

МИНОБРНАУКИ РОССИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «ВГУ»)

УТВЕРЖДАЮ

Заведующий кафедрой
Математического обеспечения ЭВМ



Абрамов Г. В.
22.03.2024 г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

ФТД.02 Фреймворки для web-приложений

Код и наименование дисциплины в соответствии с Учебным планом

1. Код и наименование направления подготовки / специальности:

01.04.02 Прикладная математика и информатика

2. Профиль подготовки / специализации / магистерская программа:

Математическое и программное обеспечение информационных систем

3. Квалификация (степень) выпускника: магистр

4. Форма обучения: очная

5. Кафедра, отвечающая за реализацию дисциплины:

кафедра МО ЭВМ

6. Составители программы: Горбенко Олег Данилович

ФИО

канд. физ.-мат.наук

доцент

ученая степень

ученое звание

oleg_dan@mail.ru

ПММ

e-mail

факультет

МО ЭВМ

кафедра

7. Рекомендована: НМС факультета ПММ, протокол №5 от 22.03. 2024 г.

(наименование рекомендующей структуры, дата, номер протокола)

8. Учебный год: 2024–2025

Семестр(-ы): 1

9. Цели и задачи учебной дисциплины:

Целями освоения учебной дисциплины являются:

- *Освоение основных возможностей программирования клиент-серверного взаимодействия в сети Интернет.*

- *Формирование способности осуществлять администрирование файловых систем и системного программного обеспечения инфокоммуникационной системы (ПК-4).*

Задачи учебной дисциплины:

- *Формирование умения использовать конкретные технологии разработки web-приложений, способы создания эффективного интерфейса взаимодействия пользователя с Web-вервером и сервером БД с использованием современных фреймворков.*

- *Закрепление знаний особенностей администрируемой операционной системы, особенности реализации сетевой технологии, регламенты проведения профилактических работ на администрируемой инфокоммуникационной системе. (ПК-4.1)*

10. Место учебной дисциплины в структуре ООП: (цикл, к которому относится дисциплина, требования к входным знаниям, умениям и компетенциям, дисциплины, для которых данная дисциплина является предшествующей)

ФТД, Факультативы..

Требования к входным знаниям, умениям и компетенциям:

- знание принципов функционирования компьютерных сетей;

- знание протоколов передачи данных в сетях;

- владение навыками разработки программ на языке программирования;

11. Планируемые результаты обучения по дисциплине/модулю (знания, умения, навыки), соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями) и индикаторами их достижения:

Код	Название компетенции	Код(ы)	Индикатор(ы)	Планируемые результаты обучения
-----	----------------------	--------	--------------	---------------------------------

ПК-4	Способен осуществлять администрирование файловых систем и системного программного обеспечения инфокоммуникационной системы, проводить анализ системных проблем обработки информации на уровне инфокоммуникационной системы.	ПК-4.1	Демонстрирует знания особенностей администрируемой операционной системы, особенности реализации сетевой технологии, регламенты проведения профилактических работ на администрируемой инфокоммуникационной системе.	Знать: - <i>современные технологии проектирования и разработки клиент-серверных приложений;</i> - <i>особенности архитектуры информационных ресурсов;</i> - <i>рынок современных средств проектирования интерфейсов.</i> Уметь: - <i>применять методы и средства проектирования интерфейсов;</i> Владеть: - <i>навыками разработки web-приложений для представления информационных ресурсов.</i>
------	---	--------	--	---

12. Объем дисциплины в зачетных единицах/час.(в соответствии с учебным планом) — 1 / 36 .

Форма промежуточной аттестации(зачет/экзамен) зачет.

13. Трудоемкость по видам учебной работы

Вид учебной работы		Трудоемкость	
		Всего	По семестрам
			1
Контактная работа		16	16
в том числе:	лекции		
	практические		
	лабораторные	16	16
	курсовая работа		
Самостоятельная работа		20	20
Промежуточная аттестация - зачет			
Итого:		36	36

13.1. Содержание дисциплины:

п/п	Наименование раздела дисциплины	Содержание раздела дисциплины	Реализация раздела дисциплины с помощью онлайн-курса, ЭУМК *
3. Лабораторные работы			
3.1	Основы работы web-сервера. Основы языка Python	<p>Основные функции web-сервера. Структура сервера <i>Apache</i>.</p> <p>Переменные и присваивание значений</p> <p>Операторы</p> <p>Стандартные типы данных в языке <i>Python</i></p> <p>Логические значения объектов</p> <p>Числа. Числовые операторы</p> <p>Встроенные и фабричные функции для работы с числами</p> <p>Последовательности и итерируемые объекты</p> <p>Списки. Строки</p> <p>Встроенные и фабричные функции последовательностей.</p> <p>Отображения: словари.</p> <p>Управление потоком выполнения.</p> <p>Условная инструкция. Циклы.</p> <p>Обработка исключений.</p> <p>Предложение <i>finally</i>.</p> <p>Возбуждение исключений с помощью инструкции <i>raise</i></p> <p>Файлы.</p> <p>Функции. Объявление и вызов функций.</p> <p>Функции как объекты. Анонимные функции.</p> <p>Декораторы.</p>	Фреймворки для web-приложений
3.2	Объектно-ориентированное программирование	<p>Определение классов. Создание экземпляров.</p> <p>Создание подклассов. Вложенные классы.</p> <p>Регулярные выражения.</p> <p>Модуль <i>re</i>.</p> <p>Поиск и соответствие.</p> <p>Модули. Изменяемость.</p> <p>Конструктор и метод инициализации.</p>	Фреймворки для web-приложений
3.3	Основы Django	<p>Создание проекта. Запуск сервера разработки.</p> <p>Создание приложения блога.</p> <p>Создание модели.</p> <p>Настройка базы данных.</p> <p>Использование сервера баз данных.</p>	Фреймворки для web-приложений

		Использование <i>SQLite</i> . Настройка автоматизированного приложения администрирования. Создание шаблона. Создание функции представления. Создание шаблона адреса URL.	
3.4	Создание динамических веб-сайтов .	Взаимодействие: <i>HTTP</i> , <i>URL</i> , запросы, ответы. Хранилища данных: <i>SQL</i> и реляционные базы данных. Представление: шаблоны отображения в разметку <i>HTML</i> и в другие форматы. Выделение уровней (<i>MVC</i>). Модели. Представления. Шаблоны.	-
3.5	Общий обзор архитектуры Django	Основные принципы <i>Django</i> . <i>DRY (Don't Repeat Yourself)</i> . Слабая зависимость и гибкость. Быстрая разработка.	Фреймворки для веб-приложений

13.2. Темы (разделы) дисциплины и виды занятий

№ п/п	Наименование темы (раздела) дисциплины	Виды занятий (часов)				
		Лекции	Практические	Лабораторные	Самостоятельная работа	Всего
1.	Основные функции web-сервера. Структура сервера <i>Apache</i> . Переменные и присваивание значений Операторы Стандартные типы данных в языке <i>Python</i> Логические значения объектов Числа. Числовые операторы Встроенные и фабричные функции для работы с числами Последовательности и итерируемые объекты Списки. Строки Встроенные и фабричные функции			2	2	4
2.	Отображения: словари. Управление потоком выполнения.			2	2	4

	<p>Условная инструкция. Циклы. Обработка исключений. Предложение <i>finally</i>. Возбуждение исключений с помощью инструкции <i>raise</i> Файлы. Функции. Объявление и вызов функций. Функции как объекты. Анонимные функции. Декораторы.</p>					
3.	<p>Определение классов. Создание экземпляров. Создание подклассов. Вложенные классы. Регулярные выражения. Модуль <i>re</i>. Поиск и соответствие. Модули. Изменяемость. Конструктор и метод инициализации</p>			2	2	4
4.	<p>Создание проекта. Запуск сервера разработки. Создание приложения блога. Создание модели. Настройка базы данных. Использование сервера баз данных. Использование <i>SQLite</i>.</p>			2	4	6
5.	<p>Настройка автоматизированного приложения администрирования. Создание шаблона. Создание функции представления. Создание шаблона адреса URL.</p>			2	4	6
6.	<p>Взаимодействие: <i>HTTP</i>, <i>URL</i>, запросы, ответы. Хранилища данных: <i>SQL</i> и реляционные базы данных.</p>			2	2	4

	Представление: шаблоны отображения в разметку <i>HTML</i> и в другие форматы.					
7.	Выделение уровней (<i>MVC</i>). Модели. Представления. Шаблоны.			2	2	4
8.	Основные принципы <i>Django</i> . <i>DRY (Don't Repeat Yourself)</i> . Слабая зависимость и гибкость. Быстрая разработка.			2	2	4
	Итого:	0	0	16	20	36

14. Методические указания для обучающихся по освоению дисциплины

1. При использовании дистанционных образовательных технологий и электронного обучения выполнять все указания преподавателей по работе на LMS-платформе, своевременно подключаться к online-занятиям, соблюдать рекомендации по организации самостоятельной работы.
2. Дополнительный материал, рекомендации, а также задания к лабораторным занятиям, к зачету размещаются преподавателем на платформе «Электронный университет» (moodle).
3. В течение семестра необходимо выполнить 2 лабораторные работы на компьютере. Работа над лабораторными заданиями должна быть завершена за неделю до окончания курса.

15. Перечень основной и дополнительной литературы, ресурсов интернет, необходимых для освоения дисциплины (список литературы оформляется в соответствии с требованиями ГОСТ и используется общая сквозная нумерация для всех видов источников)

а) основная литература:

№ п/п	Источник
1	Саммерфилд М. Python на практике [Электронный ресурс] : учебное пособие. — Электрон. Дан. — М. : ДМК Пресс, 2014. — 338 с. — Режим доступа: http://lanbook.lib.vsu.ru/books/element.php?p11_id=66480
2	Уэс Маккинли Python и анализ данных [Электронный ресурс] : . — Электрон. дан.

	— М. : ДМК Пресс, 2015. — 482 с. — Режим доступа: http://lanbook.lib.vsu.ru/books/element.php?pl1_id=73074
3	Златопольский, Д.М. Основы программирования на языке Python : учебник / Д.М. Златопольский. – Москва : ДМК Пресс, 2017. – 284 с. : ил. – Режим доступа: по подписке. – URL: https://biblioclub.ru/index.php?page=book&id=567144 (дата обращения: 19.12.2020). – ISBN 978-5-97060-552-3. – Текст : электронный.
4	Крахоткина, Е.В. Технологии разработки Internet-приложений : учебное пособие / Е.В. Крахоткина ; Северо-Кавказский федеральный университет. – Ставрополь : Северо-Кавказский Федеральный университет (СКФУ), 2016. – 124 с. : ил. – Режим доступа: по подписке. – URL: https://biblioclub.ru/index.php?page=book&id=459070 (дата обращения: 19.12.2020). – Библиогр. в кн. – Текст : электронный.

б) дополнительная литература:

№ п/п	Источник
3	Сузи, Роман. Python / Роман Сузи .— СПб. и др. : БХВ-Петербург, 2002 .— XI, 747 с.+ Прил. CD-ROM : ил., табл.
4	Головатый А. Django. Подробное руководство. 2-е издание / А.Головатый, Дж.Каплан-Мосс – Пер. с англ. – СПб.: Символ-Плюс, 2010 – 560 с.
5	Форсье Дж. Django. Разработка веб-приложений на Python / Дж.Форсье, П.Биссекс, У.Чан - Пер.с англ.- СПб.: Символ-Плюс, 2010.- 456 с

в) информационные электронно-образовательные ресурсы:

№ п/п	Источник
6	Онлайн-курс «Фреймворки для web-приложений», размещенный на LMS-платформе edu.vsu.ru
9	Гринберг, М. Разработка веб-приложений с использованием Flask на языке Python / М. Гринберг. — Москва : ДМК Пресс, 2014. — 272 с. — ISBN 978-5-97060-138-9. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/90103 (дата обращения: 19.12.2020). — Режим доступа: для авториз. пользователей.

16. Перечень учебно-методического обеспечения для самостоятельной работы (учебно-методические рекомендации, пособия, задачки, методические указания по выполнению практических (контрольных) работ и др.)

№ п/п	Источник
1	Бизли Д. Python. Подробный справочник. / Д. Бизли – Пер. с англ. – СПб.: Символ-Плюс, 2010. – 864 с., ил.
2	Дронов В.А. Django: практика создания web-сайтов на Python / В.А.Дронов – СПб:

	БХВ-Петербург, 2016.- 528 с.
3	Перепелица, Ф. А. Разработка интерактивных сайтов с использованием jQuery : учебное пособие / Ф. А. Перепелица. — Санкт-Петербург : НИУ ИТМО, 2015. — 142 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/91556 (дата обращения: 19.12.2020). — Режим доступа: для авториз. пользователей.

17. Образовательные технологии, используемые для реализации учебной дисциплины, включая дистанционные образовательные технологии (ДОТ), электронное обучение (ЭО), смешанное обучение):

При реализации дисциплины используются модульно-рейтинговая и личностно-ориентированные технологии обучения (ориентированные на индивидуальность студента, компьютерные и коммуникационные технологии). В рамках дисциплины предусмотрены следующие виды лекций: информационная, лекция-визуализация, лекция с применением обратной связи.

Web-технологии: размещение методических материалов, заданий для практической работы на платформе Электронного университета ВГУ (LMS moodle) и на персональных страницах преподавателя;

Компьютерные презентации с материалами лекций размещаются на платформе Электронного университета ВГУ (LMS moodle) и на персональных страницах преподавателя.

18. Материально-техническое обеспечение дисциплины:

Лекционная аудитория должна быть оборудована учебной мебелью, компьютером, мультимедийным оборудованием (проектор, экран, средства звуковоспроизведения), допускается переносное оборудование.

Практические занятия должны проводиться в специализированной аудитории, оснащенной учебной мебелью и персональными компьютерами с доступом в сеть Интернет (компьютерные классы, студии), мультимедийным оборудованием (мультимедийный проектор, экран, средства звуковоспроизведения), Число рабочих мест в аудитории должно быть таким, чтобы обеспечивалась индивидуальная работа студента на отдельном персональном компьютере.

Для самостоятельной работы необходимы компьютерные классы, помещения, оснащенные компьютерами с доступом к сети Интернет.

Программное обеспечение: Mozilla Firefox (свободное и/или бесплатное ПО), Notepad ++ (свободное и/или бесплатное ПО), Django, MySQL.

19. Оценочные средства для проведения текущей и промежуточной аттестаций

Порядок оценки освоения обучающимися учебного материала определяется содержанием следующих разделов дисциплины:

№ п/п	Наименование раздела дисциплины (модуля)	Компетенция(и)	Индикатор(ы) достижения компетенции	Оценочные средства
1.	<p>Основные функции web-сервера. Структура сервера <i>Apache</i>. Переменные и присваивание значений Операторы Стандартные типы данных в языке <i>Python</i> Встроенные и фабричные функции для работы с числами Последовательности и итерируемые объекты Списки. Строки Встроенные и фабричные функции</p>	ПК-4	ПК-4.1	<i>Лабораторная работа №1</i>
2.	<p>Определение классов. Создание экземпляров. Создание подклассов. Вложенные классы. Регулярные выражения. Модуль <i>re</i>. Поиск и соответствие. Модули. Изменяемость. Конструктор и метод инициализации Выделение уровней (<i>MVC</i>). Модели. Представления. Шаблоны. Основные принципы <i>Django</i>.</p>	ПК-4	ПК-4.1	<i>Лабораторная работа №2</i>

№ п/п	Наименование раздела дисциплины (модуля)	Компетенция(и)	Индикатор(ы) достижения компетенции	Оценочные средства
	<i>DRY (Don't Repeat Yourself).</i>			
Промежуточная аттестация: форма контроля - <u>зачет</u>			<i>Практическое задание</i> Лабораторная работа № 1 Лабораторная работа 2	

20 Типовые оценочные средства и методические материалы, определяющие процедуры оценивания

20.1 Текущий контроль успеваемости

Контроль успеваемости по дисциплине осуществляется с помощью следующих оценочных средств:

Лабораторные работы по проектированию и разработке динамических Web-страниц для информационной поддержки деятельности работника в некоторой предметной области. Работник должен получать доступ к данным об объектах своей деятельности, расположенным в БД на web-сервере.

Описание технологии проведения

Лабораторная работа включает в себя этапы: формализация постановки задачи, выбор структуры данных, описание модели взаимодействия пользователя с информационной системой, выбор системы управления базами данных, кодирование, отладка и тестирование разработанной динамической web-страницы.

Требования к выполнению

Язык программирования - Python.

Фреймворк - Django.

Web-страница должна предоставлять комфортный интерфейс пользователю.

20.2 Промежуточная аттестация

Промежуточная аттестация по дисциплине осуществляется с помощью следующих оценочных средств:

Лабораторная работа

Для оценивания результатов обучения на зачете используются следующие показатели:

1. Знание рынка современных средств проектирования интерфейсов.
2. Умение применять методы и средства проектирования интерфейсов;
3. Владение навыками разработки web-приложений для представления информационных ресурсов.

Для оценивания результатов обучения по окончании лабораторных практических занятий используются следующие показатели

зачтено	Полное выполнение всех лабораторных работ
не зачтено	Выполнение всех лабораторных работ с недостатками в организации ввода-вывода
	Выполнение всех лабораторных работ с ошибками в организации структур данных
	Наличие хотя бы одной невыполненной лабораторной работы.

Приложение

Лабораторная работа 1

Спроектируйте и разработайте динамическую Web-страницу для информационной поддержки деятельности преподавателя. Преподаватель должен получать доступ к данным о студентах, расположенным в БД на web-сервере.

Лабораторная работа 2

Спроектируйте и разработайте динамическую Web-страницу для информационной поддержки деятельности библиотекаря. Работник библиотеки должен получать доступ к данным о читателях, расположенным в БД на web-сервере.

Лабораторная работа 3

Спроектируйте и разработайте динамическую Web-страницу для информационной поддержки деятельности работника деканата. Работник должен получать доступ к данным о студентах и преподавателях факультета,, расположенным в БД на web-сервере.

Примерные варианты структур в базе данных.

Вариант 1.

1. Структура с именем STUDENT содержит следующие поля:

- NAME – фамилия и инициалы;
- GROUP – номер группы;
- SES - успеваемость (массив из пяти элементов).

2. Написать программу, выполняющую следующие действия :

- ввод с клавиатуры данных в массив STUD1, состоящий из десяти структур типа STUDENT; записи должны быть упорядочены по возрастанию содержимого поля GROUP;

- вывод на дисплей фамилий и номеров групп для всех студентов, включенных в массив, если средний балл студента больше 4,0;
- если таких нет, вывести соответствующее сообщение.

Вариант 2

1. Описать структуру с именем STUDENT, содержащую следующие поля:

- NAME - фамилия и инициалы;
- GROUP- номер группы;
- SES- успеваемость (массив из пяти элементов).

2. Написать программу, выполняющую следующие действия :

- ввод с клавиатуры данных в массив STUD1, состоящий из десяти структур типа STUDENT; записи должны быть упорядочены по алфавиту;
- вывод на дисплей фамилий и номеров групп для всех студентов, имеющих хотя бы одну оценку 2;
- если таких студентов нет, вывести соответствующее сообщение.

Вариант 3

Структуру с именем AEROFLOT содержит следующие поля:

- NAZN — название пункта назначения рейса;
- NUMR — номер рейса;
- TIP — тип самолета.

2. Написать программу, выполняющую следующие действия:

- ввод с клавиатуры данных в массив AIRPORT, состоящий из семи элементов типа AEROFLOT; записи должны быть упорядочены по возрастанию номера рейса;
- вывод на экран номеров рейсов и типов самолетов, вылетающих в пункт назначения, название которого совпало с названием, введенным с клавиатуры;
- если таких рейсов нет, выдать на дисплей соответствующее сообщение.

Вариант 4

Структура с именем WORKER содержит следующие поля:

- NAME — фамилия и инициалы работника;
- POS — название занимаемой должности;
- YEAR — год поступления на работу.

2. Написать программу, выполняющую следующие действия:

- ввод с клавиатуры данных в массив TABL, состоящий из десяти структур типа WORKER; записи должны быть размещены по алфавиту.
- вывод на дисплей фамилий работников, чей стаж работы в организации превышает значение, введенное с клавиатуры;
- если таких работников нет, вывести на дисплей соответствующее сообщение.

Вариант 5

Структура с именем TRAIN содержит следующие поля:

- NAZN — название пункта назначения;
- NUMR — номер поезда;

- TIME — время отправления.

2. Написать программу, выполняющую следующие действия:

- ввод с клавиатуры данных в массив RASP, состоящий из восьми элементов типа TRAIN; записи должны быть размещены в алфавитном порядке по названиям пунктов назначения;
- вывод на экран информации о поездах, отправляющихся после введенного с клавиатуры времени;
- если таких поездов нет, выдать на дисплей соответствующее сообщение.