

МИНОБРНАУКИ РОССИИ
ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
КАФЕДРА ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И СИСТЕМ

Рабочая программа дисциплины (модуля)
ВЕБ-ПРОГРАММИРОВАНИЕ

Направление и направленность (профиль)
09.03.04 Программная инженерия. Программная инженерия

Год набора на ОПОП
2021

Форма обучения
очная

Владивосток 2023

Рабочая программа дисциплины (модуля) «Веб-программирование» составлена в соответствии с требованиями ФГОС ВО по направлению подготовки 09.03.04 Программная инженерия (утв. приказом Минобрнауки России от 19.09.2017г. №920) и Порядком организации и осуществления образовательной деятельности по образовательным программам высшего образования – программам бакалавриата, программам специалитета, программам магистратуры (утв. приказом Минобрнауки России от 06.04.2021 г. N245).

Составитель(и):

Сачко М.А., кандидат технических наук, доцент, Кафедра информационных технологий и систем, maxim.sachko@vvsu.ru

Утверждена на заседании кафедры информационных технологий и систем от 30.05.2023 , протокол № 9

СОГЛАСОВАНО:

Заведующий кафедрой (разработчика)

Кийкова Е.В.

ДОКУМЕНТ ПОДПИСАН ЭЛЕКТРОННОЙ ПОДПИСЬЮ	
Сертификат	1575633692
Номер транзакции	0000000000BBE45D
Владелец	Кийкова Е.В.

1 Цель, планируемые результаты обучения по дисциплине (модулю)

Целью освоения дисциплины «Веб-программирование» является теоретическая и практическая подготовка студентов в области разработки веб-приложений с использованием современного языка программирования PHP, СУБД MySQL, языка разметки HTML, каскадных стилей CSS, а также современных сред разработок. Знания, полученные в результате освоения дисциплины, помогут при разработке/доработке систем (приложений), основанных на CMS и PHP-фреймворках (Framework), которые используются в данной области повсеместно.

Задачи освоения дисциплины состоят в изучении архитектуры Веб, стека серверных программ, клиентских технологий (HTML, Javascript, CSS), архитектуры систем управления наполнением (CMS), современной модели веб-приложения, внешних Интернет-сервисов и их API и получении навыков программирования на языке PHP и создания приложений, основанных на базе данных (MySQL).

Планируемыми результатами обучения по дисциплине (модулю), являются знания, умения, навыки. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения образовательной программы, представлен в таблице 1.

Таблица 1 – Компетенции, формируемые в результате изучения дисциплины (модуля)

Название ОПОП ВО, сокращенное	Код и формулировка компетенции	Код и формулировка индикатора достижения компетенции	Результаты обучения по дисциплине		
			Код результата	Формулировка результата	
09.03.04 «Программная инженерия» (Б-ИН)	ПКВ-2 : Способен использовать операционные системы, сетевые технологии, средства разработки программного интерфейса, применять языки и методы формальных спецификаций, системы управления базами данных	ПКВ-2.3к : Использует сетевые технологии и СУБД	РД1	Знание	технологии разработки динамических веб-страниц и баз данных
			РД2	Умение	создавать структуру информационных сервисов и баз данных
			РД3	Навык	создания динамических информационных ресурсов

2 Место дисциплины (модуля) в структуре ОПОП

Дисциплина относится к части учебного плана, формируемой участниками образовательных отношений.

3. Объем дисциплины (модуля)

Объем дисциплины (модуля) в зачетных единицах с указанием количества академических часов, выделенных на контактную работу с обучающимися (по видам учебных занятий) и на самостоятельную работу, приведен в таблице 2.

Таблица 2 – Общая трудоемкость дисциплины

Название ОПОП ВО	Форма обучения	Часть УП	Семестр (ОФО) или курс (ЗФО, ОЗФО)	Трудо-емкость	Объем контактной работы (час)					СРС	Форма аттестации	
					(З.Е.)	Всего	Аудиторная					Внеауди-торная
				лек.			прак.	лаб.	ПА			КСР
09.03.04 Программная инженерия	ОФО	Б1.В	5	3	55	18	36	0	1	0	53	ДЗ

4 Структура и содержание дисциплины (модуля)

4.1 Структура дисциплины (модуля) для ОФО

Тематический план, отражающий содержание дисциплины (перечень разделов и тем), структурированное по видам учебных занятий с указанием их объемов в соответствии с учебным планом, приведен в таблице 3.1

Таблица 3.1 – Разделы дисциплины (модуля), виды учебной деятельности и формы текущего контроля для ОФО

№	Название темы	Код результата обучения	Кол-во часов, отведенное на				Форма текущего контроля
			Лек	Практ	Лаб	СРС	
1	Основы создания информационных ресурсов	РД1, РД2	2	4	0	6	выборочный опрос
2	Клиентские технологии веб-программирования: HTML, Javascript, CSS	РД1	2	6	0	8	выборочный опрос, отчет по практической работе
3	Серверные технологии веб-программирования. Язык PHP. Среды разработки	РД1, РД3	4	10	0	3	выборочный опрос, отчет по практической работе
4	Базы данных. Разработка приложений, основанных на базах данных	РД3	2	8	0	8	выборочный опрос, отчет по практической работе
5	Современная модель веб-приложения	РД2, РД3	2	4	0	8	выборочный опрос, отчет по практической работе
6	Системы управления контентом – CMS	РД2, РД3	2	4	0	8	выборочный опрос, отчет по практической работе
7	Веб-сервисы. Облачные технологии	РД2	2	0	0	6	выборочный опрос
8	SEO. Оптимизация веб-страниц	РД2	2	0	0	6	выборочный опрос
Итого по таблице			18	36	0	53	

4.2 Содержание разделов и тем дисциплины (модуля) для ОФО

Тема 1 Основы создания информационных ресурсов.

Содержание темы: Основы разработки веб-сайтов, интернет-магазинов, сервисов и т.д. Обзор курса. Организационная структура сети Интернет. Хостинг. Клиентские технологии: HTML, Javascript, CSS. Серверные технологии: веб-сервер Apache и NGinx, СУБД MySQL, PHP, обзор других языков: Ruby, Python, Perl. CMS. Языки разметки и структурирования информации: XML, JSON. Локальный «домашний сервер»: набор программ DENWER.

Формы и методы проведения занятий по теме, применяемые образовательные технологии: Лекция, практические занятия в компьютерном классе.

Виды самостоятельной подготовки студентов по теме: подготовка к текущей и промежуточной аттестации, практическим работам.

Тема 2 Клиентские технологии веб-программирования: HTML, Javascript, CSS.

Содержание темы: Основные возможности языка разметки HTML. Введение в Javascript, его принципиальные отличия от других языков. Javascript-библиотеки и фреймворки: JQuery, AngularJS, BackboneJS, React, Ember. Каскадные таблицы стилей CSS. Обзор различных IDE для рассмотренных технологий.

Формы и методы проведения занятий по теме, применяемые образовательные технологии: Лекция, выполнение практической работы.

Виды самостоятельной подготовки студентов по теме: изучение литературы, просмотр дополнительного видео материалов в электронном курсе.

Тема 3 Серверные технологии веб-программирования. Язык PHP. Среда разработки.

Содержание темы: Модель работы серверных программ. Взаимодействие с клиентскими программами. Синхронные и асинхронные POST и GET запросы. Язык PHP: отличия и особенности от других языков. Базовый синтаксис PHP. Библиотеки функций. Среда разработки.

Формы и методы проведения занятий по теме, применяемые образовательные технологии: Лекция, выполнение практической работы.

Виды самостоятельной подготовки студентов по теме: изучение литературы, просмотр дополнительного видео материалов в электронном курсе.

Тема 4 Базы данных. Разработка приложений, основанных на базах данных.

Содержание темы: Краткое введение в базы данных (БД). Реляционная модель данных. Язык SQL для работы с БД. MySQL и PostgreSQL. IDE для работы с БД. Расширение PDO для интерпретатора PHP для работы с БД.

Формы и методы проведения занятий по теме, применяемые образовательные технологии: Лекция, выполнение практической работы.

Виды самостоятельной подготовки студентов по теме: изучение литературы, просмотр дополнительного видео материалов в электронном курсе.

Тема 5 Современная модель веб-приложения.

Содержание темы: Подход разделения данных, логики и представления в веб-приложении («Модель-Вид-Поведение» - MVC). Язык Smarty. Системы управления контентом - CMS (введение). Системы контроля версий (CVS). Системы управления проектами: Jira и другие.

Формы и методы проведения занятий по теме, применяемые образовательные технологии: Лекция, выполнение практической работы.

Виды самостоятельной подготовки студентов по теме: изучение литературы, просмотр дополнительного видео материалов в электронном курсе.

Тема 6 Системы управления контентом – CMS.

Содержание темы: Возможности CMS. Применение CMS в различных областях деятельности. Принципы, на основе которых разрабатываются CMS. Обзор CMS Joomla, WordPress и некоторых других. Плагины и шаблоны для CMS. Описание модели, обсуждение реализации подхода MVC, используемого в рассматриваемых CMS.

Формы и методы проведения занятий по теме, применяемые образовательные технологии: Лекция, выполнение практической работы.

Виды самостоятельной подготовки студентов по теме: изучение литературы, просмотр дополнительного видео материалов в электронном курсе.

Тема 7 Веб-сервисы. Облачные технологии.

Содержание темы: Обзор идеи веб-сервисов (как программных продуктов). Облачные технологии. Доступ и использование API сторонних платформ и веб-сервисов в своих веб-проектах. Клиентское и серверное взаимодействие с «чужим» сервером (сервисом).

Формы и методы проведения занятий по теме, применяемые образовательные технологии: Лекция.

Виды самостоятельной подготовки студентов по теме: изучение литературы, просмотр дополнительного видео материалов в электронном курсе.

Тема 8 SEO. Оптимизация веб-страниц.

Содержание темы: Обзор современных методов SEO-оптимизации для улучшения продвижения разработанных веб-сайтов и веб-приложений в сети Интернет.

Формы и методы проведения занятий по теме, применяемые образовательные технологии: Лекция.

Виды самостоятельной подготовки студентов по теме: изучение литературы, просмотр дополнительного видео материалов в электронном курсе.

5 Методические указания для обучающихся по изучению и реализации дисциплины (модуля)

5.1 Методические рекомендации обучающимся по изучению дисциплины и по обеспечению самостоятельной работы

В ходе изучения дисциплины «Веб-программирование» студенты могут посещать аудиторные занятия (лекции, практические занятия, консультации). Особенность изучения дисциплины «Веб-программирование» состоит в выполнении комплекса практических работ, главной задачей которого является получение навыков программирования и использования современных веб-технологий для решения различных профессиональных задач в области Интернет-продвижения предприятия.

Особое место в овладении частью тем данной дисциплины может отводиться самостоятельной работе, при этом во время аудиторных занятий могут быть рассмотрены и проработаны наиболее важные и трудные вопросы по той или иной теме дисциплины, а второстепенные и более легкие вопросы, а также вопросы, специфичные для направления подготовки, могут быть изучены студентами самостоятельно.

В соответствии с учебным планом направления подготовки процесс изучения дисциплины может предусматривать проведение лекций, практических занятий, консультаций, а также самостоятельную работу студентов. Обязательным является проведение практических занятий в специализированных компьютерных аудиториях, оснащенных подключенными к центральному серверу терминалами или персональными компьютерами с установленным на них локальным веб-сервером.

Ниже перечислены предназначенные для самостоятельного изучения студентами те вопросы из лекционных тем, которые во время проведения аудиторных занятий изучаются недостаточно или изучение которых носит обзорный характер.

Тема 1 Введение в веб-программирование.

История возникновения Интернета и веб-программирования. Стек протоколов (HTTP, TCP/IP и другие).

Тема 2. Серверные технологии веб-программирования. Язык PHP. Среды разработки.

Языки программирования, альтернативные PHP: Perl, Ruby, Java, Python и другие.

Тема 3. Базы данных. Разработка приложений, основанных на БД.

Альтернативные для реляционной модели данных. Другие СУБД: ORACLE, ACCESS, MSSQL и другие.

Тема 4. Клиентские технологии веб-программирования: HTML, Javascript, CSS.

Множество специализированных функций и библиотек Javascript.

Тема 5. Современная модель веб-приложения.

Шаблонизаторы. Smarty.

Тема 6. Системы управления контентом – CMS.

Специализированные CMS.

Тема 7. Веб-сервисы. Облачные технологии.

Другие распространенные веб-АПИ: Google Map, Twitter, Instagramm и т.д.

Тема 8. SEO. Оптимизация веб-страниц

Интернет-маркетинг.

Результаты самостоятельной работы по дисциплине проверяются с помощью тестирования в электронном курсе, так же они могут быть проверены на зачете при ответах на вопросы, вынесенные на самостоятельное изучение.

5.2 Особенности организации обучения для лиц с ограниченными возможностями здоровья и инвалидов

При необходимости обучающимся из числа лиц с ограниченными возможностями здоровья и инвалидов (по заявлению обучающегося) предоставляется учебная информация в доступных формах с учетом их индивидуальных психофизических особенностей:

- для лиц с нарушениями зрения: в печатной форме увеличенным шрифтом; в форме электронного документа; индивидуальные консультации с привлечением тифлосурдопереводчика; индивидуальные задания, консультации и др.

- для лиц с нарушениями слуха: в печатной форме; в форме электронного документа; индивидуальные консультации с привлечением сурдопереводчика; индивидуальные задания, консультации и др.

- для лиц с нарушениями опорно-двигательного аппарата: в печатной форме; в форме электронного документа; индивидуальные задания, консультации и др.

6 Фонд оценочных средств для проведения текущего контроля и промежуточной аттестации обучающихся по дисциплине (модулю)

В соответствии с требованиями ФГОС ВО для аттестации обучающихся на соответствие их персональных достижений планируемым результатам обучения по дисциплине (модулю) созданы фонды оценочных средств. Типовые контрольные задания, методические материалы, определяющие процедуры оценивания знаний, умений и навыков, а также критерии и показатели, необходимые для оценки знаний, умений, навыков и характеризующие этапы формирования компетенций в процессе освоения образовательной программы, представлены в Приложении 1.

7 Учебно-методическое и информационное обеспечение дисциплины (модуля)

7.1 Основная литература

1. Асалханов П. Г. Web-программирование: JavaScript : Информатика [Электронный ресурс] : Иркутский государственный аграрный университет имени А.А. Ежевского , 2020 - 123 - Режим доступа: <https://e.lanbook.com/book/183488>

2. Никулова Г. А. Web-программирование : Учебники [Электронный ресурс] - Липецк : Липецкий государственный педагогический университет имени П.П. Семенова-Тян-Шанского , 2017 - 62 - Режим доступа: http://biblioclub.ru/index.php?page=book_red&id=577453

3. Основы веб-программирования : Учебное пособие [Электронный ресурс] : Новосибирский государственный технический университет , 2020 - 64 - Режим доступа: <https://znanium.com/catalog/document?id=396977>

7.2 Дополнительная литература

1. Казаков. «Основы работы в MySQL» [Электронный ресурс] , 2012 - 128 - Режим доступа: <https://lib.rucont.ru/efd/210596>

2. Практика ЭВМ - HTML / Д.В. Груздев .— Воронеж : Издательский дом ВГУ, 2017 .— 108 с. — 108 с. — URL: <https://lib.rucont.ru/efd/670081> (дата обращения: 16.02.2024)

3. Справочник по HTML – <http://htmlbook.ru/>

4. Стефанова, И.А. Изучение языка гипертекстовой разметки HTML5 : задания и метод. указания к лаб. работам по информатике для студентов телекоммуникационных и экономических направлений: 38.03.05 - Бизнес-информатика, 38.03.02 - Информационный менеджмент, 09.03.05 - Прикладная информатика / И.А. Стефанова .— Самара : Изд-во ПГУТИ, 2015 .— 55 с. — URL: <https://lib.rucont.ru/efd/319861> (дата обращения: 16.02.2024)

7.3 Ресурсы информационно-телекоммуникационной сети "Интернет", включая профессиональные базы данных и информационно-справочные системы (при

необходимости):

1. PHP, MySQL и другие веб-технологии – <http://www.php.su/>
2. Интерактивные онлайн-курсы по HTML и CSS – <https://htmlacademy.ru/>
3. Маклафлин Б. PHP и MySQL. Исчерпывающее руководство. – СПб.: Питер, 2013. – 512 с.
[Электронный ресурс] Режим доступа: <http://padabum.com/d.php?id=41776>
4. СПС КонсультантПлюс <http://www.consultant.ru/>
5. Электронная библиотечная система «РУКОНТ» - Режим доступа: <https://lib.rucont.ru/>
6. Электронная библиотечная система «Университетская библиотека онлайн» - Режим доступа: <http://biblioclub.ru/>
7. Электронная библиотечная система ZNANIUM.COM - Режим доступа: <https://znanium.com/>
8. Электронно-библиотечная система "РУКОНТ"
9. Электронно-библиотечная система издательства "Лань" - Режим доступа: <https://e.lanbook.com/>
10. Open Academic Journals Index (ОАИ). Профессиональная база данных - Режим доступа: <http://oaji.net/>
11. Президентская библиотека им. Б.Н.Ельцина (база данных различных профессиональных областей) - Режим доступа: <https://www.prlib.ru/>

8 Материально-техническое обеспечение дисциплины (модуля) и перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень программного обеспечения

Основное оборудование:

- Компьютеры
- Проектор
- Экран Projecta 160*160

Программное обеспечение:

- Microsoft Office 2010 Standart
- Microsoft Windows 8 - KMS
- SQL Server
- Windows Web Server

МИНОБРНАУКИ РОССИИ

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

КАФЕДРА ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И СИСТЕМ

Фонд оценочных средств
для проведения текущего контроля
и промежуточной аттестации по дисциплине (модулю)

ВЕБ-ПРОГРАММИРОВАНИЕ

Направление и направленность (профиль)
09.03.04 Программная инженерия. Программная инженерия

Год набора на ОПОП
2021

Форма обучения
очная

Владивосток 2023

1 Перечень формируемых компетенций

Название ОПОП ВО, сокращенное	Код и формулировка компетенции	Код и формулировка индикатора достижения компетенции
09.03.04 «Программная инженерия» (Б-ИН)	ПКВ-2 : Способен использовать операционные системы, сетевые технологии, средства разработки программного интерфейса, применять языки и методы формальных спецификаций, системы управления базами данных	ПКВ-2.3к : Использует сетевые технологии и СУБД

Компетенция считается сформированной на данном этапе в случае, если полученные результаты обучения по дисциплине оценены положительно (диапазон критериев оценивания результатов обучения «зачтено», «удовлетворительно», «хорошо», «отлично»). В случае отсутствия положительной оценки компетенция на данном этапе считается несформированной.

2 Показатели оценивания планируемых результатов обучения

Компетенция ПКВ-2 «Способен использовать операционные системы, сетевые технологии, средства разработки программного интерфейса, применять языки и методы формальных спецификаций, системы управления базами данных»

Таблица 2.1 – Критерии оценки индикаторов достижения компетенции

Код и формулировка индикатора достижения компетенции	Результаты обучения по дисциплине			Критерии оценивания результата в обучения
	Код результата	Тип результата	Результат	
ПКВ-2.3к : Использует сетевые технологии и СУБД	РД 1	Знание	технологии разработки динамических веб-страниц и баз данных	сформировавшееся знание основных методов и подходов разработки динамических веб-сайтов
	РД 2	Умение	создавать структуру информационных сервисов и баз данных	сформировавшееся умение формировать структуру веб-страниц сайтов и баз данных
	РД 3	Навык	создания динамических информационных ресурсов	сформировавшееся навыки создания динамических информационных ресурсов с использованием языков веб-программирования

Таблица заполняется в соответствии с разделом 1 Рабочей программы дисциплины (модуля).

3 Перечень оценочных средств

Таблица 3 – Перечень оценочных средств по дисциплине (модулю)

Контролируемые планируемые результаты обучения	Контролируемые темы дисциплины	Наименование оценочного средства и представление его в ФОС	
		Текущий контроль	Промежуточная аттестация

Очная форма обучения						
РД1	Знание : технологии разработки динамических веб-страниц и баз данных	1.1. Основы создания информационных ресурсов	Опрос	Тест		
			Практическая работа	Тест		
		1.2. Клиентские технологии и веб-программирования: HTML, Javascript, CSS	Опрос	Тест		
			Практическая работа	Тест		
		1.3. Серверные технологии веб-программирования. Язык PHP. Среда разработки	Опрос	Тест		
			Практическая работа	Тест		
РД2	Умение : создавать структуру информационных сервисов и баз данных	1.1. Основы создания информационных ресурсов	Опрос	Тест		
			Практическая работа	Тест		
		1.5. Современная модель веб-приложения	Опрос	Тест		
			Практическая работа	Тест		
		1.6. Системы управления контентом – CMS	Опрос	Тест		
			Практическая работа	Тест		
		1.7. Веб-сервисы. Облачные технологии	Опрос	Тест		
			Практическая работа	Тест		
		1.8. SEO. Оптимизация веб-страниц	Опрос	Тест		
			Практическая работа	Тест		
		РД3	Навык : создания динамических информационных ресурсов	1.3. Серверные технологии веб-программирования. Язык PHP. Среда разработки	Опрос	Тест
					Практическая работа	Тест
1.4. Базы данных. Разработка приложений, основанных на базах данных	Опрос			Тест		
	Практическая работа			Тест		
1.5. Современная модель веб-приложения	Опрос			Тест		
	Практическая работа			Тест		
1.6. Системы управления контентом – CMS	Опрос			Тест		
	Практическая работа			Тест		

4 Описание процедуры оценивания

Качество сформированности компетенций на данном этапе оценивается по результатам текущих и промежуточных аттестаций при помощи количественной оценки, выраженной в баллах. Максимальная сумма баллов по дисциплине (модулю) равна 100 баллам.

Вид учебной деятельности	Оценочное средство			
	Компьютерный тест	Отчет по практическому заданию	Выборочный опрос	Итого
Лекции			5	5
Практические занятия		60		60
Промежуточная аттестация	20			20
Самостоятельная работа		10	5	15
Итого	20	70	10	100

Сумма баллов, набранных студентом по всем видам учебной деятельности в рамках дисциплины, переводится в оценку в соответствии с таблицей.

Сумма баллов по дисциплине	Оценка по промежуточной аттестации	Характеристика качества сформированности компетенции
от 91 до 100	«зачтено» / «отлично»	Студент демонстрирует сформированность дисциплинарных компетенций, обнаруживает всестороннее, систематическое и глубокое знание учебного материала, усвоил основную литературу и знаком с дополнительной литературой, рекомендованной программой, умеет свободно выполнять практические задания, предусмотренные программой, свободно оперирует приобретенными знаниями, умениями, применяет их в ситуациях повышенной сложности.
от 76 до 90	«зачтено» / «хорошо»	Студент демонстрирует сформированность дисциплинарных компетенций: основные знания, умения освоены, но допускаются незначительные ошибки, неточности, затруднения при аналитических операциях, переносе знаний и умений на новые, нестандартные ситуации.
от 61 до 75	«зачтено» / «удовлетворительно»	Студент демонстрирует сформированность дисциплинарных компетенций: в ходе контрольных мероприятий допускаются значительные ошибки, проявляется отсутствие отдельных знаний, умений, навыков по некоторым дисциплинарным компетенциям, студент испытывает значительные затруднения при оперировании знаниями и умениями при их переносе на новые ситуации.
от 41 до 60	«не зачтено» / «неудовлетворительно»	У студента не сформированы дисциплинарные компетенции, проявляется недостаточность знаний, умений, навыков.
от 0 до 40	«не зачтено» / «неудовлетворительно»	Дисциплинарные компетенции не сформированы. Проявляется полное или практически полное отсутствие знаний, умений, навыков.

5 Примерные оценочные средства

5.1 Примерные темы для опроса

- 1) Приведите пример html-тега с пятью атрибутами.
- 2) Напишите пример внутренней гиперссылки.
- 3) Перечислите облачные сервисы, которые вы знаете?
- 4) Напишите, какие виды селекторов вы знаете.
- 5) Зачем нужны свойства padding и margin? В чем разница между ними?
- 6) Приведите пример внутреннего стиля.
- 7) Как с помощью CSS разместить нужный html-элемент в нужном месте, заданном координатами (x: 200, y: 300)?
- 8) Какими способами можно создать переменную в JS? Какие типы переменных существуют в JS?

- 9) Создайте массив из элементов «Яблоко», «Груша», «Слива» двумя разными способами.
- 10) Дана строка: «Привет, мир!». Как узнать, с какой позиции в данной строке начинается слово «мир»?
- 11) Как узнать, что на кнопку нажали, чтобы вывести пользователю сообщение «привет»? Напишите соответствующий код.
- 12) Что такое BOM? Напишите, что знаете об этом и зачем это нужно.
- 13) Напишите на PHP функцию вычисления суммы двух чисел. Приведите пример использования этой функции.
- 14) Как работает область видимости переменных в PHP? Приведите пример создания и использования глобальной переменной.
- 15) Каким образом на PHP можно извлечь полученные параметры из пришедшего запроса? Приведите разные способы.
- 16) Какие строковые функции в PHP вы знаете? Просто перечислите.
- 17) Какие вы знаете СУПЕР-глобальные переменные в PHP?
- 18) Напишите SQL-запрос для вставки в таблицу Users строки со значениями полей: Имя=Саша, Рост=170.
- 19) Напишите SQL-запрос для выборки из таблицы Users всех пользователей, у которых рост > 170.
- 20) С помощью какой команды в PHP создаются запросы к MySQL? Как можно с помощью PHP узнать количество полученных строк из SELECT запроса к MySQL?

Краткие методические указания

Выборочный опрос по проведенной лекции проходит в конце лекций, за 20 минут до ее окончания. Выборочный опрос по самостоятельной работе проходит в начале лекций. Студенты отвечают с места, либо вызываются к доске (в зависимости от содержания вопроса) и им предлагается ответить на один из выше описанных вопросов, а также продемонстрировать ответ с помощью специализированных программ на компьютере, подключенном к проектору, либо на доске. Каждый ответ студента оценивается по пятибалльной шкале.

Шкала оценки

№	Баллы	Описание
5	5	Студент полностью ответил на вопрос.
4	4	Студент ответил на вопрос, но допустил одну незначительную ошибку.
3	2-3	Студент ответил на вопрос, но допустил несколько ошибок.
2	1	Студент показал общую осведомленность о данной теме, но не смог ответить на вопрос.
1	0	Студент не смог ответить на поставленный вопрос или отказался отвечать.

5.2 Примеры заданий для выполнения практических работ

1. Создать с помощью HTML простейший учебный «интернет-магазин» (сайт): Сайт должен состоять из трех страниц: главная страница, страница контактов, страница с каталогом товаров. Каждая из этих страниц должна иметь:

- 1) название магазина крупными буквами
- 2) логотип магазина (картинка)
- 3) меню: ссылки на другие страницы сайта (т.е., например, на главной странице должны быть ссылки на «контакты» и на «каталог товаров»).

Главная страница должна содержать текстовое описание магазина и его товаров. Страница контактов должна содержать информацию о контактах (выдуманный адрес, Е-мэйл). Страница с каталогом товаров должна содержать таблицу товаров, содержащую 3 поля: название товара, картинка товара и цена товара. В таблице товаров должно быть минимум 3 строки (с тремя различными товарами). Под таблицей товаров надо сделать кнопку «купить».

2. Написать программу, которая будет содержать две кнопки и квадрат (div). При щелчке по первой кнопке квадрат должен окраситься в красный цвет. При наведении мышкой на вторую кнопку нужно проверить, покрашен ли квадрат в красный цвет. Если покрашен, то покрасить в зеленый.

3. Написать форму для ввода строки и двух символов. Извлеките строку и исследуемые символы из параметров запроса. Узнать, каких символов в строке больше: первых или вторых. Выполнить в виде функции, которая будет на вход получать строку, первый символ и второй символ. В ответ возвращать символ, который встречается в строке чаще.

4. Пусть в текстовом файле содержится 10 названий товаров, их цвета и цены, указанных в строках и разделенных пробелом. Например: Стул Черный 1000 Стол Коричневый 2000 Задание. Прочитать в программе этот файл. Загрузить все строки товаров. Построить таблицу, используя эти данные.

5. Создать отдельную html-страничку с формой регистрации пользователя: сделать текстовое поле для ввода имени, текстовое поле для ввода роста, поле select с тремя вариантами городов (включая Владивосток). Сделать кнопку «Зарегистрировать», по нажатию на которую должен вызываться ваш скрипт php (другой, без этой формы). С помощью скрипта использовать полученные из формы данные о пользователе и внести его в таблицу users.

6. Воспользоваться php и mysql для создания таблицы товаров интернет-магазина. Для этого создать новую таблицу «товары» с полями «Название», «Цена», «Картинка». В качестве картинок пусть будут храниться названия графических файлов (ссылки) на нужные изображения. Необходимо сделать еще одну страничку сайта (админскую) для ввода новых товаров. Сделать форму для ввода данных о товаре, включая выбор файла картинки для каждого товара! Страничка «Каталог товаров» должна быть переписана на php. На php необходимо извлекать из базы данных необходимую информацию о товарах и строить для них таблицу (каталог).

7. Добавить новые параметры для товаров: фирма, год (выпуска). Добавить в каталог (страничка сайта) поле select, в котором необходимо отобразить все возможные на данный момент года выпусков товаров (в соответствии с имеющимися в базе данных). Добавить кнопку «отобразить». По нажатию на эту кнопку каталог должен отображать только те товары, которые соответствуют указанному году выпуска. Сделать аналогичную кнопку для выбора фирмы.

8. Добавить кнопку «редактировать» в виде карандашика в каждой строке таблицы товаров. При нажатии на данную кнопку данная строка таблицы должна меняться так, чтобы дать возможность пользователю ввести данные, т.е. вместо строки с названием должно появиться текстовое поле, содержащее в себе текущее название товара, аналогично для других параметров товара. А также надо добавить в эту строчку кнопку «обновить». При нажатии на данную кнопку, на скрипт php должна отправляться изменившаяся информация о товаре. Скрипт должен получить эту информацию и обновить в соответствии с ней строчку в базе данных, которая содержит информацию об указанном товаре.

9. Сделать на сайте магазина кнопку «логин». При нажатии на нее с помощью Javascript должно появляться всплывающее окошко с текстовыми полями «логин» и «пароль» и кнопкой «войти». По нажатию на эту кнопку с помощью Javascript проверить длину имени и пароля. Логин не должен быть менее 3-х символов. Пароль не должен быть менее 6 символов. Если эти условия не выполнены отобразить красным цветом сообщение об ошибке. Если эти условия выполнены, то на сервер должен отправляться AJAX-запрос с введенными данными. На сервере нужно проверить, есть ли уже такой пользователь. Если нет, тогда зарегистрировать в базе данных и вернуть идентификатор пользователя. Если пользователь уже есть, тогда надо проверить его пароль. Если пароль совпадает, то вернуть идентификатор пользователя. Если нет, то вернуть «error». На сайте проверить результат запроса. Если получен идентификатор пользователя, то закрыть окошко логина и справа вверху сайта отобразить текстовую строку, содержащую имя пользователя. Если получено «error», то вывести красным цветом в окошке логина «неверный пароль». Полученный идентификатор пользователя необходимо сохранить в переменной Javascript, он пригодится дальше.

10. Добавить таблицу «Покупки». Связать ее с таблицей «Товары» с помощью идентификатора товара и с таблицей «Пользователи» (Покупатели) с помощью идентификатора пользователя. Добавить в структуру таблицы «Товары» новое поле «Описание», чем-нибудь заполнить его в базе. В таблице товаров на сайте выводить описание не надо. Надо сделать так, чтобы по щелчку мыши на товаре в таблице (в любом месте соответствующей строки товара) с помощью Javascript появлялось всплывающее окошко со всей информацией о выбранном товаре, включая новое «описание» и новую кнопку «купить». По нажатию на эту кнопку на сервер должен отправляться AJAX-запрос на покупку выбранного товара. На сервер должна отправиться информация: идентификатор товара и идентификатор пользователя, который сейчас авторизовался на сайте. В базе данных в таблице «Покупки» должна быть добавлена новая строка с полученной информацией.

Краткие методические указания

На выполнение одной практической работы отводится не менее одного двухчасового занятия (включая затраты времени на проведение промежуточного теста на последнем в учебном периоде практических занятии). После выполнения каждой практической работы студент должен представить отчет о ее выполнении, а также, по указаниям преподавателя, выполнить дополнительные задания по данной теме работы.

Шкала оценки

№	Баллы	Описание
5	64–70	Студент демонстрирует умения на итоговом уровне: умеет свободно выполнять практические задания, предусмотренные программой, свободно оперирует приобретенными умениями, применяет их в ситуациях повышенной сложности.
4	52–63	Студент демонстрирует умения на среднем уровне: освоил основные умения, но допускаются незначительные ошибки, неточности, затруднения при аналитических операциях, переносе умений на новые, нестандартные ситуации.
3	41–51	Студент демонстрирует умения и навыки на базовом уровне: в ходе контрольных мероприятий допускаются значительные ошибки, проявляется отсутствие отдельных умений, навыков по дисциплинарной компетенции, испытываются значительные затруднения при оперировании умениями и при их переносе на новые ситуации.
2	21–40	Студент демонстрирует умения и навыки на уровне ниже базового: проявляется недостаточность умений и навыков.
1	0–20	Студентом проявляется полное или практически полное отсутствие умений и навыков.

5.3 Примеры тестовых заданий

1. Маше хочется, чтобы при наведении мыши на изображение, появлялась всплывающая подсказка с текстом “Подсказка”. Какой валидный код XHTML 1.1 для этого используется?

```
  
  
  

```

2. Какой тег нужно добавить для переноса строки, сохранив при этом валидность XHTML 1.1?

```
<hr />  
<hr >  
<br />  
<br >
```

3. Игорю нужно сделать ширину таблицы на всю страницу (либо родительского контейнера). Как ему нужно написать тег <table >?

```
<table >  
<table width="100%" >  
<table width="auto" >  
<table width="100" >
```

4. Размер окна браузера 1000 пикселей. На страницу добавили блок с шириной 40%. Затем в этот блок добавили таблицу с шириной 50%. Какова будет ширина таблица в пикселях?

Недостаточно данных
250 пикселей
500 пикселей
200 пикселей

5. Какой из представленных вариантов является валидным по XHTML 1.1?

```
  
  
  

```

6. На сайте внутри папки pages находится файл page.html. А внутри папки images находится файл foto.jpg. Причём папки images и pages лежат в корне сайта. Как правильно написать путь к foto.jpg из файла page.html?

pages/images/foto.jpg
../images/foto.jpg
../images/pages/foto.jpg
images/foto.jpg

7. Какой из представленных ниже HTML-кодов НЕ является валидным XHTML 1.1?

< u >Подчёркнутый текст< /u >
< i >Курсив< /i >
< p >Текст< /p >
< p style="font-size: 1000%;" >Текст< /p >

8. Какой из вариантов содержит ошибку?

< a href="page.html" >Ссылка
< a href="page.html#17" >Ссылка
< a href="page.html"#top >Ссылка
< a href="page.html#top" >Ссылка

9. Александру требуется написать химическую формулу тетрасульфида димышьяка (As₂S₄).

Каким образом это можно сделать?

As< sup >2S< sup >4
As< pow >2S< pow >4
As< sub >2S< sub >4
As< sup >2S< sup >4

10. Как правильно задать ссылку на адрес электронной почты?

< a href="email:myrusakov@gmail.com" >Написать
< a href="myrusakov@gmail.com" >Написать
< a href="mail:myrusakov@gmail.com" >Написать
< a href="mailto:myrusakov@gmail.com" >Написать

11. Задаёт области картинки, которые могут стать ссылками. Описывается внутри тега < map >.

Картинки с такими ссылками называются «картой изображений»:

< area >
< map >
< form >
< head >

12. С помощью тега ... можно создавать ссылки и указывать места на странице, куда именно необходимо осуществить переход (якорь):

< div >
< a >
< br >
< colgroup >

13. Этот элемент указывает «подвал» сайта. Зачастую используется с тегами < header > и < article

>:

< header >
< article >
< footer >
< form >

14. Элемент описывает заголовок первого уровня:

< h1 >
< isindex >
< li >
< h2 >

15. Элемент ... служит для отображения информации об авторе. Он может включать в себя любые теги HTML. По умолчанию текст внутри этого элемента отображается курсивным начертанием:

< address >
< canvas >
< /address >
< dd >

16. Определяет тип страницы. Такой элемент необходимо задавать самым первым в начале файла, чтобы браузер знал, как правильно отображать данные (в соответствии с какими стандартами):

< aside >
< !- - >
< !DOCTYPE >
< article >

17. Данный элемент производит выравнивание своего содержимого по горизонтали по центру:

< center >
< h1 >
< div >
< header >

18. Этот тег задается внутри блока < head > и указывает для браузера полный базовый адрес текущего документа. Он применяется для документов с относительной адресацией, позволяя корректно определить адреса ссылок, даже если эта страница была перемещена:

< head >
< target >
< body >
< base >

19. Используется для хранения содержимого страницы, которое отображается в окне браузера. Используется только раз и должен находиться в теге < html >:

< ol >
< meta >
< isindex >
< body >

20. Содержит уточняющую информацию о странице:

< head >
< nav >
< pre >
< body >

Краткие методические указания

Промежуточный тест проводится в электронной форме во время последнего в учебном периоде практических занятия. Тест состоит из 20 тестовых заданий выбранных случайным образом из базы вопросам по всем темам, включая темы для самостоятельного изучения. На выполнение теста отводится 20 минут. Во время проведения теста использование литературы и других информационных ресурсов допускается только по предварительному согласованию с преподавателем.

Шкала оценки

№	Баллы	Описание
5	19–20	Процент правильных ответов от 95% до 100%
4	16–18	Процент правильных ответов от 80 до 94%
3	13–15	Процент правильных ответов от 65 до 79%
2	9–12	Процент правильных ответов от 45 до 64%
1	0–8	Процент правильных ответов менее 45%