисциплины инвалидами и лицами с ОВЗ осуществляется с использованием специальных средств обучения. Оборудовано специализированное помещение, в котором установлен мультимедийный проектор и организовано два рабочих места с доступом к электронной информационной образовательной среде и сети Интернет. Данное помещение оснащено: индукционной петлей ИС-50Л (усиление звука для слабослышащих обучающихся); персональными компьютерами, с программой экранного доступа ("Jaws for Windows 16.0 Pro"), брайлевским дисплеем (тактильный дисплей Брайля PAC Mate 20) для студентов с нарушением зрения; специальными партами для обучающихся с нарушением опорно-двигательного аппарата; мобильным видеоувеличителем; портативной информационной индукционной системой "Исток А2" для слабослышащих обучающихся.

**Фонд оценочных средств
для проведения промежуточной аттестации обучающихся по дисциплине**

**1 Перечень компетенций и этапы формирования компетенций в процессе освоения ОПОП
ВО**

Код и наименование компетенции	Код и наименование индикаторов достижения компетенции	Этап формирования (указывается семестр)
ПК-6 Выполнение работ по созданию (модификации) и сопровождению ИС, автоматизирующих задачи организационного управления и бизнес-процессы	ПК-6.1 Разработка прототипов ИС на базе типовой ИС	7
	ПК-6.2 Установка и настройка системного и прикладного ПО, необходимого для функционирования ИС	7

2 Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

2.1 Показатели и критерии оценивания компетенций

Компетенция ПК-6 Выполнение работ по созданию (модификации) и сопровождению ИС, автоматизирующих задачи организационного управления и бизнес-процессы

ИДК - ПК-6.1 Разработка прототипов ИС на базе типовой ИС

Планируемые результаты (показатели оценивания)		Критерии оценивания			
		Ниже порогового уровня (неудовл.)	Пороговый уровень (удовл.)	Повышенный уровень (хорошо)	Высокий уровень (отлично)
		Не зачтено	Зачтено		
Знания	ПК-6.1/Зн1 языков программирования и работы с базами данных; инструменты и методов модульного тестирования, инструментов и методов тестирования нефункциональных и функциональных характеристик ИС; ин-	Отсутствие или фрагментарное знание языков программирования и работы с базами данных; инструментов и методов модульного тестирования, инструментов и методов тестирования нефункциональных и функциональных характеристик ИС; ин-	Неполное знание языков программирования и работы с базами данных; инструменты и методов модульного тестирования, инструментов и методов тестирования нефункциональных и функциональных характеристик ИС; инструментов и методов прототипи-	В целом сформировавшееся знание языков программирования и работы с базами данных; инструменты и методов модульного тестирования, инструментов и методов тестирования нефункциональных и функциональных характеристик	Сформировавшееся систематическое знание языков программирования и работы с базами данных; инструменты и методов модульного тестирования, инструментов и методов тестирования нефункциональных и функциональных ха-

	функциональных и функциональных характеристик ИС; инструментов и методов прототипирования пользовательского интерфейса	струментов и методов прототипирования пользовательского интерфейса	рования пользовательского интерфейса	ИС; инструментов и методов прототипирования пользовательского интерфейса	рактических ИС; инструментов и методов прототипирования пользовательского интерфейса
Умения	ПК-6.1/Ум1 кодировать на языках программирования в процессе разработки прототипов ИС на базе типовой ИС; тестировать результаты прототипирования; проводить презентации, проводить переговоры в ходе согласования пользовательского интерфейса с заказчиком	Отсутствие или фрагментарное умение кодировать на языках программирования в процессе разработки прототипов ИС на базе типовой ИС; тестировать результаты прототипирования; проводить презентации, проводить переговоры в ходе согласования пользовательского интерфейса с заказчиком	Неполное умение кодировать на языках программирования в процессе разработки прототипов ИС на базе типовой ИС; тестировать результаты прототипирования; проводить презентации, проводить переговоры в ходе согласования пользовательского интерфейса с заказчиком	В целом сформировавшееся умение кодировать на языках программирования в процессе разработки прототипов ИС на базе типовой ИС; тестировать результаты прототипирования; проводить презентации, проводить переговоры в ходе согласования пользовательского интерфейса с заказчиком	Сформировавшееся систематическое умение кодировать на языках программирования в процессе разработки прототипов ИС на базе типовой ИС; тестировать результаты прототипирования; проводить презентации, проводить переговоры в ходе согласования пользовательского интерфейса с заказчиком
Навыки	ПК-6.1/Нв1 разработки прототипа ИС на базе типовой ИС в соответствии с требованиями; тестирования прототипа ИС на корректность архитектурных решений, проведения анализа результатов тестирования; принятия решения о пригодности архитектуры, согласования пользовательского интерфейса с заказчиком	Отсутствие или фрагментарное владение навыками разработки прототипа ИС на базе типовой ИС в соответствии с требованиями; тестирования прототипа ИС на корректность архитектурных решений, проведения анализа результатов тестирования; принятия решения о пригодности архитектуры, согласования пользовательского интерфейса с заказчиком	Неполное владение навыками разработки прототипа ИС на базе типовой ИС в соответствии с требованиями; тестирования прототипа ИС на корректность архитектурных решений, проведения анализа результатов тестирования; принятия решения о пригодности архитектуры, согласования пользовательского интерфейса с заказчиком	В целом сформировавшееся владение навыками разработки прототипа ИС на базе типовой ИС в соответствии с требованиями; тестирования прототипа ИС на корректность архитектурных решений, проведения анализа результатов тестирования; принятия решения о пригодности архитектуры, согласования пользовательского интерфейса с заказчиком	Сформировавшееся систематическое владение навыками разработки прототипа ИС на базе типовой ИС в соответствии с требованиями; тестирования прототипа ИС на корректность архитектурных решений, проведения анализа результатов тестирования; принятия решения о пригодности архитектуры, согласования пользовательского интерфейса с заказчиком

	телефонного интерфейса с заказчиком.				
--	--	--	--	--	--

ИДК - ПК-6.2 Установка и настройка системного и прикладного ПО, необходимого для функционирования ИС

Планируемые результаты (показатели оценивания)		Критерии оценивания			
		Ниже порогового уровня (неудовл.)	Пороговый уро- вень (удовл.)	Повышенный уровень (хорошо)	Высокий уровень (отлично)
		Не зачтено	Зачтено		
Знания	ПК-6.2/Зн1 основ современных операционных систем; основ современных систем управления базами данных; основ администриров ания СУБД; устройства и функциониров ания современных ИС, коммуникацио нного оборудования, сетевых протоколов, основ системного администриров ания	Отсутствие или фрагментарное знание основ современных операционных систем; основ современных систем управления базами данных; основ администрирован ия СУБД; устройства и функционирован ия современных ИС, коммуникационн ого оборудования, сетевых протоколов, основ системного администрирован ия	Неполное знание основ современных операционных систем; основ современных систем управления базами данных; основ администрировани я СУБД; устройства и функционирования современных ИС, коммуникационно го оборудования, сетевых протоколов, основ системного администрировани я	В целом сформировавшееся знание основ современных операционных систем; основ современных систем управления базами данных; основ администрирован ия СУБД; устройства и функционировани я современных ИС, коммуникационн ого оборудования, сетевых протоколов, основ системного администрирован ия	Сформировавшееся систематическое знание основ современных операционных систем; основ современных систем управления базами данных; основ администрировани я СУБД; устройства и функционировани я современных ИС, коммуникационн ого оборудования, сетевых протоколов, основ системного администрировани я
Умения	ПК-6.2/Ум1 устанавливать и настраивать операционные системы, СУБД, прикладное ПО	Отсутствие или фрагментарное умение устанавливать и настраивать операционные системы, СУБД, прикладное ПО	Неполное умение устанавливать и настраивать операционные системы, СУБД, прикладное ПО	В целом сформировавшееся умение устанавливать и настраивать операционные системы, СУБД, прикладное ПО	Сформировавшееся систематическое умение устанавливать и настраивать операционные системы, СУБД, прикладное ПО
Навыки	ПК-6.2/Нв1 установки операционных систем, СУБД, прикладного ПО, необходимых для функциониров ания ИС; настройки операционных систем, СУБД, прикладного ПО для	Отсутствие или фрагментарное владение навыками установки операционных систем, СУБД, прикладного ПО, необходимых для функционирован ия ИС; настройки операционных систем, СУБД, прикладного ПО для	Неполное владение навыками установки операционных систем, СУБД, прикладного ПО, необходимых для функционирования ИС; настройки операционных систем, СУБД, прикладного ПО для оптимального функционирования	В целом сформировавшееся владение навыками установки операционных систем, СУБД, прикладного ПО, необходимых для функционировани я ИС; настройки операционных систем, СУБД, прикладного ПО для оптимального	Сформировавшееся систематическое владение навыками установки операционных систем, СУБД, прикладного ПО, необходимых для функционировани я ИС; настройки операционных систем, СУБД, прикладного ПО

	оптимального функционирования ИС.	оптимального функционирования ИС.	ИС.	функционирования ИС.	для оптимального функционирования ИС.
--	-----------------------------------	-----------------------------------	-----	----------------------	---------------------------------------

2.3 Критерии оценки по 5-ти балльной системе

2.3.1 Критерии оценки по пятибалльной системе (5 семестр)

Результат зачета	Критерии
«зачтено»	Обучающийся показал знания основных положений учебной дисциплины, умение решать конкретные практические задачи, предусмотренные рабочей программой, ориентироваться в рекомендованной справочной литературе, умеет правильно оценить полученные результаты расчетов или эксперимента
«не зачтено»	При ответе обучающегося выявились существенные пробелы в знаниях основных положений учебной дисциплины, неумение с помощью преподавателя получить правильное решение конкретной практической задачи из числа предусмотренных рабочей программой учебной дисциплины

3 Фонд оценочных средств для оценки достижения обучающимися результатов освоения дисциплины

Раскрываемые компетенции: ПК-6

1. Охарактеризовать архитектуру Клиент-сервер (Client-server)

Клиент-сервер — вычислительная или сетевая архитектура, в которой задания или сетевая нагрузка распределены между поставщиками услуг (сервисов), называемыми серверами, и заказчиками услуг, называемыми клиентами. Часто клиенты и серверы взаимодействуют через компьютерную сеть и могут быть как различными физическими устройствами, так и программным обеспечением.

2. Что такое Web-сервер

Web-сервер — это сервер, принимающий HTTP-запросы от клиентов, обычно web-браузеров, и выдающий им HTTP-ответы, обычно вместе с HTML-страницей, изображением, файлом, медиапоток или другими данными. Web-серверы — основа Всемирной паутины. Web-сервером называют как ПО, выполняющее функции web-сервера, так и непосредственно компьютер, на котором это ПО работает.

3. Охарактеризовать технологию AJAX (Asynchronous Javascript and XML — «асинхронный JavaScript и XML»)

Это подход к построению интерактивных пользовательских интерфейсов веб-приложений, заключающийся в «фоновом» обмене данными браузера с веб-сервером. При обновлении данных веб-страница не перезагружается полностью, и веб-приложения становятся быстрее и удобнее.

4. Охарактеризовать технологию DHTML (Dynamic HTML)

это способ (подход) создания интерактивного веб-сайта, использующий сочетание статичного языка разметки HTML, встраиваемого (и выполняемого на стороне клиента) скриптового языка JavaScript, CSS (каскадных таблиц стилей) и DOM (объектной модели документа).

5. В каких случаях используют технологию DHTML (Dynamic HTML), привести примеры использования

DHTML используется: для создания приложения в веб-браузере: например, для более простой навигации или для придания интерактивности форме; для динамического перетаскивания элементов по экрану; как инструмент для создания основанных на браузере видеоигр.

6. Охарактеризовать язык HTML (HyperText Markup Language — «язык гипертекстовой разметки»)

— стандартный язык разметки документов во Всемирной паутине, система команд, основная задача которой — представление текстовых и мультимедийных элементов Web-страницы в наиболее благоприятной для восприятия форме. Большинство веб-страниц содержат описание разметки на языке HTML (или XHTML). Язык HTML интерпретируется браузерами и отображается в виде документа в удобной для человека форме.

7. Что представляют собой расширения Web-сервера (приложения формата ISAPI, модули расширения Apache и т. п.).

Это способ, позволяющий встраивать серверные программы в сам Web-сервер, делая их его составными частями.

8. Что представляют собой активные серверные страницы (PHP, ASP, JSP и др.).

Фактически это обычные статические Web-страницы, сохраненные в файлах, которые, кроме обычного HTML-кода, включают в себя команды, обрабатываемые либо самим Web-сервером, либо его расширением.

9. Пояснить отличие серверной страницы и серверного сценария

Серверные сценарии — подобны активным серверным страницам тем, что являются интерпретируемыми, однако представляют собой "чистый" программный код, без HTML-"примесей".

10. Что такое WAMP

WAMP — акроним, обозначающий набор (комплекс) серверного программного обеспечения, широко используемый с целью разработки, предоставления веб-сервисов. WAMP назван по первым буквам входящих в его состав компонентов: Windows — операционная система от компании Microsoft; Apache — веб-сервер; MySQL — СУБД; PHP — язык программирования, используемый для создания веб-приложений.

11. Назначение языка PHP

PHP язык программирования, созданный для генерирования HTML-страниц на веб-сервере и работы с базами данных.

12. Достоинства языка PHP

Плюсы PHP: язык с открытым кодом; прост для изучения; кроссплатформенный, т.е. может использоваться на всех крупных ОС; большая библиотека готовых скриптов.

13. Назначение функции `mysqli_connect()` в PHP

Перед началом работы с данными внутри MySQL, нужно открыть соединение с сервером СУБД. В PHP это делается с помощью стандартной функции `mysqli_connect()`. Функция возвращает результат — указатель соединения. Данный указатель используется для всех следующих операций с MySQL.

14. Назначение функции `mysqli_select_db()` в PHP

Функция используется для выбора текущей базы данных в сценарии PHP `mysqli_connect()`

15. Назначение функции `mysqli_query()` в PHP

При помощи этой функции осуществляется выполнение SQL-запросов в сценариях PHP

16. Назначение функций `mysqli_fetch_array()` и `mysqli_result()` в PHP

При помощи этих функций можно получить значения, возвращаемые СУБД MySQL после работы функции `mysqli_query()`

17. Для чего нужна функция `mysqli_set_charset()` в PHP

Назначение функции `mysqli_set_charset()` в PHP - выставление кодировки

18. Что является аргументом функции `md5()`

В сценариях PHP в качестве аргумента функции `md5()` указывается **шифруемая строка**

19. Какое действие в файле с расширением `.php` осуществляет функция `mysqli_query("select * from jobs where JobID=$ID");`

Производит выборку из таблицы `jobs` всех полей тех строк, в которых значение поля `JobID` равно значению переменной `$ID`

20. Дан фрагмент кода HTML:

```
<FORM NAME="talk" ACTION="priem_vklada.php" METHOD="GET">
<HR> Прием вклада: <HR>
Фамилия И.О.:<INPUT TYPE="TEXT" NAME="name" SIZE="35">
<BR><BR>
Отделение банка:
<INPUT TYPE="RADIO" NAME="otdel" VALUE="N"> Северное
<INPUT TYPE="RADIO" NAME="otdel" VALUE="C" CHECKED> Центральное
<INPUT TYPE="RADIO" NAME="otdel" VALUE="O"> Восточное
<BR><BR><INPUT TYPE="SUBMIT" VALUE="Принять">
<INPUT TYPE="RESET" VALUE="Очистить форму">
</FORM>
```

Как в сценарии PHP можно получить данные из текстового поля этой формы

С данными, переданными из текстового поля этой формы в сценарий `priem_vklada.php`, можно оперировать, как элементами массива `$_GET["name"]` или `$_REQUEST["name"]`

21. Что выведет на веб-страницу следующий код:

```
<?php
$name = array («d» => «Mark», «a» => «David», «b» => «Peter», «c»
=> «Martha»);
$nameArrayObject = new ArrayObject($name);
$nameArrayObject->ksort();
foreach ($nameArrayObject as $key => $val)
{
    echo "$key = $val";
}
?>
```

Результат выполнения кода будет следующим: `a = David, b = Peter, c = Martha, d = Mark`

22. Как выглядит операция конкатенации в PHP

Для конкатенации строк в PHP используется оператор `.` (точка)

23. Назначение атрибута `action` у тега `form`

С помощью этого атрибута тега <form> можно задать URL, на который передаются данные введенные в форму

24 . Что означает запрос: SELECT DISTINCT Product_No, Prod_Category FROM Products;

Из таблицы «Products» сделать выборку из полей Product_No, Prod_Category без дублирования данных.

25. Назвать основные операторы модификации данных в SQL

При модификации данных с помощью SQL основными являются операторы Insert, Delete и Update

26. Для чего предназначен параметр METHOD тега FORM? Какие значения параметра METHOD используются чаще всего?

Этот атрибут определяет способ отправки данных. Протокол HTTP предоставляет несколько способов выполнить запрос; данные HTML-формы могут передаваться несколькими различными способами, наиболее распространёнными из которых являются метод GET и метод POST.

27. Особенности передачи данных методом GET

Метод GET - это метод, используемый браузером, который говорит серверу, что нужно отправить назад данный ресурс. В этом случае браузер отправляет пустое тело запроса, а данные, отправляемые на сервер, добавляются к URL-адресу.

28. Особенности передачи данных методом POST

Браузер использует этот метод для связи с сервером при запросе ответа с учётом данных, представленных в теле HTTP-запроса. Если форма отправляется с использованием этого метода, данные добавляются в тело HTTP-запроса.

29. Что собой представляет веб-интерфейс

Веб-интерфейс — это совокупность средств, при помощи которых пользователь взаимодействует с веб-сайтом или любым другим приложением через браузер.

30. Охарактеризовать язык SQL

SQL - информационно-логический язык, предназначенный для описания хранимых данных, для извлечения хранимых данных и для модификации данных.

31. Требования к имени переменной в PHP

Требования к имени переменной: начинается символом доллара (\$); содержит буквы латинского алфавита, цифры и знаки подчеркивания; после \$ должна быть буква или знак подчеркивания; не должно совпадать с ключевыми словами PHP.

32. Назначение функции die (message) в PHP

Эта функция завершает текущий скрипт и выводит в окно браузера сообщение message. Следующие операторы в скрипте не выполняются.

33. Что собой представляют суперглобальные массивы (Superglobals) в PHP

В PHP имеются предопределенные переменные типа array (массив) с глобальной областью видимости. Их еще называют суперглобальными массивами, поскольку они доступны в любом сценарии на языке PHP.

34. Назначение суперглобального массива \$_GET

`$_GET['variable']` - переменные, переданные сценарию через HTTP GET.

35. Назначение суперглобального массива \$_POST

`$_POST['variable']` - переменные, переданные сценарию через HTTP POST.

36. Поясните назначение тега и его параметров `<input type="submit" value="Ввод">` и его параметров.

HTML тег `<input>` позволяет создавать различные поля ввода или выбора значения для html-форм. Параметр `type="submit"` определяет тип элемента – кнопку для отправки данных на сервер. `value` – надпись на кнопке.

37. Назначение оператора return в PHP

Чтобы вернуть результат из функции в выражение, из которого она была вызвана, используется оператор `return`: `return <переменная или выражение>;`. Здесь переменная должна содержать возвращаемое значение, а выражение должно его вычислять.

38. Назначение функции include() в PHP

Функция `include()` включает в сценарий фрагменты кода, сохраненные в других файлах.

39. Поясните назначение функции и ее аргументов: `fopen("c:/www/html/file.txt","r");`

При помощи функции `fopen` производится открытие файлов в файловой системе сервера. В случае удачного открытия файла, функция `fopen` возвращает дескриптор файла, в случае неудачи - `false`. `c:/www/html/file.txt` - имя файла и путь к нему. `r` - открыть файл только для чтения, после открытия указатель файла устанавливается в начало файла.

40. Поясните назначение функции и ее аргументов:

`fopen("c:/www/html/picture1.jpg","rb");`

При помощи функции `fopen` производится открытие файлов в файловой системе сервера. `c:/www/html/picture1.jpg` - имя файла и путь к нему. `r` - открыть файл только для чтения, после открытия указатель файла устанавливается в начало файла. `b` - указывает на работу с двоичным файлом. В случае удачного открытия файла, функция `fopen` возвращает дескриптор файла, в случае неудачи - `false`.

41. Что такое Cookies

Cookies - это текстовые строки, хранящиеся на стороне клиента, и содержащие пары "имя - значение", с которыми связан URL, по которому браузер определяет нужно ли посылать cookies на сервер.

42. Для чего и как используются сессии и cookies

Сессии и cookies предназначены для хранения сведений о пользователях при переходах между несколькими страницами. Очень важным в веб-программировании является возможность передавать данные с одной страницы на другую. Обычно это происходит, когда имеем дело с аутентификацией пользователей, а также с сообщениями об ошибках и т.п. При использовании сессий данные сохраняются во временных файлах на сервере. Файлы с cookies хранятся на компьютере пользователя и по запросу отсылаются браузером серверу.

43. Что такое сессия

Сеанс (сессия) представляет собой группу переменных, которые, в отличие от обычных переменных, сохраняются и после завершения выполнения PHP-сценария. Сессии предназначены для хранения сведений о пользователях при переходах между несколькими страницами. При использовании сессий данные сохраняются во временных файлах на сервере.

44. Назначение функции session_start () в PHP

Функция session_start () вызывается в начале PHP-сценария и проверяет, существует ли идентификатор сессии: если идентификатора нет, то создает его; если идентификатор текущей сессии уже существует, то загружаются зарегистрированные переменные сессии.

45. Поясните назначение функции и ее аргументов: setcookie("name", \$value, time() + 600);

Это пример установки срока годности cookies. name - имя устанавливаемого cookie; value - значение, хранящееся в cookie с именем (для cookie с именем name значение будет храниться в переменной \$name); time() + 600 - время в секундах, по истечении которого текущий cookie становится недействительным. Этот cookie действителен в течение 10 мин после создания.

46. Особенности файл-серверных СУБД

В файл-серверных СУБД файлы данных располагаются централизованно на файл-сервере СУБД. Ядро СУБД располагается на каждом клиентском компьютере. Доступ к данным осуществляется через локальную сеть. Синхронизация чтений и обновлений осуществляется посредством файловых блокировок.

47. Преимущества и недостатки файл-серверных СУБД

Преимуществом этой архитектуры является низкая нагрузка на ЦП сервера, а недостатком — высокая загрузка локальной сети.

48. Особенности клиент-серверных СУБД, их отличие от файл-серверных

Такие СУБД состоят из клиентской части (которая входит в состав прикладной программы) и сервера СУБД. Клиент-серверные СУБД, в отличие от файл-серверных, обеспечивают разграничение доступа между пользователями и мало загружают сеть и клиентские машины. Сервер является внешней по отношению к клиенту программой, и по надобности его можно заменить другим.

49. Назначение функции array () в PHP

Один из способов создания массива — использование функции array. Элементы будущего массива указываются в качестве аргументов этой функции.

50. Что представляют собой формы в HTML-документе?

Формы предназначены для пересылки данных от пользователя к веб-серверу. Формы в HTML могут состоять из текстовых полей и текстовых областей, флажков и переключателей, а также раскрывающихся списков. Все это — элементы формы. Каждый элемент служит для того, чтобы передать какое-либо значение сайту.

52. Какие параметры могут быть у тега <FORM>?

У <form> могут быть следующие параметры: id — идентификатор формы, action — указывает действие, которое будет произведено с введенными данными, method — способ отправки данных серверу

53. Что представляет собой простое текстовое поле в HTML-форме, как его создать, как передаются на сервер данные из этого поля?

Однострочное текстовое поле предназначено для ввода строки символов с помощью клавиатуры. Синтаксис создания такого поля следующий. `<input атрибуты>`. Значение атрибута `type` для тега `<input>` по умолчанию определено как `text`, поэтому его можно не указывать явно. Значение элемента, переданное на сервер, будет состоять из тех данных, которые были введены пользователем в поле.

54. Какие бывают виды кнопок в HTML-форме, как их создать?

Кнопка `submit` отправляет данные формы на сервер. Кнопка `reset` служит для очистки введенных данных формы и возвращения значений в первоначальное состояние. `Button` - используется для вызова клиентских скриптов. Кнопку на веб-странице можно создать двумя способами — с помощью тега `<input>` и тега `<button>`.

55. В каких случаях следует использовать раскрывающийся список `<SELECT>` на Web-странице.

Раскрывающиеся списки, также, как и флажки или радиокнопки, используются для выбора одного значения из некоторого набора значений. Преимущество раскрывающегося списка в том, что он обычно уместит множество вариантов выбора всего в одной строке и раскрывается только при обращении к нему, что экономит место на Web-странице.

56. Как создать текстовую область в форме и как задать размеры текстовой области?

Текстовая область создается парным тегом `<TEXTAREA>`. Между тегами `<textarea>` и `</textarea>` можно поместить любой текст, который будет отображаться внутри поля. Размеры задаются с помощью параметров `rows` - количество строк, выводимых на экран без прокрутки и `cols` - ширина поля в символах.

57. В каких случаях следует использовать флажки и как создать флажок в форме документа HTML?

Флажки используют, когда необходимо выбрать любое количество вариантов из предложенного списка. Флажок создается с помощью тега. `<input type="checkbox" атрибуты>`. Параметр `checked` - предварительное выделение флажка.

58. В каких случаях следует использовать радиокнопки? Как сделать группу радиокнопок?

Тега `<input type=radio >` используется для создания группы радиокнопок (переключателей), описывающих набор взаимосвязанных параметров. Группа радиокнопок определяется путём присвоения каждой радиокнопке в данной группе одного и того же значения атрибута `name`. Тогда пользователь сможет выбрать лишь одну радиокнопку из предложенных.

59. Поясните назначение тега `<input type="text" size="15">` и его параметров.

Тег `<input>` позволяет создавать различные поля ввода или выбора значения для html-форм. Параметр `type="text"` определяет тип элемента формы – однострочное текстовое поле. `Size` - ширина текстового поля, 15 символов.

60. Для чего предназначен тег `<input type="checkbox" NAME="ch" VALUE="yes">`, какие данные будут переданы на сервер из этого элемента?

Флажок checkbox представляет собой элемент HTML формы, который может иметь два состояния: отмечен и не отмечен (стоит галочка или нет). NAME – имя элемента. Отмеченный чекбокс будет посылать на сервер содержимое атрибута value, т.е. ch=yes.

61. Дан фрагмент HTML-кода <INPUT TYPE="radio" NAME="rd" VALUE="rd1"> yes <IN-PUT TYPE="radio" NAME="rd" VALUE="rd2"> no. Поясните назначение данных тегов и их параметров.

<input> со значением type = radio используется для создания группы радиокнопок (переключателей). Группа радиокнопок определяется путём присвоения каждой радиокнопке С одинаковыми значениями name создается группа радиокнопок, чтобы пользователь мог выбрать лишь одну радиокнопку из предложенных. Value задаёт, какое значение будет отправлено на сервер.

62. Дан фрагмент HTML-кода <textarea rows="10" cols="45" name="text">Ваш отзыв</textarea>. Поясните назначение данного тега и его параметров.

Парным тегом <TEXTAREA> создается текстовая область. Параметр rows - количество строк, выводимых на экран без прокрутки и cols - ширина поля в символах, name - имя элемента.

63. Дан фрагмент HTML-кода <input type="submit" value="Войти">. Поясните назначение данного тега и его параметров.

HTML тег <input> позволяет создавать различные поля ввода или выбора значения для html-форм. Параметр type=" submit " определяет тип элемента – кнопку для отправки данных на сервер. value – надпись на кнопке.

64. Поясните назначение тега <input type="file">

<input> элемент с атрибутом type="file" позволяет пользователю выбрать один файл или более из файлового хранилища своего устройства. После выбора эти файлы могут быть загружены на сервер или обработаны JavaScript.

65. Поясните назначение нижеприведенной конструкции и входящих в нее тегов: <select> <option> Москва</option> <option> Санкт-Петербург</option> </select>..

В примере приведен раскрывающийся список. <select> определяет контейнер для списка, а тег <option> используется для определения каждого элемента списка.

66. Для чего предназначен тег <input type="password">, в чем его особенность.

Элемент формы для ввода пароля. Текст, который вводит пользователь в данное поле, заменяется в целях безопасности на специальные символы — звездочки или точки.

67. Дан фрагмент HTML-кода <input type="reset" value="Очистить">. Поясните назначение данного тега и его параметров.

HTML тег <input> позволяет создавать различные поля ввода или выбора значения для html-форм. Параметр type=" reset " определяет тип элемента – кнопку для очистки формы. value – надпись на кнопке.

68. Что такое массив, элемент массива и индекс элемента?

Массив — это пронумерованный набор переменных, представляющих собой единое целое. Переменные, входящие в массив, называются его элементами. Доступ к нужному элементу массива выполняется по его порядковому номеру, называемому индексом.

69. Назначение суперглобального массива \$_COOKIE

\$_COOKIE['variable'] - переменные, переданные сценарию через HTTP cookies.

70. Назначение суперглобального массива \$_REQUEST

`$_REQUEST['variable']` – переменные, переданные сценарию в результате ответа пользователя на запрос (GET, POST, COOKIE) и которым, поэтому, не следует доверять.

71. Назначение суперглобального массива \$_GLOBALS

`$_GLOBALS['variable']` - содержит ссылки на все переменные, которые в настоящий момент доступны в глобальной области видимости сценария. Ключами этого массива являются имена глобальных переменных.

72. Назначение суперглобального массива \$_SERVER

`$_SERVER['variable']` - переменные, установленные web-сервером или чем-то другим, напрямую связанным со средой исполнения сценария.

73. Назначение суперглобального массива \$_FILES

`$_FILES['variable']` - переменные, переданные сценарию путем передачи файла через HTTP.

73. Назначение суперглобального массива \$_SESSION

`$_SESSION['variable']` - переменные, которые определены в текущей сессии сценария.

74. Назначение тега <input type="reset" value="Сброс">

Данный HTML-тег определяет кнопку очистки формы. При активации данной кнопки происходит сброс всех введенных пользователем данных формы до первоначальных значений.

75. Как создать раскрывающийся список в форме документа HTML и как задать элементы списка?

Для создания выпадающего списка в HTML используется тег `<select>` в сочетании с тегом `<option>`. Тег `<select>` определяет контейнер для списка, а тег `<option>` используется для определения каждого элемента списка.

76. Что возвращает функция Mysqli query при выполнении запроса на выборку данных (SELECT) ?

При выполнении запроса на выборку данных (SELECT) функция Mysqli query возвращает объект, который содержит все найденные строки и методы для работы с ними. Методы этого объекта позволяют получить данные из каждой строки, отдельные значения по столбцам, количество найденных строк и другую полезную информацию.

77. Что возвращает функция Mysqli_query при выполнении запроса на изменение данных (INSERT, UPDATE, DELETE)?

При выполнении запроса на изменение данных (INSERT, UPDATE, DELETE) функция Mysqli_query возвращает true в случае успешного выполнения запроса и false в случае ошибки. Это позволяет проверить результат выполнения запроса и принять необходимые действия в зависимости от него.

78. Как происходит обработка GET – данных

Для доступа к данным, отправленным методом GET, служит массив `$_GET`. В качестве ключей этого массива выступают имена полей из переданных форм, а в качестве значений — данные, введенные в эти поля.

79. Как происходит обработка POST– данных

Информация, поступившая методом POST, будет доступна в массиве `$_POST`. В качестве ключей этого массива выступают имена полей из переданных форм, а в качестве значений — данные, введенные в эти поля.

80. Назначение функции `strlen()` в PHP

Функция `strlen()` возвращает длину строки, которую принимает в качестве аргумента

81. Назначение функции `substr(str, start[, length])` в PHP

Эта функция возвращает часть строки: первый аргумент – исходная строка; второй – позиция первого символа (отсчет начинается с нуля); третий – длина строки в символах, которую надо вернуть (если не указан, то возвращается вся оставшаяся часть строки).

82. Как выглядит оператор вывода в PHP? Привести примеры использования.

Оператор вывода `echo` выводит значение выражения, указанное после него, в то место страницы, где сам находится. Например, `echo 2+2;`. Строковое значение заключается в двойные кавычки, например: `echo "Привет";`

83. Как выполнить конфигурацию PHP?

Система PHP конфигурируется при помощи текстового конфигурационного файла `php.ini`, который располагается в корневом каталоге PHP. В файле `php.ini` содержатся параметры интерпретации PHP-кода, безопасности системы, способов работы с файлами и данными, а также информация о доступных расширениях PHP.

84. Где находятся настройки сервера Apache, как открыть их для изменения?

Все настройки сервера Apache хранятся в текстовом файле `httpd.conf`, который располагается в каталоге `conf`. Нужно лишь открыть его в любом текстовом редакторе, который поддерживает стандартный текстовый формат, и внести соответствующие изменения.

85. Что такое LAMP

LAMP — акроним, обозначающий набор (комплекс) серверного программного обеспечения, широко используемый во Всемирной паутине. LAMP назван по первым буквам входящих в его состав компонентов: Linux — операционная система Linux; Apache — веб-сервер; MySQL — СУБД; PHP — язык программирования, используемый для создания веб-приложений (помимо PHP могут подразумеваться другие языки, такие как Perl и Python).

86. Что такое XAMPP

XAMPP - кроссплатформенная сборка веб-сервера, X (любая из четырех операционных систем), Apache, MySQL, PHP, Perl. XAMPP работает со всеми 32-х разрядными ОС Microsoft Windows, а также с Linux, Mac OS X и Solaris.

87. Как выглядит оператор простого присваивания в PHP

Оператор простого присваивания в PHP это `=`. Слева от него записывается имя переменной, а справа — значение, которое должно быть ей присвоено. Например, `$a = 2; $b = 3;`

88. В каких случаях в PHP используют условное выражение (оператор if)

Условное выражение позволяет выполнить один из двух входящих в него блоков в зависимости от выполнения или невыполнения какого-либо условия. Таким условием может быть значение логической переменной или результат вычисления логического выражения.

89. Для чего в PHP используется оператор switch? Как работает этот оператор?

switch – оператор выбора. В выражении выбора значение переменной или результат вычисления выражения последовательно сравнивается со значением 1, значением 2 и т.д. и, если такое сравнение успешно, выполняется соответствующий набор выражений (первый, второй и т. д.) Если же ни одно сравнение не увенчалось успехом, выполняется набор выражений, находящийся в секции default (если она есть).

90. Назначение оператора while в PHP

Оператор while выполняет блок кода, если и до тех пор, пока условное выражение истинно. Если условное выражение истинно, тогда будет выполняться блок кода. После выполнения кода условное выражение снова будет оценено, и цикл будет продолжаться до тех пор, пока условное выражение не окажется ложным.

91. Назначение оператора do ... while в PHP

Оператор do ... while выполняет блок кода хотя бы один раз — он повторяет цикл, пока условие выполняется.

92. Назначение оператора for в PHP

Оператор for используется, когда известно, сколько раз нужно выполнить оператор или блок операторов. Для установки начального значения счетчика используется инициализатор. Для этой цели может быть объявлена переменная, ее традиционно называют \$i.

93. Назначение оператора break в PHP

Ключевое слово PHP break используется для преждевременного прекращения выполнения цикла. Оператор break находится внутри блока операторов. Он предоставляет вам полный контроль, вы можете остановить цикл, когда вам нужно. После выхода из цикла будет выполняться следующее после цикла действие.

94. Назначение оператора continue в PHP

Ключевое слово PHP continue используется для остановки текущей итерации цикла, но оно не завершает цикл. Как и оператор break, оператор continue находится внутри блока операторов, содержащих код, который выполняется циклом. Если при проходе кода, встречается оператор continue, оставшая часть кода цикла пропускается, и начинается следующий проход цикла.

95. Назначение оператора foreach в PHP

Оператор foreach используется для преобразования массивов через цикл. При каждой итерации текущему элементу массива присваивается значение \$, а указатель массива перемещается на одну позицию, и при следующем проходе будет обрабатываться следующий элемент.

96 Что представляет собой W3C (World Wide Web Consortium), в чем заключается деятельность этой организации?

W3C - Консорциум Всемирной паутины - организация, разрабатывающая и внедряющая технологические стандарты для сети Интернет. Основная цель — разрабатывать единые принципы и стандарты, называемые рекомендациями, которые ложатся в основу разработки программ и оборудования.

97. Что представляет собой язык JavaScript?

JavaScript - язык программирования, предназначенный для создания интерактивных HTML-страниц. JavaScript – встроенный в интернет браузер процедурный язык программирования. Назначение JavaScript – манипуляция элементами HTML разметки посредством программных интерфейсов объектной модели загруженной интернет страницы - DOM.

98. Что отображает метод alert языка JavaScript?

Метод alert отображает предупреждающее диалоговое окно, в котором помещается сообщение и кнопка «ОК».

99. Что отображает метод confirm языка JavaScript?

Метод confirm отображает диалоговое окно подтверждения выполнения операции. Оно содержит две кнопки – OK и Cancel (Отмена), позволяющие выбрать один из вариантов.

100. Как определяется функция в языке JavaScript?

Функция определяется оператором function . После ключевого слова function идёт имя функции, затем список параметров в круглых скобках код функции («тело функции») внутри фигурных скобок: function имя(параметры) { ...тело..}

101. Что такое функция в языке JavaScript?

Функция — это именованная последовательность операторов, которая выполняет определенную задачу и может возвращать некоторое значение.

102. Что такое обработчики событий в языке JavaScript?

Обработчики событий в языке JavaScript - это функции, которые вызываются в ответ на определенное событие, происходящее в браузере. Они позволяют программно реагировать на действия пользователя, такие как щелчок мыши, нажатие клавиши, загрузка страницы и другие события.

103. Для чего используют оператор return в языке JavaScript?

Этот оператор завершает выполнение текущей функции и возвращает её значение.

104. Где используется JavaScript

JavaScript обычно используется как встраиваемый язык для программного доступа к объектам приложений. Наиболее широкое применение находит в браузерах как язык сценариев для придания интерактивности веб-страницам.

105. Что такое библиотека (library) в программировании?

Библиотека в программировании — сборник подпрограмм или объектов для решения близких по тематике задач.

106. Назначение и возможности jQuery

jQuery— набор функций JavaScript, фокусирующийся на взаимодействии JavaScript и HTML. Библиотека jQuery помогает получать доступ к любому элементу DOM, обращаться к атрибутам и содержимому элементов DOM, манипулировать ими. Также библиотека jQuery предоставляет удобный API для работы с AJAX.

107. Где язык программирования php нашел наиболее широкое применение

Язык программирования php нашел наиболее широкое применение в Разработке web-приложений

108. Какими разделителями окружены скрипты php скрипта?

PHP-скрипт начинается <?php и заканчивается ?>

109. Что делает оператор echo "Hello, World!"

Оператор echo "Hello, World!" выводит на страницу "Hello, World!":

110. Перечислите основные типы данных в php

Основными типами данных в php являются Boolean, integer, float, string, array, object, resource, NULL.

111. Что означает динамическая типизация данных.

Динамическая типизация означает, что вам не нужно явно определять тип переменной на этапе написания кода, интерпретатор попытается самостоятельно определить тип этой переменной во время исполнения программы (динамически) исходя из его значения.

112. Что представляет собой оператор \$months = array("September", "October", "November");

В данном примере создан массив \$months из трех элементов.

113. Как в php задаются файлы cookie

В php файлы cookie задаются следующим образом: setcookie()

114. Как выглядят многострочные комментарии в php? Для чего нужны комментарии?

Для написания комментариев в пределах работающего кода на используется запись вида: /*...*/. Комментарии в PHP коде не отображаются при просмотре программы в браузере, а доступны лишь при просмотре исходного кода программы.

115. Что делает операция \$count++

\$count++ - операция инкремента, добавление 1 (единицы) к переменной \$count

116. Что может содержать \$_SERVER

В php суперглобальный массив, содержащий информацию о местоположении скриптов, путях, заголовках

117. Что делает оператор php fopen ("filename.txt", "r")

Оператор php fopen ("filename.txt", "r") который делает файл "filename.txt" доступным для чтения.

118. Какой символ указывается в конце php-оператора?

Каждый php-оператор должен заканчиваться точкой с запятой (;)

119. Дайте понятие множественного наследования в php.

В php не существует понятия «множественное наследование», т.е. у класса может быть лишь один родитель. Эмуляция множественного наследования задается функцией `_call ()` или трейтами.

120. Имеются следующие варианты написания кода: (1) `<?php echo 'Выполнить документ';?>` и (2) `<?='Выполнить документ' ?>`. Какое из приведенных ниже утверждений верное и почему?

Записи (1) и (2) представляют собой одинаковый код. Разница в том, что (1) – это стандартный синтаксис, (2) – краткий. В php приемлемы оба варианта

121. Области применения языка программирования Perl

Язык программирования Perl представляет собой символьную систему общего назначения, которая создавалась для обработки и машинного кодирования текстовых файлов. Со временем его функциональность значительно расширилась. Сейчас Perl применяется для администрирования компьютерных сетей, создания сайтов, разработки приложений, в био-информатике и создании пользовательских интерфейсов в графике.

122. Что такое прототипирование программного обеспечения (prototyping)?

Прототипирование ПО — этап разработки программного обеспечения (ПО), процесс создания прототипа программы — макета (черновой, пробной версии) программы, обычно — с целью проверки пригодности предлагаемых для применения концепций, архитектурных и/или технологических решений, а также для представления программы заказчику на ранних стадиях процесса разработки.

123. На что направленно функциональное тестирование сайта ?

Функциональное тестирование направлено на то, чтобы каждая функция веб-сайта работала в соответствии с требованиями спецификации. Тестирование функциональности веб-сайта показывает «Что делает система».

124. Особенности серверной операционной системы

Серверная ОС использует гораздо больший объем памяти для вычислений, чем обычная, а также может выполнять функции веб-сервера, сервера приложений и сервера электронной почты и многих других серверов, необходимых для работы ИТ-системы предприятия. Серверная ОС может подключать к локальной сети и к Интернет многим пользователям, а не одного, как обычная ОС. Поэтому серверная ОС и более дорогая.

125. Приведите примеры серверных ОС

В настоящее время самые популярные серверные ОС – это Microsoft Windows, Linux (Ubuntu, Debian, CentOS), Enterprise Linux от Red Hat и Free BSD.

126. Что представляет собой phpMyAdmin?

phpMyAdmin - веб-приложение с открытым кодом, написанное на языке PHP и представляющее собой веб-интерфейс для администрирования СУБД MySQL. PhpMyAdmin позволяет через браузер и не только осуществлять администрирование сервера MySQL, запускать команды

SQL и просматривать содержимое таблиц и баз данных. Приложение пользуется большой популярностью у веб-разработчиков, так как позволяет управлять СУБД MySQL без непосредственного ввода SQL команд.

127. Охарактеризовать сервер данных MySQL

MySQL — это популярный сервер данных, применяемый при создании Web-сайтов. Быстрый и нетребовательный к ресурсам компьютера сервер данных. Распространяется бесплатно. Прекрасно работает в связке с PHP - технологии создания активных серверных Web-страниц.

128. Что такое пользовательский интерфейс на веб-сайте?

Веб-интерфейс пользователя или веб-приложение позволяет пользователю взаимодействовать с контентом или программным обеспечением, работающим на удаленном сервере, через веб-браузер. Содержимое или веб-страница загружаются с веб-сервера, и пользователь может взаимодействовать с этим содержимым в веб-браузере, который действует как клиент.

129. Как работает серверная часть приложения?

Серверная часть приложения, которую иногда называют внутренней, управляет общей функциональностью веб-приложения. Когда пользователь взаимодействует с интерфейсом, он отправляет запрос в серверную часть в формате HTTP. Серверная часть обрабатывает запрос и возвращает ответ.

130. В чем заключается тестирование юзабилити прототипа сайта

Данное тестирование проводится еще на этапе прототипирования сайта. UX-дизайнер и QA-инженер проверяют, насколько пользователю удобно использовать ресурс. Тестирование отвечает на следующие вопросы: понятны ли основная цель сайта и его функционал; удобна ли навигация для пользователя; создается ли у пользователя положительное впечатление при использовании ресурса.

131. Чем занимается UX-дизайнер (user experience - «пользовательский опыт»)

UX-дизайнер - специалист, задача которого сделать продукт удобным, понятным и логичным для всех пользователей. От его работы зависит, сможет ли клиент быстро и удобно получить желаемую услугу. Перед запуском продукта он проводит аудит конкурентов, опрашивает пользователей, изучает целевую аудиторию сайта или приложения.

132. Чем занимается QA-инженер (Quality Assurance engineer)

QA-инженер это специалист, ответственный за контроль выполнения требований к продукту. Он запускает тесты и применяет другие методы для проверки качества, которое представляет собой отношение между ожиданиями и реальностью в контексте определенного продукта.

133. В чем заключается тестирование производительности

Тестирование производительности — в основном это нагрузочное тестирование. Нагрузочное тестирование сайта проверяется в большинстве случаев автоматом, то есть специальными программами. Это дает шанс проверить, как он будет работать под определенной нагрузкой.

134. В чем состоит цель нагрузочного тестирования

Цель этого тестирования, заключается в количестве виртуальных пользователей, которые задают количество запросов в одно время (возможно даже в секунду). Результат показывает, смог ли проект выдержать, к примеру, 100 пользователей, которые одновременно покупали товар или авторизовались на сайте, реально ли выдержать сайту такую нагрузку.

135. Что такое веб-программирование

Веб-программирование – направление в программировании, ориентированное на разработку приложений для сети интернет (веб-приложений). Пользователь взаимодействует с веб-приложением через интернет браузер. Само приложение загружается с удаленного веб-сервера, а общение клиента и сервера осуществляется через HTTP протокол.

136. Что такое Веб-сервис (Web service)

Веб-сервис, веб-служба– вид программного обеспечения, предоставляющего свою функциональность клиентам на уровне программных интерфейсов, построенных на базе HTTP протокола. Веб-сервис, как и любой другой ресурс всемирной паутины идентифицируется веб-адресом (URI). Веб-сервис - это частный случай веб-приложения, отличающегося тем, что не имеет пользовательского интерфейса. Клиентом веб-сервиса обычно является основное веб-приложение или другой веб-сервис, поскольку сам по себе веб-сервис - это компонент сервис-ориентированной архитектуры распределенного приложения.

137. Что такое фреймворк?

Фреймворк (Framework) – вид программного обеспечения, являющегося основой (каркасом) различных прикладных программных продуктов. Фреймворк упрощает процесс функционального объединения программных компонентов различного назначения, их взаимодействия и настройки, что позволяет быстро и качественно собрать целевой программный продукт, удовлетворяющий требованиям конкретного потребителя.

138. Что такое рефакторинг кода (Refactoring)

Рефакторинг кода (Refactoring) – процесс внесения изменений в программный код в соответствии с некоторым набором правил – приемов рефакторинга, которые не меняют смысл программы, но делают ее код более стройным и легким для интерпретации человеком. Также рефакторинг способствует облегчению поиска ошибок и “узких мест” в программе.

139. Что такое Application programming interface (API)

API – интерфейс программирования приложения – функциональность приложения, доступная на программном уровне внешним программным компонентам. API, чаще всего, представляет собой набор программных компонентов, интерфейсов, глобальных функций и т.п. API приложения используется для интеграции с другими приложениями или для расширения и настройки собственной функциональности.

140. Что представляет собой HTTP (HyperText Transfer Protocol)

HTTP – сетевой протокол прикладного уровня на базе TCP/IP, предназначенный для передачи гипертекста в условиях клиент-серверного окружения. В основном используется для обмена данными между пользователями и ресурсами в сети интернет – является средством общения распределенных компонентов всемирной паутины.

141. Что такое спецификация программы

Спецификация программы – формализованное представление требований, предъявляемых к программе, которые должны быть удовлетворены при ее разработке, а также описание задачи, условия и эффекта действия без указания способа его достижения

142. Что такое верификация программы

Верификация программы – доказательство того, что работа программы соответствует спецификации на эту программу.

143. Что такое плагин (Plug-In)

Plug-In – внешний по отношению к основному программному продукту компонент, который в ходе работы основного приложения может быть динамически к нему подключен и запущен.

144. Что представляет собой функциональная часть веб-сайта

Функциональная часть веб-сайта - совокупность программного кода и баз данных для обработки информации. Популярные языки программирования для веб-сайтов - PHP, Python, SQL, Go, Ruby, Java.

145. Что такое мобильная оптимизация веб-сайта

Мобильная оптимизация веб-сайта - оптимизация страниц веб-сайта для быстрой загрузки на мобильных устройствах и корректного отображения. Этот параметр важен для поисковых систем и влияет на восприятие информации посетителями.

146. Что представляют собой веб-фреймворки, привести примеры веб-фреймворков

Веб-фреймворки - готовые программные решения, которые позволяют разработать веб-сайт без использования CMS, для администрирования контента и внешнего представления. Такая разработка занимает существенно больше времени, чем разработка с использованием CMS, но позволяет создать свое гибкое и максимально удобное решение. Например, Laravel, Django, Yii2, Symfony.

147. Что представляет собой кэш браузера (cache), как и зачем нужно очищать кэш?

Кэш браузера (cache) - сохраненная копия данных Веб-сайта для быстрого доступа браузера к этим данным. Во время разработки веб-сайта в браузере нужно чистить кеш нажатием комбинаций клавиш CTRL+F5, чтобы увидеть изменения.

148. Что такое мобильное приложение? Назовите средства его разработки

Мобильное приложение - специально разработанное для мобильных устройств решение, которое учитывает особенности популярных мобильных устройств и их размеры экранов. Популярные языки программирования для мобильных приложений - Swift, Kotlin, Dart, React Native, Javascript. Популярные площадки для мобильных приложений - App Store от Apple для iPhone на iOS, Play Market от Google для устройств на Android.

149. Что такое хостинг?

Хостинг - хранилище файлов веб-сайта и баз данных со специальным программным обеспечением для взаимодействия с пользователями.

150. Что такое SSL-сертификат

SSL-сертификат - цифровой сертификат, который позволяет безопасно связываться с веб-сайтом. В зависимости от типа сертификата, он может быть выдан только на домен, а может быть выдан на домен и любые поддомены. Веб-сайт с сертификатом доступен по https протоколу, например, <https://www.webstripe.ru>

Активные и интерактивные формы обучения используемые при преподавании дисциплины, способствующие реализации у обучающихся навыков командной работы и т.д.

Занятия по дисциплине Web-программирование в форме *активного метода* проходят лабораторные работы по принципу занятий с элементами групповых дискуссий — это метод активного обучения, одна из организационных форм познавательной деятельности учащихся, позволяющая закрепить полученные ранее знания, восполнить недостающую информацию, сформировать уме-

ния решать проблемы, укрепить позиции, научить культуре ведения дискуссии. **Такой метод используется на лабораторной работе по теме «Основы PHP. Создание простых сценариев».**

Занятия в *интерактивной форме* по принципу *метода анализа ситуации* – это педагогическая технология, основанная на моделировании ситуации или использования реальной ситуации в целях анализа данного случая, выявления проблем, поиска альтернативных решений и принятия оптимального решения проблем. **Такие методы используются на лабораторной работе по теме «Основы PHP. Работа с СУБД MySQL. Создание базы данных и таблиц».**

4 Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Контроль результатов обучения обучающимися, этапов и уровня формирования компетенций по дисциплине «Web-программирование» осуществляется через проведение входного, текущего, рубежных, выходного контролей и контроля самостоятельной работы.

Процедура проведения зачета приведена в Положении о текущем контроле успеваемости и промежуточной аттестации.

