

РОСЖЕЛДОР

**Федеральное государственное бюджетное
образовательное учреждение высшего образования
"Ростовский государственный университет путей сообщения"
(ФГБОУ ВО РГУПС)**

Кафедра " Вычислительная техника и автоматизированные системы управления "

**УЧЕБНО-МЕТОДИЧЕСКИЕ УКАЗАНИЯ
ПО ПОДГОТОВКЕ К ЭКЗАМЕНУ
ПРОФЕССИОНАЛЬНОГО МОДУЛЯ**

ПМ.04 РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЯ НА СТОРОНЕ СЕРВЕРА

Для специальности среднего профессионального образования
09.02.09 Веб-разработка

Автор-составитель: Игнатьева О.В.

Ростов-на-Дону
2025 г.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ О ФОРМАХ ПОДГОТОВКИ К ЭКЗАМЕНУ И КРИТЕРИЕВ ОЦЕНКИ	5
КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ	24
ОСВОЕНИЯ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ	24
ЗАКЛЮЧЕНИЕ	28
ИНФОРМАЦИОННЫЕ ИСТОЧНИКИ	29

ВВЕДЕНИЕ

Учебно-методическое пособие по подготовке к экзамену профессионального модуля ПМ.04 «Разработка веб-приложения на стороне сервера» составлено на основании и в соответствии с ФГОС СПО и рабочей программы, которая является частью ОПОП по специальности 09.02.09 Веб-разработка.

Учебно-методическое пособие рассчитано на базовую подготовку обучающихся среднего профессионального образования и содержит основные требования и рекомендации по подготовке к экзамену по освоению дисциплины.

Основной целью учебно-методического пособия является оказание методической помощи по организации учебной работы обучающихся для расширения, углубления и закрепления знаний и умений, а также формирования общих (ОК) и профессиональных (ПК) компетенций в определенных ФГОС СПО видах деятельности:

Выполнение проектирования и разработки информационных ресурсов; сформировать первоначальные практические профессиональные умения системного анализа, построения концептуальных моделей информационных ресурсов, разработки тестовых сценариев программного средства, тестирования информационного ресурса и работы с системой контроля версий.

Перечень профессиональных компетенций

Код	Наименование вида деятельности, профессиональных компетенций
ВД	Проектирование и разработка информационных ресурсов
ПК 4.1	Администрировать среды и платформы разработки информационных ресурсов
ПК 4.2	Создавать программный код на стороне сервера в соответствии с техническим заданием (спецификацией) с использованием языков программирования, библиотек и фреймворков
ПК 4.3	Осуществлять отладку программного кода на стороне сервера на уровне программных модулей, межмодульных взаимодействий и взаимодействий с окружением

В результате освоения профессионального модуля обучающийся должен

знать	ПК-4.1 сетевые протоколы и основы веб-технологий; современные серверные языки программирования, фреймворки и библиотеки для разработки веб-приложений, принципы организации сред разработки; технологии сред развертывания и функционирования веб-приложений (контейнеризация, оркестрация, виртуализация и облака); современные стандарты взаимодействия компонентов распределенных приложений; технологии защиты и безопасности веб-ресурсов на стороне сервера; базовые операции администрирования баз данных ПК-4.2 архитектурные модели серверных приложений; синтаксис и основные возможности серверных языков программирования (например, Python, JavaScript)
-------	---

	<p>(Node.js), PHP, Ruby и др.) с использованием фреймворков (например, Django, NestJS, Laravel и др.) и специальных библиотек; основы баз данных: реляционных (например, PostgreSQL, MySQL и др.) и нереляционных (например, MongoDB, Redis и др.); принципы работы с базами данных на стороне сервера: подключение, выполнение запросов, ORM-технология; методы обработки HTTP-запросов и формирования HTTP-ответов; архитектурный шаблон взаимодействия «клиент (браузер) – сервер – база данных»; язык структурированных запросов (SQL) для операций манипуляции данными; основы аутентификации и авторизации в веб-приложениях; основы архитектуры RESTful API, принципы шаблонизации и разделения кода (MVC);</p> <p>принципы построения защищенных веб-приложений: защита от SQL-инъекций, валидация и экранирование пользовательского ввода; основы реализации бизнес-логики типовых веб-модулей: аутентификации (хеширование паролей), CRUD-операций, пагинации, простых отчетов;</p> <p>ПК-4.3</p> <p>классификация ошибок серверного кода: синтаксические, логические, ошибки времени выполнения; методы и инструменты отладки для различных серверных языков; методы диагностики проблем взаимодействия модулей: трассировка вызовов, профилирование; особенности отладки взаимодействия с окружением: переменные среды, конфигурационные файлы, права доступа; подходы к отладке асинхронного и параллельного кода; методы анализа и оптимизации производительности на уровне кода и запросов; типы и форматы сообщений об ошибках, предупреждений; классификацию ошибок в веб-разработке: синтаксические ошибки, ошибки выполнения (исключения), логические ошибки в SQL, ошибки подключения к БД; методы и инструменты диагностики: встроенные сообщения об ошибках в языках программирования и СУБД, вывод отладочной информации, анализ журналов ошибок; методы диагностики проблем взаимодействия: проверка подключения к БД, анализ передаваемых параметров (ID, данные форм), проверка состояния сессий; способы использования технологических журналов, форматы и типы записей журналов; современные компиляторы, отладчики и оптимизаторы программного кода</p>
уметь	<p>ПК-4.1</p> <p>использовать технологии среды развертывания и функционирования веб-приложений; подбирать и настраивать конфигурацию веб-сервера; администрировать веб-сервера, настраивать и мониторить серверную инфраструктуру; использовать методы защиты на стороне сервера, реализовывать механизмы аутентификации и авторизации; использовать современные Фреймворки и библиотеки для разработки серверных скриптов; применять скрипты для реализации серверной логики; применять принципы работы с облачными платформами; выполнять базовые операции администрирования баз данных, настраивать подключение скриптов к базе данных;</p> <p>ПК-4.2</p> <p>разрабатывать серверные веб-приложения на одном из языков программирования (например, Python, JavaScript (Node.js), PHP, Ruby и др.) с использованием фреймворков (например, Django, NestJS, Laravel и др.) и специальных библиотек; разрабатывать серверные веб-приложения на основе баз данных: реляционных (например, PostgreSQL, MySQL и др.) и нереляционных (например, MongoDB, Redis и др.); использовать выбранную среду программирования и средства системы управления базами данных; анализировать техническое задание и проектировать архитектуру серверной части приложения; реализовывать бизнес-логику приложения на серверном языке программирования; использо-</p>

	<p>вать API для обработки клиентских запросов; организовывать структуру проекта с разделением логики и представления (шаблонизация); применять принципы маршрутизации (роутинга) в серверных приложениях; реализовывать серверный код для работы с базой данных</p> <p>ПК-4.3</p> <p>выявлять ошибки в программном коде; применять методы и приёмы отладки программного кода; интерпретировать сообщения об ошибках, предупреждения, записи технологических журналов; применять современные компиляторы, отладчики и оптимизаторы программного кода; проверять корректность и эффективность SQL-запросов, использовать транзакции для обеспечения целостности данных при отладке комплексных операций; использовать инструменты мониторинга для выявления и анализа проблем в работе веб-приложения</p>
--	---

Учебная работа обучающихся организуется с целью:

1. систематизации и закрепления практического опыта, умений и знаний, общих и профессиональных компетенций, определенных в качестве основополагающих требованиями ФГОС СПО по дисциплинам, профессиональным курсам и междисциплинарным модулям;
2. формирования готовности к поиску, обработке и применению информации для решения профессиональных задач;
3. развития познавательных способностей и активности обучающихся, творческой инициативы, самостоятельности, ответственности и организованности;
4. формирования самостоятельности мышления, способностей к саморазвитию, самосовершенствованию и самореализации;
5. формирования умений использовать нормативную, правовую, справочную и специальную литературу;
6. выработка навыков эффективной профессиональной деятельности.

ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ О ФОРМАХ ПОДГОТОВКИ К ЭКЗАМЕНУ И КРИТЕРИЕВ ОЦЕНКИ

Структура профессионального модуля:

Коды профессиональных и общих компетенций	Наименования разделов профессионального модуля	Всего часов
ПК 4.1.; ПК 4.2.; ПК 4.3.;	МДК.04.01 Бэкенд-разработка (серверная часть)	144
ПК 4.1.; ПК 4.2.; ПК 4.3.;	МДК.04.02 Разработка серверных приложений с базами данных	144
ПК 4.1.; ПК 4.2.; ПК 4.3.;	УП.04.01 Учебная практика	36

ПК 4.1.; ПК 4.2.; ПК 4.3.;	ПМ.04.01(К) Экзамен по модулю	6
ПК 4.1.; ПК 4.2.; ПК 4.3.;	Всего:	330

Наименование междисциплинарных курсов (МДК)	Содержание учебного материала, практические занятия, самостоятельная работа обучающихся, курсовая работа	Объем часов
1	2	3
МДК.04.01 Бэкенд-разработка (серверная часть)	Темы лекций:	52
	Введение в бэкенд-разработку и PHP. Роль серверного языка. Принципы работы PHP-скрипта на веб-сервере. Синтаксис PHP: теги <?php ?>, инструкции.	2
	Базовый синтаксис PHP. Переменные, типы данных, операторы. Вывод данных (echo, print). Константы.	2
	Условные операторы и циклы в PHP. if-else, switch. Циклы for, while, foreach. Особенности foreach для массивов.	2
	Массивы в PHP. Индексированные и ассоциативные массивы. Многомерные массивы. Основные функции для работы с массивами (count, array_push, array_merge).	2
	Функции в PHP. Пользовательские функции: объявление, параметры, возврат значений. Глобальные и локальные области видимости. Встроенные функции.	2
	Работа с формами и методом GET. Суперглобальный массив \$_GET. Передача данных через URL. Обработка данных из формы с методом GET.	2
	Работа с формами и методом POST. Суперглобальный массив \$_POST. Отличия от GET. Обработка данных из формы с методом POST.	2
	HTTP-запросы и ответы. Заголовки. Основы протокола HTTP. Суперглобальный массив \$_SERVER. Отправка HTTP-заголовков с помощью header() (редирект).	2
	Работа с файлами на сервере. Чтение из файла (file_get_contents), запись в файл (file_put_contents). Работа с CSV-файлами как с простой БД.	2
	Сессии (Sessions). Суперглобальный массив \$_SESSION. Механизм работы сессий. Авторизация на основе сессий (базовый принцип).	2
	Куки (Cookies). Суперглобальный массив \$_COOKIE. Установка, чтение, удаление cookies. Отличия от сессий.	2
	Введение в функции для работы со строками. Поиск и замена подстрок (str_replace, strpos). Форматирование строк. Функции для безопасности (htmlspecialchars, trim).	2
	Функции для работы с датой и временем. time(), date(), strtotime(). Форматирование и операции с датами.	2
	Структура типичного PHP-приложения. Разделение логики и представления (принцип).	2
	Углублённое изучение HTTP-протокола. Методы запросов, коды ответов, заголовки. Работа с \$_SERVER и \$_REQUEST.	2

	Расширенная работа с файлами и директориями. Функции для работы с файловой системой: scandir(), is_file(), is_dir(), рекурсивный обход.	2
	Сессии и куки: расширенные возможности. Хранение сложных данных в сессиях, безопасность сессий, время жизни кук.	2
	Работа с JSON в PHP. Кодирование и декодирование JSON (json_encode(), json_decode()). Использование JSON для хранения данных.	2
	Шаблонизация и разделение логики. Принципы шаблонизации, создание простого шаблонизатора. Подключение шаблонов.	2
	Маршрутизация (роутинг) в PHP-приложениях. Понятие фронт-контроллера. Простая реализация роутера.	2
	Обработка ошибок и исключения. Конструкция try-catch. Пользовательские исключения. Логирование ошибок.	2
	Безопасность: защита от CSRF-атак. Генерация и проверка токенов. Реализация защиты формы.	2
	Работа с почтой в PHP. Отправка email через mail() и библиотеки. Формирование писем с вложением.	2
	Введение в REST API. Основы REST, создание простого API для обработки GET и POST-запросов.	2
	Оптимизация и отладка PHP-кода. Профилирование, использование var_dump() и debug_backtrace().	2
	Архитектура веб-приложений. Паттерн MVC (модель-представление-контроллер) на простом примере. Подготовка к экзамену.	2
	Лабораторные работы:	52
	Настройка рабочего окружения и базовый синтаксис. Установка и конфигурация серверного окружения. Создание первого исполняемого скрипта. Вывод информации о состоянии сервера и интерпретатора.	2
	Операции с базовыми типами данных. Выполнение арифметических и строковых операций. Практика преобразования типов данных и конкатенации.	2
	Реализация алгоритмов с использованием условных конструкций и циклов. Генерация HTML-структур на основе логических условий и итеративной обработки данных.	2
	Работа с индексированными и ассоциативными массивами. Создание, модификация и преобразование массивов. Форматированный вывод элементов массива в виде HTML-разметки.	2
	Структурирование приложения с помощью пользовательских функций. Создание библиотеки вспомогательных функций для выполнения типовых задач обработки данных.	2
	Обработка данных, передаваемых через GET-запрос. Чтение параметров из URL. Фильтрация и вывод данных на основе переданных пользователем критериев.	2
	Обработка данных, передаваемых через POST-запрос. Создание HTML-форм, валидация введённых пользователем данных на стороне сервера.	2
	Работа с файловыми загрузками от пользователя. Приём, валидация и сохранение файлов, загружаемых через HTTP-форму. Обработка метаданных загрузки.	2
	Взаимодействие с файловой системой на стороне сервера. Чтение из	2

	и запись в текстовые файлы для организации хранения данных приложения.	
	Реализация механизма аутентификации с использованием сессий. Организация защиты разделов веб-приложения, управление состоянием пользователя в рамках сессии.	2
	Работа с механизмом cookies. Использование cookies для долговременного хранения пользовательских предпочтений и реализации функционала "запомнить меня".	2
	Безопасная обработка пользовательского ввода. Практика применения функций для экранирования и очистки внешних данных перед их выводом в HTML или сохранением.	2
	Принципы разделения кода и организации простого приложения. Выделение общих элементов разметки в отдельные файлы. Построение логики для отображения списков и отдельных записей.	2
	Разработка простого контент-ориентированного приложения. Интеграция изученных технологий: создание интерфейса для добавления данных и их последующего отображения на клиентской стороне.	2
	Расширенная работа с HTTP. Создание скрипта, обрабатывающего различные HTTP-методы. Анализ заголовков запроса.	2
	Рекурсивный обход директорий. Создание скрипта для вывода дерева файлов и папок с учётом вложенности.	2
	Расширенная работа с сессиями. Реализация корзины покупок или многошаговой формы с сохранением данных в сессии.	2
	Работа с JSON. Создание системы хранения данных в JSON-файлах (например, гостевой книги или списка задач).	2
	Создание простого шаблонизатора. Разделение PHP-логики и HTML-шаблонов. Реализация наследования шаблонов.	2
	Реализация простого роутера. Создание фронт-контроллера, который направляет запросы на нужные скрипты.	2
	Обработка ошибок и исключений. Написание кода с использованием try-catch и создание пользовательских исключений для валидации данных.	2
	Защита от CSRF-атак. Реализация генерации и проверки токенов для всех форм в приложении.	2
	Отправка email. Создание формы обратной связи с отправкой письма на почту, включая обработку вложений.	2
	Создание простого REST API. Разработка API для работы с сущностью (например, статьи или товары) через GET и POST-запросы.	2
	Оптимизация и отладка. Поиск и устранение ошибок в предоставленном коде, улучшение его структуры.	2
	Итоговая работа: разработка модуля с использованием MVC. Создание простого приложения (например, блога или каталога) с разделением на модель, представление и контроллер.	2
	Самостоятельная работа обучающихся:	32
	Изучение дополнительных возможностей массивов и функций в PHP.	5
	Выполнение разделов курсовой работы.	
	Безопасная обработка пользовательского ввода и валидация данных.	5
	Выполнение разделов курсовой работы.	
	Углубленное изучение механизмов сессий и кук.	
	Выполнение разделов курсовой работы.	4
	Работа с альтернативными форматами данных (XML, YAML).	6
	Изучение паттернов проектирования веб-приложений.	6

	Инструменты отладки и профилирования PHP-приложений.	6
	Промежуточная аттестация (защита курсовой работы)	2
	Промежуточная аттестация: зачет и экзамен	8
МДК.04.02	Темы лекций:	52
Разработка серверных приложений с базами данных	Архитектура веб-приложения с БД. Роль СУБД MySQL. Взаимодействие PHP и MySQL. Установка и настройка Denwer (PHP, MySQL, phpMyAdmin). Основные объекты БД: базы, таблицы, столбцы.	2
	Проектирование таблиц для веб-приложения. Типы данных. Разработка структуры таблиц для сайта. Выбор типов данных (INT, VARCHAR, TEXT, DATETIME). Первичный ключ, AUTO_INCREMENT.	2
	Создание БД и таблиц через phpMyAdmin и SQL (DDL). Команды CREATE DATABASE, CREATE TABLE. Операторы изменения таблиц (ALTER TABLE). Удаление (DROP). Практика в интерфейсе phpMyAdmin.	2
	Основы SQL для веб-разработчика: добавление и выборка данных (DML). Команда INSERT для добавления записей. Команда SELECT для выборки. Фильтрация с WHERE. Сортировка ORDER BY. Ограничение выборки LIMIT.	2
	Реляционные связи между таблицами. Внешние ключи (FOREIGN KEY). Связь "один-ко-многим" (пользователь → статьи). Создание таблиц со связями. Понятие целостности данных. Каскадные операции.	2
	Объединение таблиц (JOIN) для получения связанных данных. Необходимость JOIN в веб-разработке. INNER JOIN для выборки статей с именами авторов. Практические примеры для сайта.	2
	Изменение и удаление данных в БД. Команды UPDATE (обновление профиля, статьи) и DELETE (удаление записи). Критическая важность условия WHERE. Мягкое удаление (флаг is_deleted).	2
	Подключение PHP к MySQL. Расширение PDO (PHP Data Objects). Создание объекта PDO для подключения. Обработка ошибок подключения. Выполнение простого SQL-запроса (query()).	2
	Выполнение SELECT-запросов из PHP и вывод данных на сайт. Методы fetch(), fetchAll(). Цикл для отображения списка товаров/статей. Обработка пустого результата.	2
	Защита от SQL-инъекций. Подготовленные запросы (Prepared Statements). Принцип SQL-инъекции. Использование prepare() и execute(). Привязка параметров. Пример с фильтрацией.	2
	Вставка данных из HTML-формы в БД (INSERT через PHP). Получение данных из \$_POST. Валидация. Формирование подготовленного INSERT-запроса. Редирект после успешного добавления.	2
	Реализация страницы детального просмотра (SELECT по ID). Получение параметра id из URL (\$_GET). Подготовленный запрос с WHERE id = ?. Вывод полной информации о записи.	2
	Обновление данных через веб-форму (UPDATE через PHP). Форма редактирования с предзаполненными данными	2

	(SELECT перед отображением). Обработка формы и выполнение UPDATE-запроса.	
	Базовое управление пользователями в БД. Хеширование паролей. Таблица users. Функции password_hash() и password_verify(). Регистрация и аутентификация (логин) - принципы.	2
	Нормализация базы данных. Проектирование сложной схемы. Аномалии данных. Нормальные формы (1NF, 2NF, 3NF) на примере интернет-магазина. Проектирование схемы "Заказ" с учетом нормализации.	2
	Транзакции в MySQL. Обеспечение целостности данных. Понятие ACID. Операторы START TRANSACTION, COMMIT, ROLLBACK. Пример использования при оформлении заказа (списание товара + создание заказа).	2
	Индексы в БД: ускорение выборок на сайте. Типы индексов. Создание индексов (CREATE INDEX). Анализ запросов (EXPLAIN). Когда индексы полезны, а когда мешают.	2
	Оптимизация SQL-запросов для веб-приложений. Анализ медленных запросов. Оптимизация JOIN, подзапросов. Кэширование запросов на уровне приложения.	2
	Расширенные возможности SELECT для веб-отчетов. Группировка (GROUP BY) для вывода количества товаров в категориях. Агрегатные функции (COUNT, SUM, AVG) для статистики в админке.	2
	Пагинация данных из БД. Принцип разбиения на страницы. Использование LIMIT и OFFSET. Расчет общего количества страниц. Параметры ?page= в URL.	2
	Реализация корзины покупок (сессия + БД). Хранение корзины в \$_SESSION. При оформлении заказа - перенос данных в таблицы orders и order_items. Сравнение с хранением корзины в БД.	2
	Построение дерева категорий (вложенная структура). Метод "parent_id" для организации дерева. Рекурсивный или итеративный вывод на сайте. Альтернатива - Closure Table (обзорно).	2
	Полнотекстовый поиск в MySQL. Создание полнотекстового индекса. Операторы MATCH() ... AGAINST(). Реализация поиска по сайту с релевантностью.	2
	Безопасность веб-приложений: SQL-инъекции, XSS, CSRF. Глубокая защита от SQL-инъекций (подготовленные запросы). Экранирование вывода (htmlspecialchars). Токены для защиты форм от CSRF.	2
	Резервное копирование и восстановление БД. Миграции. Утилита mysqldump. Создание дампа базы данных. Восстановление из дампа. Понятие миграций схемы БД.	2
	Обзор современных тенденций: ORM, NoSQL. Что такое ORM (Object-Relational Mapping), преимущества и недостатки. Краткое знакомство с Redis как кэш-хранилищем.	2
	Лабораторные работы:	52
	Установка окружения. Создание БД для блога через phpMyAdmin. Создание базы blog_db. Создание таблиц users (id, login, email) и posts (id, title, content, user_id, created_at).	2

	Наполнение БД тестовыми данными через SQL. Написание и выполнение 5-10 запросов INSERT для таблиц. Практика различных SELECT-запросов в phpMyAdmin.	2
	Создание структуры БД для сайта. Создание таблиц categories (id, name) и products (id, name, description, price, category_id). Установка внешнего ключа от products.category_id к categories.id.	2
	Написание SQL-запросов для каталога сайта. Выборка всех товаров с сортировкой по цене. Выборка товаров определенной категории. Выборка товаров по диапазону цен (BETWEEN).	2
	Реализация связи "многие-ко-многим". Создание таблиц orders и order_items для заказов. Написание запроса для вывода состава конкретного заказа с названиями товаров (через JOIN).	2
	Подключение PHP-скрипта к БД и вывод списка статей. Создание файла config.php с данными подключения. Скрипт index.php, который выбирает все статьи из posts и выводит их в виде HTML-списка.	2
	Разработка страницы "Статья" с использованием подготовленного запроса. Скрипт article.php?id=X. Защита от инъекций. Вывод заголовка, содержимого и автора статьи (пока ID автора).	2
	Создание формы добавления новой категории товара. HTML-форма с полем "Название категории". Обработчик на PHP: валидация, подготовленный INSERT. Редирект на страницу списка категорий.	2
	Реализация формы обратной связи с сохранением в БД. Создание таблицы feedback (id, name, email, message, date). Форма на сайте. Скрипт обработки формы с записью данных в БД.	2
	Создание страницы редактирования статьи. Скрипт edit_article.php?id=X. SELECT для предзаполнения формы (textarea). Обработка POST-запроса с UPDATE. Редирект на страницу статьи.	2
	Реализация простого поиска по статьям. Форма поиска с полем "запрос". Скрипт поиска: подготовленный запрос с WHERE title LIKE ? или WHERE content LIKE ?. Вывод результатов.	2
	Создание регистрации пользователя с хешированием пароля. Форма регистрации. Проверка уникальности логина. Хеширование пароля (password_hash). INSERT в таблицу users.	2
	Реализация входа на сайт (аутентификация). Форма входа. SELECT пользователя по логину. Проверка пароля (password_verify). Запуск сессии и сохранение user_id.	2
	Итоговая практика: CRUD для товаров (без интерфейса). Написание набора скриптов (в виде функций или отдельных файлов), выполняющих Create, Read, Update, Delete для таблицы products. Тестирование через форму и вывод.	2
	Проектирование и создание БД для форума. Разработка схемы: пользователи, разделы, темы, сообщения. Создание всех таблиц со связями через phpMyAdmin и SQL-скриптом.	2
	Реализация транзакции: перевод средств между счетами. Создание таблицы accounts (id, user_id, balance). Скрипт, который внутри транзакции делает два UPDATE (списание + зачисление).	2

	Оптимизация каталога товаров с помощью индексов. Добавление индексов на поля category_id, price, name. Сравнение скорости выполнения запросов до и после с помощью EXPLAIN.	2
	Создание страницы "Статистика сайта" для админа. Запросы для подсчета: общее количество пользователей, товаров, заказов. Средний чек. Количество заказов за последнюю неделю.	2
	Реализация пагинации для каталога товаров. Модификация страницы каталога. Расчет общего числа товаров, номера страницы из GET-параметра. Вывод ссылок для постраничной навигации.	2
	Разработка модуля "Корзина покупок" (сессия + оформление). Хранение корзины в сессии. Страница корзины с возможностью изменения количества. Форма оформления заказа с переносом данных в БД.	2
	Создание системы комментариев к статьям. Таблица comments (id, article_id, user_id, text, date). Форма добавления комментария. Вывод списка комментариев к статье.	2
	Реализация рейтинга или лайков для товаров/статей. Таблица ratings (user_id, item_id, value). Ограничение UNIQUE для пары (user_id, item_id). Подсчет среднего рейтинга.	2
	Настройка полнотекстового поиска по статьям. Добавление полнотекстового индекса на поля title и content. Создание страницы поиска с использованием MATCH() AGAINST().	2
	Защита административного раздела. Проверка роли пользователя в сессии. Создание простой админ-панели с CRUD для товаров. Добавление CSRF-токенов в формы админки.	2
	Создание резервной копии БД и развертывание на "чистом" сервере. Экспорт БД проекта через mysqldump (или phpMyAdmin). Создание скрипта для автоматического восстановления структуры и данных.	2
	Итоговый проект: "Блог с расширенным функционалом". Реализация: много пользователей с ролями, статьи с категориями, комментарии с деревом ответов, теги к статьям, личный кабинет, админ-панель. Сдача проекта с полной документацией по БД.	2
	Самостоятельная работа обучающихся:	32
	Принципы проектирования интерфейсов администратора (Admin Panel) для работы с данными.	5
	Автоматизация рутинных операций: SQL-скрипты и консольные утилиты для администрирования БД.	5
	Альтернативные методы аутентификации и основы безопасности сессий.	4
	Введение в систему контроля версий Git для совместной разработки проекта с БД.	6
	Выполнение разделов курсовой работы.	
	Проектирование и реализация простого REST API для мобильного клиента.	6
	Выполнение разделов курсовой работы.	
	Стратегии кэширования данных для повышения производительности веб-приложения.	6
	Выполнение разделов курсовой работы.	
	Промежуточная аттестация (защита курсовой работы)	2

Промежуточная аттестация: зачет и экзамен	8
Учебная практика: Выполнение видов работ, предусмотренных рабочей программы практики	36
Экзамен по модулю:	6
Всего:	330

Тематический план и содержание профессионального модуля

Общие методические рекомендации студенту при изучении темы дисциплины.

Большая часть самостоятельной работы выполняется студентом вне учебных занятий при подготовке домашних заданий. Общие требования к выполнению этого вида самостоятельной работы заключаются в следующем:

- активно работать на уроке, усваиваю основную часть нового материала;
- если что-то непонятно, не стесняться задавать вопросы преподавателю;
- большое задание необходимо разбивать на части и работать над каждой из них в отдельности;
- выполняя домашнее задание, надо не просто думать, что надо сделать, а еще и решать, с помощью каких средств и приемов этого можно добиться;
- в процессе приготовления домашнего задания необходимо делать перерывы;
- готовиться к докладам, рефератам, защите курсовых работ и проектов, практических и лабораторных занятий надо заранее, равномерно распределяя нагрузку, а не оставлять такую ответственную работу на последний день;
- изучая заданный материал, сначала надо его понять, а уже потом запомнить;
- научиться находить интересующую нужную информацию с помощью компьютера;
- не стесняться обращаться за помощью к взрослым и однокурсникам;
- надо составлять план устного ответа и проверять себя;
- на письменном столе должно лежать только то, что необходимо для выполнения одного задания. После его завершения со стола убираются уже использованные материалы, и кладутся те учебные принадлежности, которые необходимы для выполнения следующего задания;
- нужно решить, в какой последовательности лучше выполнять задания и сколько времени понадобится на каждое из них;
- трудный материал урока лучше повторить в тот же день, чтобы сразу закрепить его и запомнить;
- читая учебник, надо задавать самому себе вопросы по тексту.

Подготовка тематических сообщений, докладов, рефератов

Реферат доклад, сообщение (от латинского *refero* - передаю, сообщаю) - краткое письменное изложение материала по определенной теме с целью привития студентам навыков самостоятельного поиска и анализа информации, формирования умения подбора и изучения литературных источников, используя при этом дополнительную научную, методическую и периодическую литературу.

Тема реферата выбирается по желанию студента из списка, предлагаемого преподавателем. Тема может быть сформулирована студентом самостоятельно.

Выбранная тема согласовывается с преподавателем.

После выбора темы требуется:

- составить план реферата;
- подобрать необходимую информацию;
- изучить подобранные информацию;
- составить текст реферата.

План реферата должен включать в себя введение, основной текст и заключение. Во введении аргументируется актуальность выбранной темы, указываются цели и задачи исследования. В нем также отражается методика исследования и структура работы. Основная часть работы предполагает освещение материала в соответствии с планом. В заключении излагаются основные выводы и рекомендации по теме исследования.

Реферат оформляется согласно требованиям, установленным в учебном заведении. Он должен содержать: титульный лист, оглавление и список использованной литературы. На титульном листе указываются: название учебного заведения, название профессионального модуля, междисциплинарного курса, тема работы, курс, группа, фамилии, имена, отчества студента и руководителя работы, название города, в котором находится учебное заведение, год написания данной работы. Реферат может содержать приложения в форме схем, образцов документов и другие изображения в соответствии с темой исследования. Все страницы работы, включая оглавление и список литературы, нумеруются по порядку с титульного листа (на нем цифра не ставится) до последней страницы без пропусков и повторений. Введение, заключение, новые главы, список использованных источников и литературы должны начинаться с нового листа. Подбор литературы производится студентом из предложенного преподавателем списка литературы. Текст реферата необходимо набирать на компьютере на одной стороне листа. Размер левого поля 30 мм, правого - 15 мм, верхнего - 20 мм, нижнего - 20 мм. Шрифт - Times New Roman, размер - 14, межстрочный интервал - 1,5. Фразы, начинающиеся с новой строки, печатаются с абзацным отступом от начала строки (1,25 см). Реферат, выполненный небрежно, неразборчиво, без соблюдения требований по оформлению, возвращается студенту без проверки с указанием причин возврата на титульном листе.

Критерии оценки:

- знание и понимание проблемы;
- умение систематизировать и анализировать материал, четко и обоснованно формулировать выводы;
- «трудозатратность» (объем изученной литературы, добросовестное отношение к анализу проблемы);
- самостоятельность, способность к определению собственной позиции по проблеме и к практической адаптации материала, недопустимость плагиата;

- выполнение необходимых формальностей (точность в цитировании и указании источника текстового фрагмента, аккуратность оформления).

Проработка занятый, учебных изданий и специальной технической литературы

Работа с конспектом лекций по темам междисциплинарных курсов заключается в том, что студент после рассмотрения темы на учебных занятиях в период между очередными лекциями изучает материал конспекта. При этом непонятные положения конспекта необходимо выяснить у преподавателя на консультациях или при чтении основной и дополнительной литературы.

При работе с книгой необходимо научиться правильно ее читать, вести записи. Для подбора литературы в библиотеке используются алфавитный и систематический каталоги. Правильный подбор учебников рекомендуется преподавателем. Необходимая литература может быть также указана в методических разработках. Изучая материал по учебнику, следует переходить к следующему вопросу только после правильного уяснения предыдущего, описывая на бумаге все выкладки и определения (в том числе те, которые в учебнике опущены или на лекции даны для самостоятельного вывода). Полезно составлять опорные конспекты. При изучении материала по учебнику, полезно в тетради (на специально отведенных полях) дополнять конспект лекций, написанный на учебных занятиях. Там же следует отмечать вопросы, выделенные студентом для консультации с преподавателем. Выводы, полученные в результате изучения, рекомендуется в конспекте выделять, чтобы они при пропитывании записей лучше запоминались. Различают два вида чтения; первичное и вторичное. Первичное - это внимательное, неторопливое чтение, при котором можно остановиться на трудных местах. После него не должно остаться ни одного непонятного слова. Содержание не всегда может быть понятно после первичного чтения. Задача вторичного чтения - полное усвоение смысла целого (по счету это чтение может быть и не вторым, а третьим или четвертым).

Чтение научного текста является частью познавательной деятельности. Ее цель - извлечение из текста необходимой информации. От того на сколько осознанна читающим собственная внутренняя установка при обращении к печатному слову (найти нужные сведения, усвоить информацию полностью или частично, критически проанализировать материал и т.п.) во многом зависит эффективность осуществляемого действия. Выделяют четыре основные установки в чтении научного текста:

- информационно-поисковая, задача которой - найти, выделить искомую информацию;
- усваивающая, при которой усилия читателя направлены на то, чтобы как можно полнее осознать и запомнить как сами сведения, излагаемые автором, так и всю логику его рассуждений;
- аналитико-критическая - читатель стремится критически осмыслить материал, проанализировав его, определив свое отношение к нему;

- творческая, создающая у читателя готовность в том или ином виде использовать суждения автора, ход его мыслей, результат наблюдения, разработанную методику, дополнить их, подвергнуть новой проверке.

Самостоятельная работа при чтении учебной литературы начинается с изучения конспекта материала, полученного при слушании лекций преподавателя. Полученную информацию необходимо осмыслить. При необходимости, в конспект лекций могут быть внесены схемы, эскизы рисунков, другая дополнительная информация.

Составление конспекта, тематических схем, таблиц

При изучении нового материала, как правило, составляется конспект. Конспект - изложение текста, которому присущи краткость, связность и последовательность. При этом максимально точно записываются формулы, определения, схемы, трудные для запоминания места.

При оформлении конспекта необходимо стремиться к емкости каждого предложения. Мысли автора книги следует излагать кратко, заботясь о стиле и выразительности написанного. Число дополнительных элементов конспекта должно быть логически обоснованным, записи должны распределяться в определенной последовательности, отвечающей логической структуре текста. Для уточнения и дополнения необходимо оставлять поля. Овладение навыками конспектирования требует от студента целеустремленности, повседневной самостоятельной работы.

Классификация конспектов:

- плановый конспект, для чего сначала нужно написать план текста, а затем на пункты плана делаются комментарии: свободно изложенный текст либо цитаты;
- обзорный конспект - краткое изложение данной темы с использованием нескольких источников;
- текстуальный конспект состоит из цитат одного текста;
- свободный конспект предполагает цитаты текста и собственные формулировки прочитанного текста;
- сложный - конспект, в котором отражается определенная тема или вопрос;
- хронологический конспект отражает последовательность событий;
- опорный конспект, в котором излагается информация в виде опорных знаков, слов, сигналов.

Методические рекомендации по составлению конспекта:

- определить цель написания конспекта;
- внимательно прочитать текст, уточнить в справочной литературе непонятные слова;
- выделить основные смысловые части текста;
- определить главное, составить план;
- кратко сформулировать основные положения текста, отметить аргументацию автора;

- составить текст конспекта, изложив информацию кратко и своими словами, четко следуя пунктам плана, записи следует вести четко, ясно;
- грамотно записывать цитаты, учитывая лаконичность, значимость мысли;
- в тексте конспекта желательно приводить не только тезисные положения, но и их доказательства.

При составлении тематических схем, таблиц необходимо внимательно прочитать текст соответствующий параграф учебника. Продумать «конструкцию» таблицы или схемы, расположение порядковых номеров, терминов, примеров и пояснений (и прочего). Начертить схему или таблицу и заполнить ее графы необходимым содержимым.

Подготовка к лабораторным работам и практическим занятиям, оформление отчетов по лабораторным работам и практическим занятиям, подготовка к их защите

Программы профессиональных модулей предусматривают выполнение практических и лабораторных занятий.

Лабораторное занятие - форма учебного занятия, ведущей дидактической целью которого является экспериментальное подтверждение и проверка существующих теоретических положений (законов, зависимостей), формирование учебных и профессиональных практических умений и навыков.

Практическое занятие - это одна из форм учебной работы, которая ориентирована на закрепление изученного теоретического материала, его более глубокое усвоение и формирование умения применять теоретические знания в практических целях. Особое внимание на практических занятиях уделяется выработке учебных или профессиональных навыков. Такие навыки формируются в процессе выполнения конкретных заданий - упражнений, задач - под руководством и контролем преподавателя.

Подготовка к практическим и лабораторным занятиям заключается в работе с конспектом лекций по данной теме, в изучении соответствующего раздела учебника или учебного пособия, в просмотре дополнительной литературы. Этапы подготовки к практическому или лабораторному занятию заключаются в следующем: освежить в памяти теоретические сведения, полученные на лекциях и в процессе самостоятельной работы, подобрать необходимую учебную и справочную литературу. Отобрать те материалы, которые позволят в полной мере реализовать цели и задачи предстоящей работы. Еще раз проверить соответствие отобранного материала. Студент должен прийти на лабораторное или практическое занятие подготовленным по данной теме.

При выполнении заданий практического или лабораторного занятия студент должен быть ознакомлен преподавателем с целью и ходом выполнения задания и, по необходимости, с правилами техники безопасности. Если у студентов во время выполнения заданий возникают вопросы, то преподаватель консультирует студентов. Порядок выполнения того или иного задания излагается в инструкционных картах или рабочих тетрадях.

После проведения занятия студент представляет письменный отчет, который оформляется в соответствие с принятыми в образовательном учреждении правилами. Отчеты оформляются на листах писчей бумаги формата А4 или в специальных рабочих тетрадях, разработанных преподавателем. Содержание отчета указано в инструкционных картах или рабочих тетрадях.

При подготовке к защите практических и лабораторных занятий студент должен ответить на контрольные вопросы, указанные также в инструкционных картах или рабочих тетрадях, проштудировав при этом конспект лекций, учебную литературу.

Моделирование и решение производственных процессов и ситуационных задач

При изучении дисциплины очень часто студенту приходится сталкиваться с профессиональными задачами и ситуациями, которые необходимо решить самостоятельно, как во время аудиторной работы, так и во время внеаудиторной. При решении таких задач необходимо:

- провести анализ ситуации для определения проблемы в целом; представить ситуацию и себя в качестве действующего в ней лица; проанализировать ошибочные или правильные действия всех участников ситуации;
- определить проблемные узлы - возможные причины и прогнозируемые последствия развития данной ситуации;
- рассмотреть условное прогнозирование развития ситуации: определить окончательную гипотезу, представить обоснованный и доказательный прогноз вероятностного развития ситуации; предложить варианты действий, обоснованные теоретически и, по возможности, подкрепленные практическим личным опытом, опираясь на принципы профессиональной этики; определить способы и методы воздействия на предлагаемую ситуацию;
- сформулировать итоговые выводы, используя профессиональные термины, доказательства правильности своего решения.

Подготовка презентаций

Подготовка презентации позволит студенту логически выстроить изучаемый материал, систематизировать его, сформировать коммуникативные компетенции. Материал презентации представляется в виде текста, схем, диаграмм, таблиц, которые призваны дополнить текстовую информацию или передать ее в более наглядном виде. Желательно избегать в презентации изображений, не несущих смысловой нагрузки, если они не являются частью стилевого оформления. Цвет графических изображений не должен резко контрастировать с общим стилевым оформлением слайдов, иллюстрации рекомендуется сопровождать пояснительным текстом.

Анимационные эффекты используются для привлечения внимания слушателей или для демонстрации динамики развития какого - либо процесса. В этих случаях использование анимации оправдано, но не стоит чрезмерно

насыщать презентацию такими эффектами, иначе это вызовет негативную реакцию аудитории.

Звуковое сопровождение должно отражать суть или подчеркивать особенность темы слайда, презентации. Фоновая музыка не должна отвлекать внимание слушателей и заглушать слова докладчика.

Оптимальное количество слайдов, как правило, десять - пятнадцать. Для оформления слайдов презентации рекомендуется использовать несложные шаблоны, соблюдать единый стиль. Не рекомендуется на одном слайде использовать более трех цветов. Смену слайдов для управления презентацией докладчиком желательно устанавливать по щелчку без времени. Шрифт, выбираемый для презентации, должен обеспечивать читаемость информации на экране и соответствовать выбранному шаблону оформления. Не желательно использовать разные шрифты в одной презентации.

Алгоритм выстраивания презентации должен соответствовать логической структуре работы и отражать последовательность ее этапов. Независимо от алгоритма выстраивания презентации на первом слайде рекомендуется выносить следующие данные: полное наименование образовательной организации; тема презентации; фамилия, имя, отчество студента; специальность обучения; фамилия, имя, отчество руководителя. Последний слайд должен содержать фразу «Спасибо за внимание».

Работа с электронными ресурсами в сети Интернет

Для повышения эффективности самостоятельной работы студент должен учиться работать в поисковой системе сети Интернет, в электронно-библиотечной системе и использовать найденную информацию при подготовке к занятиям.

Интернет сегодня - правомерный источник научных статей, статистической и аналитической информации, и использование его наряду с книгами давно уже стало нормой. Однако, несмотря на то, что ресурсы Интернета позволяют достаточно быстро и эффективно осуществлять поиск необходимой информации, следует помнить о том, что эта информация может быть неточной или вовсе не соответствовать действительности. В связи с этим при поиске материала по заданной тематике следует обращать внимание на научные труды признанных авторов, которые посоветовали вам преподаватели.

Поиск информации можно вести по автору, заглавию, виду издания, году издания или издательству. Также в сети Интернет доступна услуга по скачиванию методических указаний и учебных пособий, подбору необходимой учебной и научно - технической литературы.

Подготовка к семинару

Семинар — это особая форма учебно-теоретических занятий, которая, как правило, служит дополнением к лекционному курсу. Семинар обычно посвящен детальному изучению отдельной темы.

Этапы подготовки к семинару:

- проанализировать тему семинара, подумать о цели и основных проблемах, вынесенных на обсуждение;
- внимательно прочитать материал, данный преподавателем по этой теме на лекции;
- изучить рекомендованную литературу, делая при этом конспекты прочитанного или выписки, которые понадобятся при обсуждении на семинаре;
- постараться сформулировать свое мнение по каждому вопросу и аргументированно его обосновать;
- записать возникшие во время самостоятельной работы с учебниками и научной литературой вопросы, чтобы затем на семинаре получить на них ответы.

При подготовке к семинарским занятиям следует руководствоваться указаниями и рекомендациями преподавателя, использовать основную и дополнительную литературу из представленного им списка.

При подготовке доклада на семинарское занятие желательно заранее обсудить с преподавателем перечень используемой литературы, за день до семинарского занятия предупредить его о необходимых для представления материала технических средствах. Напечатанный текст доклада представить преподавателю на рецензию.

Подготовка к зачетам, экзаменам

Изучение выше перечисленных тем дисциплины завершается зачетами или экзаменами.

Подготовка к зачету или экзамену способствует закреплению, углублению и обобщению знаний, получаемых в процессе обучения, а также применению их к решению практических задач. Готовясь к зачету или экзамену, студент ликвидирует имеющиеся пробелы в знаниях, углубляет, систематизирует и упорядочивает свои знания. На зачете или экзамене студент демонстрирует то, что он приобрел в процессе обучения конкретным темам междисциплинарных курсов или модулям в целом.

Экзаменационная сессия - это серия экзаменов, установленных учебным планом. Между экзаменами, согласно графику их проведения, дается интервал времени в несколько дней. Не следует думать, что их достаточно для успешной подготовки к экзаменам. В эти дни нужно систематизировать уже имеющиеся знания. На консультации перед экзаменом студентов познакомят с основными требованиями, ответят на возникшие у них вопросы. Поэтому посещение консультаций обязательно.

Требования к организации подготовки студента к экзаменам те же, что и при занятиях в течение семестра, но соблюдаются они более строго. Во-первых, очень важно соблюдение режима дня: сон не менее 8 часов в сутки, занятия должны заканчиваться не позднее, чем за 2-3 часа до сна.

Оптимальное время занятий - утренние и дневные часы. В перерывах между занятиями рекомендуются прогулки на свежем воздухе, неутомительные занятия спортом. Во-вторых, наличие хороших собственных конспектов лекций. Даже в том случае, если была пропущена какая-либо лекция, необходимо во

время ее восстановить, обдумать, снять возникшие вопросы для того, чтобы запоминание материала было осознанным. В-третьих, при подготовке к зачету или экзамену у студента должен быть хороший учебник или конспект литературы, прочитанной по указанию преподавателя в течение семестра. Здесь можно эффективно использовать листы опорных конспектов. Вначале следует просмотреть весь материал по сдаваемой теме, отметить для себя трудные вопросы, обязательно в них разобраться. В заключение еще раз целесообразно повторить основные положения. Систематическая подготовка к занятиям в течение семестра позволит использовать время экзаменационной сессии для систематизации знаний.

Правила подготовки к экзамену:

- сориентироваться во всем материале и обязательно расположить его согласно экзаменационным вопросам или вопросам, обсуждаемым на семинарах, учебных занятиях. Эта работа может занять много времени, но все остальное - уже технические детали, главное - это ориентировка в материале;

- постараться максимально запомнить материал, переосмыслить его, рассмотреть альтернативные идеи;

- подготовить «шпаргалки», главный смысл которых систематизация и оптимизация знаний, однако пользоваться таким подспорьем не рекомендуется. Это очень сложная и важная для студента работа, более сложная и важная, чем простое поглощение массы учебной информации. Если студент самостоятельно подготовил такие «шпаргалки», то, скорее всего, он и экзамены сдавать будет более уверенно, так как у него уже сформирована общая ориентировка в сложном материале. Как это ни парадоксально, но использование «шпаргалок» часто позволяет отвечающему студенту лучше демонстрировать свои познания, точнее - ориентировку в знаниях, что намного важнее знания «запомненного» и «тут же забытого» после сдачи экзамена.

При ответе на экзамене студент сначала должен продемонстрировать преподавателю усвоенный по программе обучения материал, и лишь после этого высказать иную, желательно аргументированную точку зрения.

Критерии оценки конспекта (сообщения)

№	Критерии оценивания	«Отлично»	«Хорошо»	«Удовлетворительно»	«Не удовлетворительно»
1	Объем выполненной работы	Оптimalен для конспектирования материала	Оптimalен для конспектирования материала	Занижен завышен	Занижен завышен

2	Логическая последовательность и связанность материала	+	Незначительно нарушена	Нет, нарушена	нет
3	Полнота изложения содержания	+	Не выдержана	Не выдержана	Не выдержана
4	Сохранение основной идеи через весь конспект	+	+	нарушено	нет
5	Использование дополнительной литературы(при постановке подобной задачи)	+	+	Не достаточно	Не используется
6	Оформление	+	+	Наличие отклонений	Наличие отклонений
7	Соблюдение языковой и речевой норм (как дополнительный критерий)	+	+	Соблюдаются слабо	нарушены

Критерии оценки практической/лабораторной работы:

№	Критерии оценивания	«Отлично»	«Хорошо»	«Удовлетворительно»	«Не удовлетворительно»
1	Правильность выполнения расчетов или иллюстраций	Выполнено с высокой точностью	Имеются незначительные помарки, ошибки	Неточность в изложении материала	Грубые ошибки в расчетах, отсутствуют выводы
	Логичная последовательность и связанность материала	+	Незначительно нарушена	нарушена	Отсутствует
	Оформление	+	+	Наличие отклонений	Наличие отклонений
	Соблюдение языковой и речевой	+	—	Соблюдаются слабо	Нарушены, отсутствует

	норм, профессио- нальная лексика и терминология				
--	---	--	--	--	--

КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

Код и наименование профессиональных и общих компетенций, формируемых в рамках модуля	Основные показатели оценки результата	Формы и методы контроля и оценки
ПК 4.1 Администрировать среды и платформы разработки информационных ресурсов	Знать: ПК-4.1 сетевые протоколы и основы веб-технологий; современные серверные языки программирования, фреймворки и библиотеки для разработки веб-приложений, принципы организации сред разработки; технологии сред развертывания и функционирования веб-приложений (контейнеризация, оркестрация, виртуализация и облака); современные стандарты взаимодействия компонентов распределенных приложений; технологии защиты и безопасности веб-ресурсов на стороне сервера; базовые операции администрирования баз данных	Оценка результатов: - устный опрос; - контроль выполнения индивидуальных заданий; - выполнение лабораторных работ; - выполнение тестовых заданий по темам МДК; - результаты выполнения работ по учебной и производственной практикам; - экзамен и зачет по МДК.04.01;
ПК 4.2 Создавать программный код на стороне сервера в соответствии с техническим заданием (спецификацией) с использованием языков программирования, библиотек и фреймворков	Знать: ПК-4.2	

<p>ПК 4.3 Осуществлять отладку программного кода на стороне сервера на уровне программных модулей, межмодульных взаимодействий и взаимодействий с окружением</p>	<p>архитектурные модели серверных приложений; синтаксис и основные возможности серверных языков программирования (например, Python, JavaScript (Node.js), PHP, Ruby и др.) с использованием фреймворков (например, Django, NestJS, Laravel и др.) и специальных библиотек; основы баз данных: реляционных (например, PostgreSQL, MySQL и др.) и нереляционных (например, MongoDB, Redis и др.); принципы работы с базами данных на стороне сервера: подключение, выполнение запросов, ORM-технология; методы обработки HTTP-запросов и формирования HTTP-ответов; архитектурный шаблон взаимодействия «клиент (браузер) – сервер – база данных»; язык структурированных запросов (SQL) для операций манипуляции данными; основы аутентификации и авторизации в веб-приложениях; основы архитектуры RESTful API, принципы шаблонизации и разделения кода (MVC); принципы построения защищенных веб-приложений: защита от SQL-инъекций, валидация и экранирование пользовательского ввода; основы реализации бизнес-логики типовых веб-модулей: аутентификации (хеширование паролей), CRUD-операций, пагинации, простых отчетов;</p> <p>Знать:</p> <p>ПК-4.3</p> <p>классификация ошибок серверного кода: синтаксические, логические, ошибки времени выполнения; методы и инструменты отладки для различных серверных языков; методы диагностики проблем взаимодействия модулей: трассировка вызовов, профилирование; особенности отладки взаимодействия с окружением: переменные среды, конфигурационные файлы, права доступа; подходы к отладке асинхронного и параллельного кода; методы анализа и оптимизации производительности на уровне кода и запросов; типы и форматы сообщений об ошибках, предупреждений; классификацию ошибок в веб-разработке: синтаксические ошибки, ошибки выполнения (исключения), логические ошибки в SQL, ошибки подключения к БД; методы и инструменты диагностики: встроенные сообщения об ошибках в языках программирования</p>	<ul style="list-style-type: none"> - курсовая работа по МДК.04.01; - экзамен и зачет по МДК.04.02; - курсовая работа по МДК.04.02; - экзамен по модулю.
<p>Экспертная оценка по результатам деятельности обучающихся в процессе освоения образовательной программы:</p> <ul style="list-style-type: none"> - на лабораторных занятиях; - при выполнении работ на различных этапах учебной и производственной практик. 		

<p>рования и СУБД, вывод отладочной информации, анализ журналов ошибок; методы диагностики проблем взаимодействия: проверка подключения к БД, анализ передаваемых параметров (ID, данные формы), проверка состояния сессий; способы использования технологических журналов, форматы и типы записей журналов; современные компиляторы, отладчики и оптимизаторы программного кода</p> <p>Уметь:</p> <p>ПК-4.1</p> <p>использовать технологии среды развертывания и функционирования веб-приложений; подбирать и настраивать конфигурацию веб-сервера; администрировать веб-сервера, настраивать и мониторить серверную инфраструктуру; использовать методы защиты на стороне сервера, реализовывать механизмы аутентификации и авторизации; использовать современные Фреймворки и библиотеки для разработки серверных скриптов; применять скрипты для реализации серверной логики; применять принципы работы с облачными платформами; выполнять базовые операции администрирования баз данных, настраивать подключение скриптов к базе данных;</p> <p>Уметь:</p> <p>ПК-4.2</p> <p>разрабатывать серверные веб-приложения на одном из языков программирования (например, Python, JavaScript (Node.js), PHP, Ruby и др.) с использованием фреймворков (например, Django, NestJS, Laravel и др.) и специальных библиотек; разрабатывать серверные веб-приложения на основе баз данных: реляционных (например, PostgreSQL, MySQL и др.) и нереляционных (например, MongoDB, Redis и др.); использовать выбранную среду программирования и средства системы управления базами данных; анализировать техническое задание и проектировать архитектуру серверной части приложения; реализовывать бизнес-логику приложения на серверном языке программирования; использовать API для обработки клиентских запросов; организовывать структуру проекта с разделением логики и представления (шаблонами, стилями и т.д.); использовать технологии миграции баз данных.</p>	
---	--

	<p>лонизация); применять принципы маршрутизации (роутинга) в серверных приложениях; реализовывать серверный код для работы с базой данных</p> <p>Уметь:</p> <p>ПК-4.3</p> <p>выявлять ошибки в программном коде; применять методы и приёмы отладки программного кода; интерпретировать сообщения об ошибках, предупреждения, записи технологических журналов; применять современные компиляторы, отладчики и оптимизаторы программного кода; проверять корректность и эффективность SQL-запросов, использовать транзакции для обеспечения целостности данных при отладке комплексных операций; использовать инструменты мониторинга для выявления и анализа проблем в работе веб-приложения</p>	
--	---	--

ЗАКЛЮЧЕНИЕ

На современном рынке труда конкурентоспособным может стать только квалифицированный работник соответствующего уровня и профиля, компетентный, свободно владеющей своей профессией и ориентированный в смежных областях деятельности, способный к эффективной работе по специальности на уровне мировых стандартов и готовый к постоянному профессиональному росту. В современных реалиях задача преподавателя заключается в организации и направлении познавательной деятельности обучающихся, эффективность которой во многом зависит от их самостоятельной работы. В свою очередь, самостоятельная внеаудиторная учебная работа обучающихся выступает средством достижения прочных и глубоких знаний, инструментом формирования активности и самостоятельности обучающихся.

Основной целью учебно-методического пособия является оказание методической помощи по подготовки к зачету/экзамену обучающихся для расширения, углубления и закрепления знаний и умений обучающихся, а также формирования общих (ОК) и профессиональных (ПК) компетенций.

Учебно-методическое пособие рассчитано на базовую подготовку обучающихся среднего профессионального образования и содержит основные требования и рекомендации по подготовке к сдаче зачета/экзамена.

ИНФОРМАЦИОННЫЕ ИСТОЧНИКИ

Основная учебная литература:

1. Полуэктова, Н. Р. Разработка веб-приложений : учебник для среднего профессионального образования / Н. Р. Полуэктова. – 2-е изд. – Москва : Издательство Юрайт, 2025. – 204 с. – (Профессиональное образование). – ISBN 978-5-534-18644-4. – Текст : электронный
2. Тузовский, А. Ф. Проектирование и разработка web-приложений : учебник для среднего профессионального образования / А. Ф. Тузовский. – Москва : Издательство Юрайт, 2025. – 219 с. – (Профессиональное образование). – ISBN 978-5-534-16767-2. – Текст : электронный
3. Сысолетин, Е. Г. Разработка интернет-приложений : учебник для среднего профессионального образования / Е. Г. Сысолетин, С. Д. Ростунцев. – Москва : Издательство Юрайт, 2025. – 80 с. – (Профессиональное образование). – ISBN 978-5-534-19603-0. – Текст : электронный
4. Чертыховцев, В. К. Проектирование интерфейсов пользователя. Человеко-машинное взаимодействие : учебник для среднего профессионального образования / В. К. Чертыховцев. – Москва : Издательство Юрайт, 2024. – 111 с. – (Профессиональное образование). – ISBN 978-5-534-20809-2. – Текст : электронный
5. Ломаш, Д.А. Интернет-технологии и мультимедиа: учеб. пособие / Д. А. Ломаш, О. Г. Ведерникова; ФГБОУ ВО РГУПС. - Ростов н/Д: [б. и.], 2017. - 118 с.: ил. - Библиогр. : 11 назв.- Текст : электронный
6. Панасов, В.Л. Разработка интерфейсов для интерактивных web-приложений: учеб. пособие / В. Л. Панасов; ФГБОУ ВО РГУПС. - Ростов н/Д: [б. и.], 2016. - 63 с.: ил. - Библиогр.: 6 назв.- Текст : электронный
7. Панасов, В.Л. Автоматизация разработки программного обеспечения для web: учеб. пособие / В. Л. Панасов, А. М. Лященко; ФГБОУ ВО РГУПС. - Ростов н/Д: [б. и.], 2019. - 90 с.: ил., табл. - Библиогр. : 22 назв..- Текст : электронный