

**Министерство науки и высшего образования РФ**  
**Федеральное государственное бюджетное образовательное учреждение**  
**высшего образования**  
**«Национальный исследовательский университет «МЭИ»**

---

Направление подготовки/специальность: 09.03.01 Информатика и вычислительная техника

Наименование образовательной программы: Вычислительные машины, комплексы, системы и сети

Уровень образования: высшее образование - бакалавриат

Форма обучения: Очная

**Рабочая программа дисциплины**  
**WEB-ТЕХНОЛОГИИ**


<b>Блок:</b>	<b>Блок 1 «Дисциплины (модули)»</b>
<b>Часть образовательной программы:</b>	<b>Часть, формируемая участниками образовательных отношений</b>
<b>№ дисциплины по учебному плану:</b>	<b>Б1.Ч.06</b>
<b>Трудоемкость в зачетных единицах:</b>	<b>5 семестр - 3; 6 семестр - 3; всего - 6</b>
<b>Часов (всего) по учебному плану:</b>	<b>216 часов</b>
<b>Лекции</b>	<b>5 семестр - 16 часов; 6 семестр - 14 часов; всего - 30 часов</b>
<b>Практические занятия</b>	<b>не предусмотрено учебным планом</b>
<b>Лабораторные работы</b>	<b>5 семестр - 16 часов; 6 семестр - 12 часов; всего - 28 часа</b>
<b>Консультации</b>	<b>6 семестр - 14 часов;</b>
<b>Самостоятельная работа</b>	<b>5 семестр - 75,7 часа; 6 семестр - 63,4 часа; всего - 139,1 часа</b>
<b>в том числе на КП/КР</b>	<b>6 семестр - 14 часов;</b>
<b>Иная контактная работа</b>	<b>6 семестр - 4 часа;</b>
<b>включая:</b>	
<b>Кейс (решение конкретных производственных ситуаций)</b>	
<b>Промежуточная аттестация:</b>	
<b>Зачет с оценкой</b>	<b>5 семестр - 0,3 часа;</b>
<b>Зачет с оценкой</b>	<b>6 семестр - 0,3 часа;</b>
<b>Защита курсовой работы</b>	<b>6 семестр - 0,3 часа; всего - 0,9 часа</b>

**Москва 2025**

**ПРОГРАММУ СОСТАВИЛ:**

Преподаватель

---

	Подписано электронной подписью ФГБОУ ВО «НИУ «МЭИ»	
	Сведения о владельце ЦЭП МЭИ	
	Владелец	Мишин А.А.
	Идентификатор	Rf569097b-MishinAIA-2f64a6ba


А.А. Мишин

---

**СОГЛАСОВАНО:**

Руководитель  
образовательной программы

---


	Подписано электронной подписью ФГБОУ ВО «НИУ «МЭИ»	
	Сведения о владельце ЦЭП МЭИ	
	Владелец	Гольцов А.Г.
	Идентификатор	R64210572-GoltsovAG-cebbd3e8

А.Г. Гольцов

---

Заведующий выпускающей  
кафедрой

---

	Подписано электронной подписью ФГБОУ ВО «НИУ «МЭИ»	
	Сведения о владельце ЦЭП МЭИ	
	Владелец	Вишняков С.В.
	Идентификатор	R35b26072-VishniakovSV-02810d9

С.В. Вишняков

---

## 1. ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

**Цель освоения дисциплины:** Изучение технологий разработки динамических веб-приложений и систем управления данными с использованием языка Python, фреймворка Django и реляционных баз данных..

### Задачи дисциплины

- Приобретение навыков настройки рабочего окружения и использования систем контроля версий для совместной разработки.;
- Приобретение навыков настройки рабочего окружения и использования систем контроля версий для совместной разработки.;
- Формирование практических умений по созданию, отладке и развертыванию веб-приложений в современных операционных системах.

Формируемые у обучающегося **компетенции** и запланированные **результаты обучения** по дисциплине, соотнесенные с **индикаторами достижения компетенций**:

Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Запланированные результаты обучения
РПК-1 Способен принимать участие в концептуальном, функциональном и логическом проектировании компьютерных систем	ИД-ЗРПК-1 Демонстрирует знание языков программирования высокого и низкого уровня, методов разработки и отладки программного обеспечения	знать: <ul style="list-style-type: none"><li>- назначение и структуру тестов в Pytest и Django;</li><li>- структуру и механизмы аутентификации и авторизации пользователей в Django;</li><li>- синтаксис и принципы работы условных операторов;</li><li>- назначение стандартных модулей Python;</li><li>- основные коллекционные типы и их свойства;</li><li>- принципы организации циклов for и while;</li><li>- основы конфигурации среды разработки Python;</li><li>- принципы работы Git и GitHub;</li><li>- инструменты анализа и проверки кода Python;</li><li>- объектную модель Python;</li><li>- ключевые принципы ООП;</li><li>- принципы работы протокола HTTP и типы запросов;</li><li>- назначение URL-адресов и различие абсолютных и относительных путей;</li><li>- структуру Django-проекта и назначение основных компонентов;</li><li>- механизм маршрутизации Django и связь URL с view-функциями;</li><li>- структуру шаблонов Django и принципы их наследования;</li><li>- типы связей между таблицами реляционных БД;</li><li>- структуру и назначение моделей Django;</li></ul>

Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Запланированные результаты обучения
		<ul style="list-style-type: none"> <li>- назначение и возможности админ-панели Django;</li> <li>- основные операции CRUD в Django ORM;</li> <li>- принципы работы форм и валидации данных в Django;</li> <li>- базовые типы и операции Python.</li> </ul> <p>уметь:</p> <ul style="list-style-type: none"> <li>- создавать и подключать приложения в проекте Django;</li> <li>- создавать и обрабатывать формы, включая ModelForm и кастомные валидаторы;</li> <li>- формировать и интерпретировать URL-параметры запроса;</li> <li>- выполнять базовые операции с репозиторием;</li> <li>- использовать импортируемые типы данных и функции внешних модулей;</li> <li>- использовать шаблонизатор Django для вывода данных и организации верстки;</li> <li>- описывать маршруты с параметрами и связывать их с view;</li> <li>- описывать модели и связи в Django ORM;</li> <li>- писать и запускать тесты для моделей, view-функций и форм;</li> <li>- получать и фильтровать данные из моделей;</li> <li>- применять наследование и полиморфизм для расширения поведения и согласованного интерфейса;</li> <li>- применять отладку и автоматическую проверку стиля;</li> <li>- применять циклы и управляющие конструкции для обработки последовательностей;</li> <li>- применять числовые и логические типы в вычислениях;</li> <li>- проектировать простые типы;</li> <li>- проектировать таблицы с внешними ключами;</li> <li>- реализовывать выбор логики выполнения программы через if, elif, else;</li> <li>- реализовывать регистрацию, вход и</li> </ul>

Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Запланированные результаты обучения
		ограничение доступа с помощью LoginRequiredMixin и декоратора login_required; - регистрировать и настраивать модели в админ-зоне; - создавать и изменять коллекции для хранения и обработки данных; - создавать и настраивать виртуальное окружение; - анализировать структуру HTTP-запроса и ответа.
РПК-1 Способен принимать участие в концептуальном, функциональном и логическом проектировании компьютерных систем	ИД-5 <sub>РПК-1</sub> Осуществляет разработку аппаратных и программных средств различного назначения в соответствии с техническим заданием	знать: - принципы деплоя и сборки CI/CD в Docker и GitHub Actions; - назначение и структуру docker-compose.yml; - основные команды управления контейнерами Docker; - принципы контейнеризации и структуру Docker-образов; - принципы взаимодействия REST API с клиентскими приложениями; - принципы аутентификации, авторизации и троттлинга в DRF; - структуру DRF и базовые механизмы конфигурации API; - архитектуру DRF и принципы работы сериализаторов, вьюсетов и роутеров; - принципы взаимодействия с внешними сервисами через HTTP-запросы и обработку ошибок; - назначение API и различия между Client API и Bot API.  уметь: - публиковать образы на Docker Hub и разворачивать их на удалённом сервере; - описывать и запускать мультиконтейнерные приложения; - просматривать, запускать и останавливать контейнеры на сервере; - собирать и запускать контейнеры для Python-проектов; - отправлять и принимать JSON-данные с помощью HTTP-запросов; - применять классы разрешений и ограничения скорости запросов; - подключать и настраивать сериализаторы, пагинацию и

Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Запланированные результаты обучения
		фильтрацию; - создавать и настраивать API с помощью сериализаторов и ViewSet; - выполнять запросы к внешнему API, обрабатывать ответы и исключения; - подключаться к API и обрабатывать ключи доступа через переменные окружения.

## 2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ ВО

Дисциплина относится к основной профессиональной образовательной программе Вычислительные машины, комплексы, системы и сети (далее – ОПОП), направления подготовки 09.03.01 Информатика и вычислительная техника, уровень образования: высшее образование - бакалавриат.

Базируется на уровне среднего общего образования.

Результаты обучения, полученные при освоении дисциплины, необходимы при выполнении выпускной квалификационной работы.

### 3. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

#### 3.1 Структура дисциплины

Общая трудоемкость дисциплины составляет 6 зачетных единиц, 216 часов.

№ п/п	Разделы/темы дисциплины/формы промежуточной аттестации	Всего часов на раздел	Семестр	Распределение трудоемкости раздела (в часах) по видам учебной работы										Содержание самостоятельной работы/ методические указания
				Контактная работа							СР			
				Лек	Лаб	Пр	Консультация		ИКР		ПА	Работа в семестре	Подготовка к аттестации /контроль	
КПР	ГК	ИККП	ТК											
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	Основы Python	21.2	5	3.0	3.0	-	-	-	-	-	-	15.2	-	<b><u>Подготовка к практическим занятиям:</u></b> Изучение материала по разделу "Основы Python" подготовка к выполнению заданий на практических занятиях
1.1	Погружение в Python	2.9		0.4	0.4	-	-	-	-	-	-	2.1	-	
1.2	Простые типы данных	2.9		0.4	0.4	-	-	-	-	-	-	2.1	-	
1.3	Ветвления	2.9		0.4	0.4	-	-	-	-	-	-	2.1	-	
1.4	Импортируемые типы данных	2.9		0.4	0.4	-	-	-	-	-	-	2.1	-	
1.5	Коллекции	2.9		0.4	0.4	-	-	-	-	-	-	2.1	-	
1.6	Итерации и циклы	2.9		0.4	0.4	-	-	-	-	-	-	2.1	-	
1.7	Проект. Приложение «Холодильник»	3.8		0.6	0.6	-	-	-	-	-	-	2.6	-	
2	Настройка рабочего окружения	9.6		1.6	1.6	-	-	-	-	-	-	6.4	-	<b><u>Подготовка к практическим занятиям:</u></b> Изучение материала по разделу "Настройка рабочего окружения" подготовка к выполнению заданий на практических занятиях
2.1	Настройка рабочего окружения	2.4		0.4	0.4	-	-	-	-	-	-	1.6	-	
2.2	Система контроля и управления версиями	2.4		0.4	0.4	-	-	-	-	-	-	1.6	-	
2.3	Требования к коду	2.4		0.4	0.4	-	-	-	-	-	-	1.6	-	
2.4	Отладка программ	2.4		0.4	0.4	-	-	-	-	-	-	1.6	-	
3	ООП и другие полезные вещи, которые надо знать	17.0		2.6	2.6	-	-	-	-	-	-	11.8	-	<b><u>Подготовка к практическим занятиям:</u></b> Изучение материала по разделу "ООП и другие полезные вещи, которые надо знать" подготовка к выполнению заданий на практических занятиях
3.1	Объекты и классы	3.2		0.5	0.5	-	-	-	-	-	-	2.2	-	
3.2	Знакомство с ООП	3.2		0.5	0.5	-	-	-	-	-	-	2.2	-	
3.3	Python: новый уровень	3.2		0.5	0.5	-	-	-	-	-	-	2.2	-	
3.4	Расширенные возможности Python	3.2		0.5	0.5	-	-	-	-	-	-	2.2	-	
3.5	Проект. Игра «Изгиб	4.2		0.6	0.6	-	-	-	-	-	-	3	-	

	Питона»												
4	Основы веб-разработки	19.00	2.7 0	2.7 0	-	-	-	-	-	-	13.6	-	<u><b>Подготовка к практическим занятиям:</b></u> Изучение материала по разделу "Основы веб-разработки" подготовка к выполнению заданий на практических занятиях <u><b>Изучение материалов литературных источников:</b></u> [1], 29-100
4.1	Протокол HTTP	2.50	0.3 5	0.3 5	-	-	-	-	-	-	1.8	-	
4.2	Сетевые запросы	2.50	0.3 5	0.3 5	-	-	-	-	-	-	1.8	-	
4.3	Фреймворк Django	2.50	0.3 5	0.3 5	-	-	-	-	-	-	1.8	-	
4.4	Пути и view-функции	2.50	0.3 5	0.3 5	-	-	-	-	-	-	1.8	-	
4.5	Вёрстка для бэкендера	2.50	0.3 5	0.3 5	-	-	-	-	-	-	1.8	-	
4.6	HTML и шаблоны Django	2.50	0.3 5	0.3 5	-	-	-	-	-	-	1.8	-	
4.7	Проект. Приложение «Блогикум». Часть 1	4.0	0.6	0.6	-	-	-	-	-	-	2.8	-	<u><b>Подготовка к практическим занятиям:</b></u> Изучение материала по разделу "Базы данных и Django ORM" подготовка к выполнению заданий на практических занятиях
5	Базы данных и Django ORM	19.7	2.6	2.6	-	-	-	-	-	-	14.5	-	
5.1	Введение в базы данных	3.1	0.4	0.4	-	-	-	-	-	-	2.3	-	
5.2	Отношения между таблицами	3.1	0.4	0.4	-	-	-	-	-	-	2.3	-	
5.3	Django ORM. Модели	3.1	0.4	0.4	-	-	-	-	-	-	2.3	-	
5.4	Админ-зона Django	3.1	0.4	0.4	-	-	-	-	-	-	2.3	-	
5.5	Django ORM. Получение информации из БД	3.1	0.4	0.4	-	-	-	-	-	-	2.3	-	
5.6	Проект. Приложение «Блогикум». Часть 2. Публикации	4.2	0.6	0.6	-	-	-	-	-	-	3	-	<u><b>Подготовка к практическим занятиям:</b></u> Изучение материала по разделу "Формы в Django" подготовка к выполнению заданий на практических занятиях
6	Формы в Django	15.00	2.5 0	2.5 0	-	-	-	-	-	-	10.0	-	
6.1	Работа с формами	2.70	0.4 5	0.4 5	-	-	-	-	-	-	1.8	-	
6.2	Представления: расширенные возможности	2.70	0.4 5	0.4 5	-	-	-	-	-	-	1.8	-	
6.3	Пользователи в	2.70	0.4	0.4	-	-	-	-	-	-	1.8	-	



	Django			5	5								
6.4	Доработка проекта	2.70		0.4	0.4	-	-	-	-	-	-	1.8	-
6.5	Проект. Приложение «Блогикум». Часть 3. Доработка	4.2		0.7	0.7	-	-	-	-	-	-	2.8	-
7	Тестирование на Python	6.2		1.0	1.0	-	-	-	-	-	-	4.2	-
7.1	Тестирование	2.0		0.3	0.3	-	-	-	-	-	-	1.4	-
7.2	Библиотеки для тестирования	2.0		0.3	0.3	-	-	-	-	-	-	1.4	-
7.3	Pytest для Django	2.2		0.4	0.4	-	-	-	-	-	-	1.4	-
	Зачет с оценкой	0.3		-	-	-	-	-	-	-	0.3	-	-
	<b>Всего за семестр</b>	<b>108.00</b>		<b>16.0</b>	<b>16.0</b>	-	-	-	-	-	<b>0.3</b>	<b>75.7</b>	-
	<b>Итого за семестр</b>	<b>108.00</b>		<b>16.0</b>	<b>16.0</b>	-	-	-	-	-	<b>0.3</b>	<b>75.7</b>	-
8	Работа с API	11.2	6	2	1.6	-	-	-	-	-	-	7.6	-
8.1	Что такое API	5.6		1	0.8	-	-	-	-	-	-	3.8	-
8.2	Работа с внешними API	5.6		1	0.8	-	-	-	-	-	-	3.8	-
9	Django REST Framework	15.2		2.8	2.6	-	-	-	-	-	-	9.8	-
9.1	Django Rest Framework	6.0		1.2	1	-	-	-	-	-	-	3.8	-
9.2	Проектная работа. Проект CRUD для Yatube	9.2		1.6	1.6	-	-	-	-	-	-	6	-
10	Настройка API-сервиса	21.4		4.0	3.4	-	-	-	-	-	-	14	-
10.1	Права лимиты запросов	6.0		1.2	0.8	-	-	-	-	-	-	4	-
10.2	Взаимодействие фронтенда и бэкенда	6.2		1.2	1	-	-	-	-	-	-	4	-
10.3	Проект. API для Yatube	9.2		1.6	1.6	-	-	-	-	-	-	6	-
11	Docker для Python-разработчика	27.6		5.2	4.4	-	-	-	-	-	-	18	-
11.1	Работа на сервере	6.0		1.2	0.8	-	-	-	-	-	-	4	-

**Подготовка к практическим занятиям:**

Изучение материала по разделу "Тестирование на Python" подготовка к выполнению заданий на практических занятиях

**Подготовка к практическим занятиям:**

Изучение материала по разделу "Работа с API" подготовка к выполнению заданий на практических занятиях

**Подготовка к практическим занятиям:**

Изучение материала по разделу "Django REST Framework" подготовка к выполнению заданий на практических занятиях

**Подготовка к практическим занятиям:**

Изучение материала по разделу "Настройка API-сервиса" подготовка к выполнению заданий на практических занятиях

**Подготовка к практическим занятиям:**

Изучение материала по разделу "Docker для Python-разработчика" подготовка к

11.2	Docker: приложение в коробке	6.2		1.2	1	-	-	-	-	-	-	4	-	выполнению заданий на практических занятиях
11.3	Автоматизация. Деплой на сервер	6.2		1.2	1	-	-	-	-	-	-	4	-	
11.4	Проект. Контейнеры и CI/CD для Kittygram	9.2		1.6	1.6	-	-	-	-	-	-	6	-	
	Зачет с оценкой	0.3		-	-	-	-	-	-	-	0.3	-	-	
	Курсовая работа (КР)	32.3		-	-	-	14	-	4	-	0.3	14	-	
	Всего за семестр	108.0		14.0	12.0	-	14	-	4	-	0.6	63.4	-	
	Итого за семестр	108.0		14.0	12.0	-	14		4		0.6	63.4		
	<b>ИТОГО</b>	<b>216.00</b>	-	<b>30.00</b>	<b>28.00</b>	-	<b>14</b>		<b>4</b>		<b>0.9</b>	<b>139.1</b>		

**Примечание:** Лек – лекции; Лаб – лабораторные работы; Пр – практические занятия; КПр – аудиторные консультации по курсовым проектам/работам; ИККП – индивидуальные консультации по курсовым проектам/работам; ГК- групповые консультации по разделам дисциплины; СР – самостоятельная работа студента; ИКР – иная контактная работа; ТК – текущий контроль; ПА – промежуточная аттестация

## 3.2 Краткое содержание разделов

### 1. Основы Python

#### 1.1. Погружение в Python

Введение. Основы Python. Переменные и их значения. Типизация и типы данных. Константы. Функции. Правписание. Пишем функции. Модули в Python. Делаем ошибки. Практика по теме.

#### 1.2. Простые типы данных

Числовые типы. Значение None. Логический тип данных и операторы сравнения. Практика по теме.

#### 1.3. Ветвления

Ветвление if...else. if...else: задача. Логические операторы. Практика по теме.

#### 1.4. Импортируемые типы данных

Модуль decimal — когда точность важна. Дата и время: ещё один тип данных. Практика по теме.

#### 1.5. Коллекции

Коллекции в Python. Последовательности: список. Последовательности: кортеж. Изменяемые и неизменяемые типы данных. Последовательности: строки. Последовательности: диапазон. Работа с последовательностями. Словарь. Работа со словарями. Неупорядоченные коллекции: множество. Практика по теме.

#### 1.6. Итерации и циклы

Цикл for. Цикл while. Управление циклами. Списковые и словарные включения. Списковые включения: сценарии использования на практике. Практика по теме.

#### 1.7. Проект. Приложение «Холодильник»

Подготовка к работе над проектом. Проект. Приложение «Холодильник». Обратная связь о курсе.

### 2. Настройка рабочего окружения

#### 2.1. Настройка рабочего окружения

Введение. Стартовый набор разработчика. Терминал, консоль и командная строка. Работаем в командной строке. Установка интерпретатора Python. Выбор и настройка редактора кода. Виртуальное окружение. pip, uv, модули и виртуальное окружение. Видеолекция: работа в виртуальном окружении. Практика по теме.

#### 2.2. Система контроля и управления версиями

Знакомство с Git и GitHub. Настройка Git. Настройка GitHub. Создание и клонирование репозитория. Отслеживание файлов проекта. Работаем с Git. История изменения кода. Работа с Git через интерфейс VS Code. Видеолекция: работа с Git и GitHub. Практика по теме.

#### 2.3. Требования к коду

Оформление кода, линтеры и формatters. Аннотация типов. Аннотации типов: задача. Практика по теме.

#### 2.4. Отладка программ

Ошибки и баги. Дебаггинг через VS Code. Видеолекция: отладка программ. Практика по теме. Обратная связь о курсе.

### 3. ООП и другие полезные вещи, которые надо знать

#### 3.1. Объекты и классы

Введение. ООП. Классы и объекты в Python. Собственные классы и объекты. Атрибуты класса и объекта. Методы объекта. Магический метод `__str__`. Практика по теме.

#### 3.2. Знакомство с ООП

Принципы ООП. Наследование. Полиморфизм. Инкапсуляция. Практика по теме. Итоги.

#### 3.3. Python: новый уровень

Новая задача. Пользовательские модули и пакеты. Интроспекция. Документирование кода. Исключения. Доработка проекта. Работа с файлами. Контекстные менеджеры. Новый интерфейс для проекта. Практика по теме.

#### 3.4. Расширенные возможности Python

Пространство имён и области видимости. Итераторы и генераторы. Lambda-функции. Декораторы. Декораторы для методов класса. Видеолекция: декораторы. Практика по теме.

#### 3.5. Проект. Игра «Изгиб Питона»

Подготовка к работе над проектом. Подготовка проекта к автотестам. Проверка проекта автотестами. Проект. Игра «Изгиб Питона». Чек-лист проекта «Изгиб Питона». Обратная связь о курсе.

### 4. Основы веб-разработки

#### 4.1. Протокол HTTP

Введение в спринт. Что такое протокол. Подробнее об URL. Кириллица в адресной строке. Исходный код страницы. HTTP-ответы. HTTP-запросы. Методы HTTP-запросов.

#### 4.2. Сетевые запросы

Python вместо браузера. Передаём параметры в URL. Заголовки запросов и ответов. Обработка ошибок. Анфиса на все руки.

#### 4.3. Фреймворк Django

Фреймворк Django. Создание Django-проекта. Приложения в Django-проекте.

#### 4.4. Пути и view-функции

Планирование адресов и конвертеры путей. Распределение адресов по приложениям. View-функции: от запроса до ответа.

#### 4.5. Вёрстка для бэкендера

Веб-страницы: язык HTML. Структура HTML-документа. CSS: стили, селекторы и синтаксис. Иерархия HTML-документа и наследование стилей. Фреймворк Bootstrap: быстрая HTML-вёрстка.

#### 4.6. HTML и шаблоны Django

HTML в Django: шаблоны веб-страниц. HTML-шаблоны из частей: теги `include`, `extends`, `block`. Словарь контекста, ветвление и циклы в шаблонах. Ссылки в шаблонах: `name` и `namespace`. Фильтры в шаблонах: модифицируем контекст страницы. Подключение стилей, картинок и скриптов к шаблонам. Из тренажёра на компьютер. Настройка VS Code. Правила оформления HTML и шаблонов Django. Создание шаблонов из HTML-документов. Практика.

#### 4.7. Проект. Приложение «Блогикум». Часть 1

Подготовка к работе над проектом. Проект. Приложение «Блогикум». Часть 1. Чек-лист проекта «Блогикум». Часть 1. Обратная связь о курсе.

### 5. Базы данных и Django ORM

#### 5.1. Введение в базы данных

Введение в спринт. Базы данных: реляционные и не только. SQL: язык структурированных запросов. Первые запросы, первые результаты. Работа с базой данных из Python. Получение данных. Практика запросов. Сортировка, ограничение и сдвиг выборки. Агрегирующие функции. Группировка записей в выборке.

#### 5.2. Отношения между таблицами

Отношения между таблицами. Отношения «один к одному». Отношения «многие к одному». Объединение таблиц: JOIN. Отношения между таблицами. «Многие ко многим». Изменение таблиц в БД. Ссылочная целостность.

#### 5.3. Django ORM. Модели

Описание таблиц с помощью классов. Описание связей между моделями. Наследование от абстрактных моделей. Миграции. Django shell. CRUD-операции в Django ORM. Заполнение БД и выгрузка информации из неё.

#### 5.4. Админ-зона Django

Создание суперпользователя. Регистрация моделей в админке. Локализация и перевод. Тонкая настройка админ-зоны.

#### 5.5. Django ORM. Получение информации из БД

Django Debug Toolbar. SELECT и FROM через ORM. Фильтрация WHERE: методы `filter()` и `exclude`. Условия AND, OR, NOT. Q-объекты. Сортировка ORDER BY, ограничение LIMIT и сдвиг OFFSET. Получение отдельных объектов модели. Запросы к связанным моделям.

#### 5.6. Проект. Приложение «Блогикум». Часть 2. Публикации

Подготовка к работе над проектом. Проект. Приложение «Блогикум». Часть 2. Публикации. Чек-лист проекта «Блогикум». Часть 2. Обратная связь о курсе.

### 6. Формы в Django

### 6.1. Работа с формами

Вводный урок к спринту. Формы в HTML. Формы в Django: класс Forms. Обработка данных, полученных из веб-формы. Подключение стилей Bootstrap к HTML-форме. Формы на основе моделей. Защита от атак: csrf-токен. Создание записей в БД через форму. Редактирование и удаление объектов через форму. Валидация формы. Работа с изображениями в формах.

### 6.2. Представления: расширенные возможности

Постраничный вывод информации. CBV: view-классы вместо view-функций. Представление отдельного объекта: класс DetailView. Статичные страницы и класс TemplateView.

### 6.3. Пользователи в Django

Работа с пользователями в Django. Отправка писем. Эмуляция почтового сервера. Настройка страниц входа и выхода пользователей. Настройка страниц смены и восстановления пароля. Настройка страницы регистрации.

### 6.4. Доработка проекта

Кастомные страницы ошибок. Поздравления: комментирование записей. Оптимизация запросов к связанным моделям.

### 6.5. Проект. Приложение «Блогикум». Часть 3. Доработка

Подготовка к работе над проектом. Проект. Приложение «Блогикум». Часть 3. Доработка. Как сдать проект. Обратная связь о курсе.

## 7. Тестирование на Python

### 7.1. Тестирование

Введение к спринту. Зачем программисту тестирование?. Что тестировать и как писать тесты?. Ключевое слово assert.

### 7.2. Библиотеки для тестирования

Библиотека unittest. Библиотека pytest. Выборочный запуск тестов. Запуск упавших тестов. Кеш pytest. Отладка тестов. Маркеры pytest: пропуск тестов, ожидаемое падение и параметризация. Фикстуры в pytest. Файл конфигураций.

### 7.3. Pytest для Django

Плагин pytest-django. Тестирование маршрутов. Тестирование контента. Тестирование логики приложения. Обратная связь о курсе.

## 8. Работа с API

### 8.1. Что такое API

Вводный урок к спринту. Форматы обмена данными. Взаимодействие программ по сети. API First. Архитектура REST. REST: ресурсы, эндпоинты и HTTP-методы. Исследование запросов. Механизмы авторизации, протокол OAuth 2.0.

### 8.2. Работа с внешними API

Client API в Telegram. Bot API в Telegram. Отправка сообщения из кода. Обработка входящих сообщений и команд. Всё о чате и пользователе: объект message. Отправка изображений. А где взять котиков?. Кнопки вместо текста. Код-ревью: хранение секретов. Код-ревью: обработка исключений. Код-ревью: переиспользование функций. Журнал ошибок. Обратная связь о курсе.

## 9. Django REST Framework

### 9.1. Django Rest Framework

Вводный урок к спринту. REST API: Проектирование. Инструментарий для тестирования API. Преобразование форматов. Сериализаторы. Учебный проект Kittygram. View-функции API. View-классы API. Вьюсеты и роутеры. Сериализаторы для связанных моделей. Сериализаторы: дополнительные настройки. Регулярные выражения. Вьюсеты. Расширенные возможности. Аутентификация по токену. JWT + Djoser. Kittygram 2: новые возможности.

### 9.2. Проектная работа. Проект CRUD для Yatube

Подготовка к работе над проектом. Проект спринта: CRUD для Yatube. Чек-лист проекта «CRUD для Yatube». Обратная связь о курсе.

## 10. Настройка API-сервиса

### 10.1. Права лимиты запросов

Что будет в спринте. Проверка прав: Permissions. Throttling: ограничение количества запросов. Пагинация в API. Фильтрация, сортировка и поиск.

### 10.2. Взаимодействие фронтенда и бэкенда

Добавление фотографий. Упрощаем взаимодействие с фронтендом. Multi Page Application vs Single Page Application. Документация для API. SPA для Kittygram. CORS и политика единого источника. Обзор проекта Kittygram.

### 10.3. Проект. API для Yatube

Подготовка к работе над проектом. Проверка проекта. Проект. API для Yatube. Чек-лист для проверки. Обратная связь о курсе.

## 11. Docker для Python-разработчика

### 11.1. Работа на сервере

Введение в спринт. Введение в DevOps. Куда деплоить приложение. Подключение к серверу. Внеплановая задача и первый деплой. Server Gateway Interface: выбор, установка и управление. Веб- и обратный прокси-сервер Nginx: установка и настройка.

### 11.2. Docker: приложение в коробке

Новая задача: проект в коробочке. Виртуальные машины и контейнеры. Установка Docker. Запуск приложений в docker-контейнерах. Упаковка проекта в Docker-образ. Сервер Gunicorn и Docker volume. DockerHub. Архитектура приложения в Docker. PostgreSQL и Docker Network. Docker-compose: сборка проекта. Docker-compose: Nginx и статика.

### 11.3. Автоматизация. Деплой на сервер

Автоматизация деплоя: CI/CD. GitHub Actions. Первый workflow. Workflow для CI: линтер и тесты. Workflow для CD: сборка образов и перезапуск контейнеров.

#### 11.4. Проект. Контейнеры и CI/CD для Kittygram

Подготовка к работе над проектом. Проект. Контейнеры и CI/CD для Kittygram. Чек-лист. Автоматическое тестирование и деплой проекта Kittygram с применением CI/CD. Обратная связь о курсе.

### 3.3. Темы практических занятий не предусмотрено

### 3.4. Темы лабораторных работ

1. Основы Python;
2. Настройка рабочего окружения;
3. ООП и другие полезные вещи, которые надо знать;
4. Основы веб-разработки;
5. Базы данных и Django ORM;
6. Формы в Django;
7. Тестирование на Python;
8. Работа с API;
9. Django REST Framework;
10. Настройка API-сервиса;
11. Docker для Python-разработчика.

### 3.5 Консультации

#### Групповые консультации по разделам дисциплины (ГК)

1. Обсуждение материалов по кейсам раздела "Основы Python"
2. Обсуждение материалов по кейсам раздела "Настройка рабочего окружения"
3. Обсуждение материалов по кейсам раздела "ООП и другие полезные вещи, которые надо знать"
4. Обсуждение материалов по кейсам раздела "Основы веб-разработки"
5. Обсуждение материалов по кейсам раздела "Базы данных и Django ORM"
6. Обсуждение материалов по кейсам раздела "Формы в Django"
7. Обсуждение материалов по кейсам раздела "Тестирование на Python"
8. Обсуждение материалов по кейсам раздела "Работа с API"
9. Обсуждение материалов по кейсам раздела "Django REST Framework"
10. Обсуждение материалов по кейсам раздела "Настройка API-сервиса"
11. Обсуждение материалов по кейсам раздела "Docker для Python-разработчика"

### 3.6 Тематика курсовых проектов/курсовых работ 6 Семестр

Курсовая работа (КР)

#### **График выполнения курсового проекта**

Неделя	1 - 8	9 - 14	Зачетная
Раздел курсового проекта	1	1	Защита курсового проекта
Объем раздела, %	20	80	-
Выполненный объем нарастающим	20	100	-



итогом, %			
-----------	--	--	--

Номер раздела	Раздел курсового проекта
1	Выполнение финального проекта

### 3.7. Соответствие разделов дисциплины и формируемых в них компетенций

Запланированные результаты обучения по дисциплине (в соответствии с разделом 1)	Коды индикаторов	Номер раздела дисциплины (в соответствии с п.3.1)											Оценочное средство (тип и наименование)
		1	2	3	4	5	6	7	8	9	10	11	
<b>Знать:</b>													
базовые типы и операции Python	ИД-ЗРПК-1	+											Кейс (решение конкретных производственных ситуаций)/Спринт 1. Основы Python
принципы работы форм и валидации данных в Django	ИД-ЗРПК-1						+						Кейс (решение конкретных производственных ситуаций)/Спринт 6. Формы в Django и Спринт 7. Тестирование на Python
основные операции CRUD в Django ORM	ИД-ЗРПК-1					+							Кейс (решение конкретных производственных ситуаций)/Спринт 5. Базы данных и Django ORM
назначение и возможности админ-панели Django	ИД-ЗРПК-1					+							Кейс (решение конкретных производственных ситуаций)/Спринт 5. Базы данных и Django ORM
структуру и назначение моделей Django	ИД-ЗРПК-1					+							Кейс (решение конкретных производственных ситуаций)/Спринт 5. Базы данных и Django ORM
типы связей между таблицами реляционных БД	ИД-ЗРПК-1					+							Кейс (решение конкретных производственных ситуаций)/Спринт 5. Базы данных и Django ORM
структуру шаблонов Django и принципы их наследования	ИД-ЗРПК-1				+								Кейс (решение конкретных производственных ситуаций)/Спринт 4. Основы веб-разработки
механизм маршрутизации Django и связь URL с view-функциями	ИД-ЗРПК-1				+								Кейс (решение конкретных производственных ситуаций)/Спринт 4. Основы веб-разработки
структуру Django-проекта и назначение основных компонентов	ИД-ЗРПК-1				+								Кейс (решение конкретных производственных ситуаций)/Спринт

													4. Основы веб-разработки
назначение URL-адресов и различие абсолютных и относительных путей	ИД-3РПК-1				+								Кейс (решение конкретных производственных ситуаций)/Спринт 4. Основы веб-разработки
принципы работы протокола HTTP и типы запросов	ИД-3РПК-1				+								Кейс (решение конкретных производственных ситуаций)/Спринт 4. Основы веб-разработки
ключевые принципы ООП	ИД-3РПК-1			+									Кейс (решение конкретных производственных ситуаций)/Спринт 3. ООП и другие полезные вещи
объектную модель Python	ИД-3РПК-1			+									Кейс (решение конкретных производственных ситуаций)/Спринт 3. ООП и другие полезные вещи
инструменты анализа и проверки кода Python	ИД-3РПК-1		+										Кейс (решение конкретных производственных ситуаций)/Спринт 2. Настройка рабочего окружения
принципы работы Git и GitHub	ИД-3РПК-1		+										Кейс (решение конкретных производственных ситуаций)/Спринт 2. Настройка рабочего окружения
основы конфигурации среды разработки Python	ИД-3РПК-1		+										Кейс (решение конкретных производственных ситуаций)/Спринт 2. Настройка рабочего окружения
принципы организации циклов for и while	ИД-3РПК-1	+											Кейс (решение конкретных производственных ситуаций)/Спринт 1. Основы Python
основные коллекционные типы и их свойства	ИД-3РПК-1	+											Кейс (решение конкретных производственных ситуаций)/Спринт 1. Основы Python
назначение стандартных модулей Python	ИД-3РПК-1	+											Кейс (решение конкретных производственных ситуаций)/Спринт 1. Основы Python
синтаксис и принципы работы	ИД-3РПК-1	+											Кейс (решение конкретных

условных операторов													производственных ситуаций)/Спринт 1. Основы Python
структуру и механизмы аутентификации и авторизации пользователей в Django	ИД-3РПК-1						+						Кейс (решение конкретных производственных ситуаций)/Спринт 6. Формы в Django и Спринт 7. Тестирование на Python
назначение и структуру тестов в Pytest и Django	ИД-3РПК-1							+					Кейс (решение конкретных производственных ситуаций)/Спринт 6. Формы в Django и Спринт 7. Тестирование на Python
назначение API и различия между Client API и Bot API	ИД-5РПК-1								+				Кейс (решение конкретных производственных ситуаций)/Спринт 8. Работа с API
принципы взаимодействия с внешними сервисами через HTTP-запросы и обработку ошибок	ИД-5РПК-1								+				Кейс (решение конкретных производственных ситуаций)/Спринт 8. Работа с API
архитектуру DRF и принципы работы сериализаторов, вьюсетов и роутеров	ИД-5РПК-1									+			Кейс (решение конкретных производственных ситуаций)/Спринт 9. Django REST Framework
структуру DRF и базовые механизмы конфигурации API	ИД-5РПК-1										+		Кейс (решение конкретных производственных ситуаций)/Спринт 10. Настройка API-сервиса
принципы аутентификации, авторизации и троттлинга в DRF	ИД-5РПК-1										+		Кейс (решение конкретных производственных ситуаций)/Спринт 10. Настройка API-сервиса
принципы взаимодействия REST API с клиентскими приложениями	ИД-5РПК-1										+		Кейс (решение конкретных производственных ситуаций)/Спринт 10. Настройка API-сервиса
принципы контейнеризации и структуру Docker-образов	ИД-5РПК-1											+	Кейс (решение конкретных производственных ситуаций)/Спринт 11. Docker для Python-разработчика
основные команды управления контейнерами Docker	ИД-5РПК-1											+	Кейс (решение конкретных производственных ситуаций)/Спринт

													11. Docker для Python-разработчика
назначение и структуру docker-compose.yml	ИД-5РПК-1											+	Кейс (решение конкретных производственных ситуаций)/Спринт 11. Docker для Python-разработчика
принципы деплоя и сборки CI/CD в Docker и GitHub Actions	ИД-5РПК-1											+	Кейс (решение конкретных производственных ситуаций)/Спринт 11. Docker для Python-разработчика
<b>Уметь:</b>													
анализировать структуру HTTP-запроса и ответа	ИД-3РПК-1				+								Кейс (решение конкретных производственных ситуаций)/Спринт 4. Основы веб-разработки
создавать и настраивать виртуальное окружение	ИД-3РПК-1		+										Кейс (решение конкретных производственных ситуаций)/Спринт 2. Настройка рабочего окружения
создавать и изменять коллекции для хранения и обработки данных	ИД-3РПК-1	+											Кейс (решение конкретных производственных ситуаций)/Спринт 1. Основы Python
регистрировать и настраивать модели в админ-зоне	ИД-3РПК-1					+							Кейс (решение конкретных производственных ситуаций)/Спринт 5. Базы данных и Django ORM
реализовывать регистрацию, вход и ограничение доступа с помощью LoginRequiredMixin и декоратора login_required	ИД-3РПК-1						+						Кейс (решение конкретных производственных ситуаций)/Спринт 6. Формы в Django и Спринт 7. Тестирование на Python
реализовывать выбор логики выполнения программы через if, elif, else	ИД-3РПК-1	+											Кейс (решение конкретных производственных ситуаций)/Спринт 1. Основы Python
проектировать таблицы с внешними ключами	ИД-3РПК-1					+							Кейс (решение конкретных производственных ситуаций)/Спринт 5. Базы данных и Django ORM
проектировать простые типы	ИД-3РПК-1			+									Кейс (решение конкретных производственных ситуаций)/Спринт

													3. ООП и другие полезные вещи
применять числовые и логические типы в вычислениях	ИД-ЗРПК-1	+											Кейс (решение конкретных производственных ситуаций)/Спринт 1. Основы Python
применять циклы и управляющие конструкции для обработки последовательностей	ИД-ЗРПК-1	+											Кейс (решение конкретных производственных ситуаций)/Спринт 1. Основы Python
применять отладку и автоматическую проверку стиля	ИД-ЗРПК-1		+										Кейс (решение конкретных производственных ситуаций)/Спринт 2. Настройка рабочего окружения
применять наследование и полиморфизм для расширения поведения и согласованного интерфейса	ИД-ЗРПК-1			+									Кейс (решение конкретных производственных ситуаций)/Спринт 3. ООП и другие полезные вещи
получать и фильтровать данные из моделей	ИД-ЗРПК-1					+							Кейс (решение конкретных производственных ситуаций)/Спринт 5. Базы данных и Django ORM
писать и запускать тесты для моделей, view-функций и форм	ИД-ЗРПК-1							+					Кейс (решение конкретных производственных ситуаций)/Спринт 6. Формы в Django и Спринт 7. Тестирование на Python
описывать модели и связи в Django ORM	ИД-ЗРПК-1					+							Кейс (решение конкретных производственных ситуаций)/Спринт 5. Базы данных и Django ORM
описывать маршруты с параметрами и связывать их с view	ИД-ЗРПК-1				+								Кейс (решение конкретных производственных ситуаций)/Спринт 4. Основы веб-разработки
использовать шаблонизатор Django для вывода данных и организации верстки	ИД-ЗРПК-1				+								Кейс (решение конкретных производственных ситуаций)/Спринт 4. Основы веб-разработки
использовать импортируемые типы данных и функции внешних модулей	ИД-ЗРПК-1	+											Кейс (решение конкретных производственных ситуаций)/Спринт 1. Основы Python

выполнять базовые операции с репозиторием	ИД-3РПК-1		+									Кейс (решение конкретных производственных ситуаций)/Спринт 2. Настройка рабочего окружения	
формировать и интерпретировать URL-параметры запроса	ИД-3РПК-1				+							Кейс (решение конкретных производственных ситуаций)/Спринт 4. Основы веб-разработки	
создавать и обрабатывать формы, включая ModelForm и кастомные валидаторы	ИД-3РПК-1						+					Кейс (решение конкретных производственных ситуаций)/Спринт 6. Формы в Django и Спринт 7. Тестирование на Python	
создавать и подключать приложения в проекте Django	ИД-3РПК-1				+							Кейс (решение конкретных производственных ситуаций)/Спринт 4. Основы веб-разработки	
подключаться к API и обрабатывать ключи доступа через переменные окружения	ИД-5РПК-1								+			Кейс (решение конкретных производственных ситуаций)/Спринт 8. Работа с API	
выполнять запросы к внешнему API, обрабатывать ответы и исключения	ИД-5РПК-1								+			Кейс (решение конкретных производственных ситуаций)/Спринт 8. Работа с API	
создавать и настраивать API с помощью сериализаторов и ViewSet	ИД-5РПК-1									+		Кейс (решение конкретных производственных ситуаций)/Спринт 9. Django REST Framework	
подключать и настраивать сериализаторы, пагинацию и фильтрацию	ИД-5РПК-1										+	Кейс (решение конкретных производственных ситуаций)/Спринт 10. Настройка API-сервиса	
применять классы разрешений и ограничения скорости запросов	ИД-5РПК-1										+	Кейс (решение конкретных производственных ситуаций)/Спринт 10. Настройка API-сервиса	
отправлять и принимать JSON-данные с помощью HTTP-запросов	ИД-5РПК-1										+	Кейс (решение конкретных производственных ситуаций)/Спринт 10. Настройка API-сервиса	
собирать и запускать контейнеры для Python-проектов	ИД-5РПК-1											+	Кейс (решение конкретных производственных ситуаций)/Спринт

													11. Docker для Python-разработчика
просматривать, запускать и останавливать контейнеры на сервере	ИД-5 <sub>РПК-1</sub>											+	Кейс (решение конкретных производственных ситуаций)/Спринт 11. Docker для Python-разработчика
описывать и запускать мультиконтейнерные приложения	ИД-5 <sub>РПК-1</sub>											+	Кейс (решение конкретных производственных ситуаций)/Спринт 11. Docker для Python-разработчика
публиковать образы на Docker Hub и разворачивать их на удалённом сервере	ИД-5 <sub>РПК-1</sub>											+	Кейс (решение конкретных производственных ситуаций)/Спринт 11. Docker для Python-разработчика



#### **4. КОМПЕТЕНТНОСТНО-ОРИЕНТИРОВАННЫЕ ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ КОНТРОЛЯ ОСВОЕНИЯ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ (ТЕКУЩИЙ КОНТРОЛЬ УСПЕВАЕМОСТИ, ПРОМЕЖУТОЧНАЯ АТТЕСТАЦИЯ ПО ДИСЦИПЛИНЕ)**

##### **4.1. Текущий контроль успеваемости**

###### **5 семестр**

Форма реализации: Компьютерное задание

1. Спринт 1. Основы Python (Кейс (решение конкретных производственных ситуаций))
2. Спринт 2. Настройка рабочего окружения (Кейс (решение конкретных производственных ситуаций))
3. Спринт 3. ООП и другие полезные вещи (Кейс (решение конкретных производственных ситуаций))
4. Спринт 4. Основы веб-разработки (Кейс (решение конкретных производственных ситуаций))
5. Спринт 5. Базы данных и Django ORM (Кейс (решение конкретных производственных ситуаций))
6. Спринт 6. Формы в Django и Спринт 7. Тестирование на Python (Кейс (решение конкретных производственных ситуаций))

###### **6 семестр**

Форма реализации: Компьютерное задание

1. Спринт 10. Настройка API-сервиса (Кейс (решение конкретных производственных ситуаций))
2. Спринт 11. Docker для Python-разработчика (Кейс (решение конкретных производственных ситуаций))
3. Спринт 8. Работа с API (Кейс (решение конкретных производственных ситуаций))
4. Спринт 9. Django REST Framework (Кейс (решение конкретных производственных ситуаций))

Балльно-рейтинговая структура дисциплины является приложением А.

Балльно-рейтинговая структура курсовой работы является приложением Б.

##### **4.2 Промежуточная аттестация по дисциплине**

###### *Зачет с оценкой (Семестр №5)*

Оценка определяется по совокупности результатов текущего контроля успеваемости в соответствии с Положением о балльно-рейтинговой системе для студентов НИУ «МЭИ»

###### *Курсовая работа (КР) (Семестр №6)*

Оценка за курсовую работу определяется в соответствии с Положением о балльно-рейтинговой системе для студентов НИУ «МЭИ»

###### *Зачет с оценкой (Семестр №6)*

Оценка определяется по совокупности результатов текущего контроля успеваемости в соответствии с Положением о балльно-рейтинговой системе для студентов НИУ «МЭИ»

В диплом выставляется оценка за 6 семестр.

**Примечание:** Оценочные материалы по дисциплине приведены в фонде оценочных материалов ОПОП.

## 5. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

### 5.1 Печатные и электронные издания:

1. Меле А.- "Django 4 в примерах", Издательство: "ДМК Пресс", Москва, 2023 - (800 с.)  
<https://e.lanbook.com/book/348113>.

### 5.2 Лицензионное и свободно распространяемое программное обеспечение:

1. Python;
2. Git;
3. Libre Office;
4. ОС Linux.

### 5.3 Интернет-ресурсы, включая профессиональные базы данных и информационно-справочные системы:

1. ЭБС Лань - <https://e.lanbook.com/>
2. ЭБС "Университетская библиотека онлайн" - [http://biblioclub.ru/index.php?page=main\\_ub\\_red](http://biblioclub.ru/index.php?page=main_ub_red)
3. Научная электронная библиотека - <https://elibrary.ru/>
4. База данных журналов издательства Elsevier - <https://www.sciencedirect.com/>
5. Электронные ресурсы издательства Springer - <https://link.springer.com/>
6. База данных Web of Science - <http://webofscience.com/>
7. База данных Scopus - <http://www.scopus.com>

## 6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Тип помещения	Номер аудитории, наименование	Оснащение
Учебные аудитории для проведения лекционных занятий и текущего контроля	Ж-120, Машинный зал ИВЦ	сервер, кондиционер
	3-505, Лекционная аудитория каф. ВМСС	парта, стол преподавателя, стул, мультимедийный проектор, доска маркерная, компьютер персональный, мел, маркер, стилус
Учебные аудитории для проведения практических занятий, КР и КП	Ж-120, Машинный зал ИВЦ	сервер, кондиционер
	3-316, Учебно-исследовательская лаборатория сетевых технологий каф. ВМСС	стол, стул, шкаф, доска меловая
Учебные аудитории для проведения лабораторных занятий	Ж-120, Машинный зал ИВЦ	сервер, кондиционер
	3-316, Учебно-исследовательская лаборатория сетевых технологий каф. ВМСС	стол, стул, шкаф, доска меловая
Учебные аудитории для проведения промежуточной аттестации	Ж-120, Машинный зал ИВЦ	сервер, кондиционер
	3-316, Учебно-исследовательская лаборатория сетевых технологий каф. ВМСС	стол, стул, шкаф, доска меловая
Помещения для	НТБ-303, Лекционная	стол компьютерный, стул, стол

самостоятельной работы	аудитория	письменный, вешалка для одежды, компьютерная сеть с выходом в Интернет, компьютер персональный, принтер, кондиционер
	3-601, Класс самостоятельных занятий каф. ВМСС	
Помещения для консультирования	3-605, Кабинет сотрудников каф. ВМСС	
Помещения для хранения оборудования и учебного инвентаря	3-604, Склад	стол, стол компьютерный, стул, шкаф, компьютерная сеть с выходом в Интернет

## БАЛЛЬНО-РЕЙТИНГОВАЯ СТРУКТУРА ДИСЦИПЛИНЫ

## Web-технологии

(название дисциплины)

## 5 семестр

**Перечень контрольных мероприятий текущего контроля успеваемости по дисциплине:**

- КМ-1 Спринт 1. Основы Python (Кейс (решение конкретных производственных ситуаций))
- КМ-2 Спринт 2. Настройка рабочего окружения (Кейс (решение конкретных производственных ситуаций))
- КМ-3 Спринт 3. ООП и другие полезные вещи (Кейс (решение конкретных производственных ситуаций))
- КМ-4 Спринт 4. Основы веб-разработки (Кейс (решение конкретных производственных ситуаций))
- КМ-5 Спринт 5. Базы данных и Django ORM (Кейс (решение конкретных производственных ситуаций))
- КМ-6 Спринт 6. Формы в Django и Спринт 7. Тестирование на Python (Кейс (решение конкретных производственных ситуаций))

**Вид промежуточной аттестации – Зачет с оценкой.**

Номер раздела	Раздел дисциплины	Индекс КМ:	КМ-1	КМ-2	КМ-3	КМ-4	КМ-5	КМ-6
		Неделя КМ:	5	6	8	10	12	16
1	Основы Python							
1.1	Погружение в Python		+					
1.2	Простые типы данных		+					
1.3	Ветвления		+					
1.4	Импортируемые типы данных		+					
1.5	Коллекции		+					
1.6	Итерации и циклы		+					
1.7	Проект. Приложение «Холодильник»		+					
2	Настройка рабочего окружения							
2.1	Настройка рабочего окружения			+				
2.2	Система контроля и управления версиями			+				
2.3	Требования к коду			+				

2.4	Отладка программ		+				
3	ООП и другие полезные вещи, которые надо знать						
3.1	Объекты и классы			+			
3.2	Знакомство с ООП			+			
3.3	Python: новый уровень			+			
3.4	Расширенные возможности Python			+			
3.5	Проект. Игра «Изгиб Питона»			+			
4	Основы веб-разработки						
4.1	Протокол HTTP				+		
4.2	Сетевые запросы				+		
4.3	Фреймворк Django				+		
4.4	Пути и view-функции				+		
4.5	Вёрстка для бэкендера				+		
4.6	HTML и шаблоны Django				+		
4.7	Проект. Приложение «Блогикум». Часть 1				+		
5	Базы данных и Django ORM						
5.1	Введение в базы данных					+	
5.2	Отношения между таблицами					+	
5.3	Django ORM. Модели					+	
5.4	Админ-зона Django					+	
5.5	Django ORM. Получение информации из БД					+	
5.6	Проект. Приложение «Блогикум». Часть 2. Публикации					+	
6	Формы в Django						
6.1	Работа с формами						+
6.2	Представления: расширенные возможности						+

6.3	Пользователи в Django						+
6.4	Доработка проекта						+
6.5	Проект. Приложение «Блогикум». Часть 3. Доработка						+
7	Тестирование на Python						
7.1	Тестирование						+
7.2	Библиотеки для тестирования						+
7.3	Pytest для Django						+
Вес КМ, %:		16	16	16	16	16	20

### 6 семестр

#### Перечень контрольных мероприятий текущего контроля успеваемости по дисциплине:

- КМ-7 Спринт 8. Работа с API (Кейс (решение конкретных производственных ситуаций))
- КМ-8 Спринт 9. Django REST Framework (Кейс (решение конкретных производственных ситуаций))
- КМ-9 Спринт 10. Настройка API-сервиса (Кейс (решение конкретных производственных ситуаций))
- КМ-10 Спринт 11. Docker для Python-разработчика (Кейс (решение конкретных производственных ситуаций))

**Вид промежуточной аттестации** – Зачет с оценкой.

Номер раздела	Раздел дисциплины	Индекс КМ:	КМ-7	КМ-8	КМ-9	КМ-10
		Неделя КМ:	2	5	8	11
1	Работа с API					
1.1	Что такое API		+			
1.2	Работа с внешними API		+			
2	Django REST Framework					
2.1	Django Rest Framework			+		
2.2	Проектная работа. Проект CRUD для Yatube			+		
3	Настройка API-сервиса					
3.1	Права лимиты запросов				+	
3.2	Взаимодействие фронтенда и бэкенда				+	

3.3	Проект. API для Yatube			+	
4	Docker для Python-разработчика				
4.1	Работа на сервере				+
4.2	Docker: приложение в коробке				+
4.3	Автоматизация. Деплой на сервер				+
4.4	Проект. Контейнеры и CI/CD для Kittygram				+
Вес КМ, %:		25	25	25	25

# БАЛЛЬНО-РЕЙТИНГОВАЯ СТРУКТУРА КУРСОВОГО ПРОЕКТА/РАБОТЫ ПО ДИСЦИПЛИНЕ

Web-технологии

(название дисциплины)

6 семестр

Перечень контрольных мероприятий текущего контроля успеваемости по курсовой работе:

КМ-1 Проверка первой части курсовой работы

КМ-2 Проверка второй части курсовой работы

Вид промежуточной аттестации – защита КР.

Номер раздела	Раздел курсового проекта/курсовой работы	Индекс КМ:	КМ-1	КМ-2
		Неделя КМ:	8	14
1	Выполнение финального проекта		+	+
Вес КМ, %:			20	80