

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Смоленский государственный университет»

Кафедра прикладной математики и информатики

«Утверждаю»  
Проректор по учебно-методической  
работе

\_\_\_\_\_ Ю.А. Устименко  
«23» июня 2022 г.

**Рабочая программа дисциплины**  
**Б1.В.ДВ.06.01 Разработка веб-приложений средствами Django**

Направление подготовки: **01.03.02 Прикладная математика и информатика**  
Направленность (профиль): **Математическое и информационное моделирование**  
Форма обучения: очная  
Курс – 4  
Семестр – 8  
Всего зачетных единиц – 2, часов – 72

Форма отчетности: зачет – 8 семестр

Программу разработал  
кандидат технических наук, доцент Т.А.Самойлова

Одобрена на заседании кафедры  
«16» июня 2022 г., протокол № 10

Заведующий кафедрой \_\_\_\_\_ С.В. Козлов

Смоленск  
2022

## 1. Место дисциплины в структуре ОП

Дисциплина «Разработка веб – приложений средствами Django» относится к дисциплинам по выбору и является вспомогательной для производственной практики студентов на предприятиях. Она изучается в 8 семестре.

При изучении данной дисциплины необходимы компетенции студентов, сформированные при изучении таких дисциплин, как «Информационные системы», «Информационные технологии», «Базы данных», «Защита информации». Рассматриваются возможные подходы к разработке веб - приложений, предназначенных для построения корпоративных информационных систем современных организаций и предприятий, обеспечивающих возможность удаленного доступа к данным и коллективной работы сотрудников. Дисциплина «Разработка веб – приложений средствами Django» обеспечивает проведение технологической практики студентов на предприятиях.

## 2. Планируемые результаты обучения по дисциплине

Компетенция	Индикаторы достижения
<b>ПК-1.</b> Способен осуществлять поиск, анализ, систематизацию научной информации в области прикладной математики и информатики для реализации научно-исследовательских проектов и решения прикладных задач по проектированию и разработке программного обеспечения.	<b>Знает:</b> теоретические основы и технологии организации научно-исследовательской деятельности. <b>Умеет:</b> осуществлять поиск, анализ, систематизацию научной информации в области прикладной математики и информатики для реализации научно-исследовательских проектов и решения прикладных задач по проектированию и разработке программного обеспечения. <b>Владеет:</b> навыками организации и проведения научно-исследовательской деятельности в ходе выполнения профессиональных функций.
<b>ПК-2.</b> Способен анализировать требования и проектировать программное и информационное обеспечение компьютерных сетей, вычислительные модели и модели данных для реализации элементов новых (или известных) программных продуктов.	<b>Знает:</b> возможности существующей программно-технической аппаратуры, современных и перспективных средств разработки программных продуктов, технических средств; методологии разработки программного обеспечения, технологии программирования; методы и средства проектирования программного обеспечения, баз данных, программных интерфейсов; принципы построения архитектуры программного обеспечения и виды архитектуры программного обеспечения, типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения; методики формализации и алгоритмизации поставленных задач. <b>Умеет:</b> проводить анализ требований к программному обеспечению, вырабатывать варианты их реализации, проводить оценку и обоснование вырабатываемых решений; использовать существующие типовые решения и шаблоны проектирования программного обеспечения, применять методы и средства проектирования программного обеспечения,

	<p>структур данных, баз данных, программных интерфейсов; использовать методы и приемы формализации и алгоритмизации задач, применять стандартные алгоритмы, использовать программные средства для графического отображения алгоритмов.</p> <p><b>Владеет:</b> методами анализа возможностей реализации требований к программному обеспечению, оценки времени и трудоемкости их реализации, навыками по проектированию программного обеспечения, баз данных, программных интерфейсов, информационных ресурсов сети Интернет.</p>
<p><b>ПК-3.</b> Способен разрабатывать и отлаживать программный код</p>	<p><b>Знает:</b> методологию разработки программного обеспечения, информационно-коммуникационных систем, баз данных, информационных ресурсов в сети Интернет; технологии программирования; особенности выбранной среды программирования и системы управления базами данных, синтаксис выбранного языка программирования, особенности программирования на нем, стандартные библиотеки языка программирования; компоненты программно-технических архитектур; методы повышения читаемости кода, системы кодировки символов, форматы хранения исходных текстов программ; методы и приемы отладки кода, типы и форматы сообщений об ошибках и состоянии аппаратных средств, современные компиляторы, отладчики оптимизаторы программного кода.</p> <p><b>Умеет:</b> применять выбранные языки программирования для написания программного кода, использовать выбранную среду программирования и средства системы управления базами данных, использовать возможности имеющейся технической и программной архитектуры; структурировать, комментировать, размечать, форматировать программный код в соответствии с требованиями; выявлять ошибки в программном коде, применять методы и приемы его отладки, интерпретировать сообщения об ошибках, предупреждения, применять современные компиляторы, отладчики, оптимизаторы программного кода.</p> <p><b>Владеет:</b> навыками по созданию программного кода в соответствии с техническим заданием, оптимизации программного кода с использованием специализированных программных средств,</p>

форматированию программного кода, анализу, проверке, отладке исходного программного кода.
---

### 3. Содержание дисциплины

В дисциплине «Разработка веб – приложений средствами Django» рассматриваются следующие темы.

1. Основы языка Python. Сферы применения, парадигма. Синтаксис. Числовые, булевы, строковые литералы. Основные операции над простыми типами. Простые управляющие конструкции. Организация кода в модули и пакеты. Коллекции: списки, строки, словари, кортежи, множества. Объектно-ориентированное программирование, элементы функционального программирования. Работа с файлами и базой данных SQLite. Системные библиотеки math, time, os, shutil, sys, random.
2. Архитектура веб-приложений. Модель MVC. Сеть интернет. OSI-модель. Эволюция web-приложений. HTTP-протокол. Запрос, ответ, параметры, GET, POST запросы. Анализ GET и POST запросов с помощью средств разработки в браузерах.
3. Введение в фреймворк Django. Реализация концепции MVC в Django. Структура приложений на Django. Установка и настройка Django. Работа с URL в Django. Базовые модели, view, шаблоны. Класс HttpResponse.
4. Модели в Django. Реляционные модели. ORM-подход. Сравнение подходов. Класс models.Model, реализация основных полей в Django. Использование миграций и fixtures. Работа с формами в Django. Модельные формы.
5. Разработка веб-сервисов в Django. Клиентские приложения для UWP и Android. Эмуляторы мобильных устройств для отладки и тестирования.
6. Шаблоны в Django. Основы языка шаблонов в Django. Подключение шаблонов. Пути к шаблонам. Сбор шаблонов. Основные операции в шаблонах. Наследование шаблонов. Передача параметров в шаблоны. Фильтры.
7. Представления в Django. Представления как реализация контроллера в MVC-модели. Обработка запроса через urls.py, вызов представления. Обработка POST и GET данных. Библиотека компонентов веб-интерфейса Bootstrap. Параметр request. Метод render, класс HttpResponseRedirect. Представления, основанные на классах.
8. Механизмы сессий, авторизации и cookie. Авторизация, аутентификация, регистрация. Принцип работы сессий и куки. Механизмы авторизации в Django. Права доступа. Декораторы. Доступ к сессии и cookie с помощью request.
9. Работа со static и media файлами. Языки HTML, CSS, JS. Подключение static-файлов, сбор static-файлов, пути их поиска. Загрузка файлов, пути к медиа-файлам. Работа с версткой и JS. AJAX-запрос.

### 4. Тематический план

№ п/п	Разделы и темы	Всего часов	Формы занятий			
			лекции	практические занятия	лабораторные занятия	самостоятельная работа
1	Основы языка Python. Примеры проектов с использованием языка.	8	2	–	2	4
2	Архитектура веб- приложений. Модель MVC	8	2	–	2	4
3	Введение в фреймворк Django. Реализация концепции MVC в Django	8	2	–	2	4

4	Модели в Django.	10	2	–	4	4
5	Веб-сервисы Django. Разработка клиентских приложений.	8	2	–	2	4
6	Шаблоны в Django.	6	2	–	2	2
7	Представления в Django. Библиотека Bootstrap.	8	2	–	4	2
8	Механизмы сессий, авторизации и cookie	8	2		4	2
9	Работа со static и media файлами	8	2		4	2
ИТОГО		72	18	–	26	28

## 5. Виды образовательной деятельности

### Занятия лекционного типа

1. Основы языка Python. Сферы применения, парадигма. Синтаксис. Числовые, булевы, строковые литералы. Основные операции над простыми типами. Простые управляющие конструкции. Организация кода в модули и пакеты. Коллекции: списки, строки, словари, кортежи, множества. Объектно-ориентированное программирование, элементы функционального программирования. Работа с файлами и базой данных SQLite. Системные библиотеки math, time, os, shutil, sys, random.
2. Архитектура веб-приложений. Модель MVC. Сеть интернет. OSI-модель. Эволюция web-приложений. HTTP-протокол. Запрос, ответ, параметры, GET, POST запросы. Анализ GET и POST запросов с помощью средств разработки в браузерах.
3. Введение в фреймворк Django. Реализация концепции MVC в Django. Структура приложений на Django. Установка и настройка Django. Работа с URL в Django. Базовые модели, view, шаблоны. Класс HttpResponse.
4. Модели в Django. Реляционные модели. ORM-подход. Сравнение подходов. Класс models.Model, реализация основных полей в Django. Использование миграций и fixtures. Работа с формами в Django. Модельные формы.
5. Разработка веб-сервисов в Django. Клиентские приложения для UWP и Android. Эмуляторы мобильных устройств для отладки и тестирования.
6. Шаблоны в Django. Основы языка шаблонов в Django. Подключение шаблонов. Пути к шаблонам. Сбор шаблонов. Основные операции в шаблонах. Наследование шаблонов. Передача параметров в шаблоны. Фильтры.
7. Представления в Django. Представления как реализация контроллера в MVC-модели. Обработка запроса через urls.py, вызов представления. Обработка POST и GET данных. Библиотека компонентов веб-интерфейса Bootstrap. Параметр request. Метод render, класс HttpResponseRedirect. Представления, основанные на классах.
8. Механизмы сессий, авторизации и cookie. Авторизация, аутентификация, регистрация. Принцип работы сессий и куки. Механизмы авторизации в Django. Права доступа. Декораторы. Доступ к сессии и cookie с помощью request.
9. Работа со static и media файлами. Языки HTML, CSS, JS. Подключение static-файлов, сбор static-файлов, пути их поиска. Загрузка файлов, пути к медиа-файлам. Работа с версткой и JS. AJAX-запрос.

### Лабораторные работы

№1. Установка Django в среде PyCharm и создание проекта.

№2. Разработка простейшего веб-приложения Django в среде PyCharm.

№3. Верстка базового шаблона средствами HTML и CSS для последующей установки в проект Django.

- №4-5. Разработка многостраничного веб- приложения Django в среде PyCharm.
- №6-7. Администрирование веб-приложения и базы данных средствами Django.
- №8-9. Реализация проекта «Блог» в среде PyCharm.
- №10-11. Реализация проекта «Опросы избирателей».
- №12-13. Реализация проекта «Интернет – магазин».

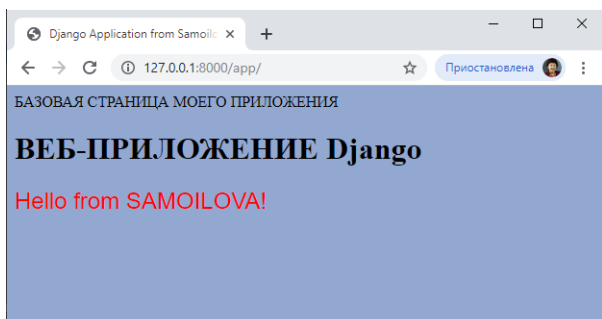
### Задания для лабораторных занятий

#### Лабораторная работа 1 «Установка Django в среде PyCharm Community Edition и создание проекта»

**Задание.** Установите виртуальную среду и фреймворк Django в среде IDE PyCharm Community Edition. Средствами **django-admin** создайте простейший Django-проект

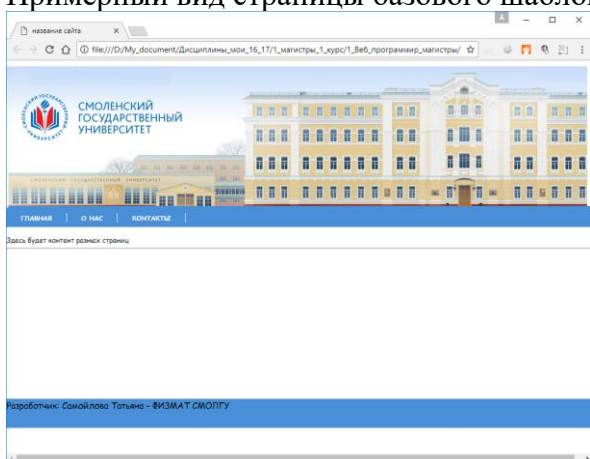
#### Лабораторная работа 2 «Разработка простейшего веб-приложения Django в среде PyCharm Community Edition»

**Задание.** На базе Django-проекта, созданного в предыдущей работе, разработайте веб-приложение, содержащее простейшую страницу. Пример страницы веб-приложения:



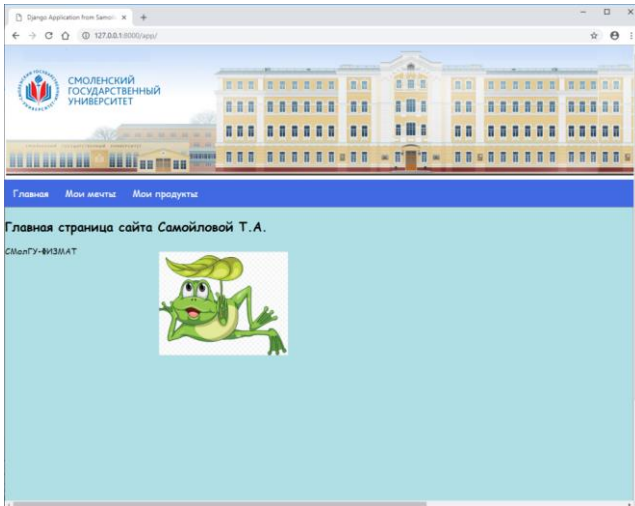
#### Лабораторная работа 3 «Верстка базового шаблона средствами HTML и CSS для последующей установки в проект Django»

**Задание.** Без использования CMS и средств автоматизации разработать базовый шаблон многостраничного веб-сайта. Реализация шаблона выполняется текстовым редактором "Блокнот" и только средствами языков HTML и CSS разметки документов и оформления стилей. Главная страница включает логотип, меню и информацию о разработчике (подвал). Примерный вид страницы базового шаблона (можете создать свой шаблон):

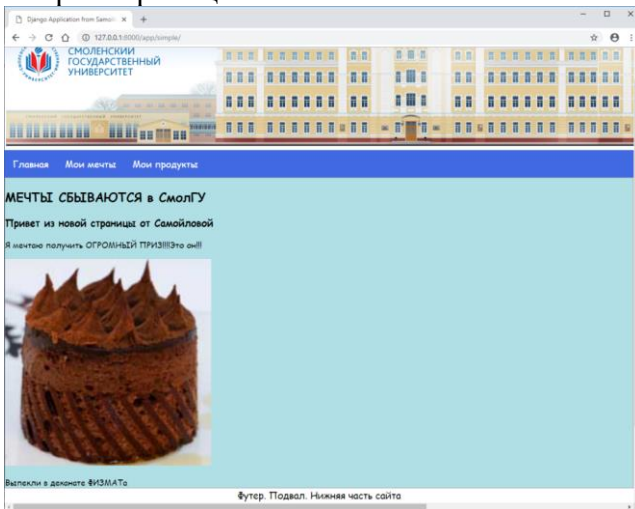


#### Лабораторная работа 4-5 «Разработка многостраничного веб- приложения Django в среде PyCharm Community Edition»

**Задание.** Создайте многостраничное веб-приложение Django средствами IDE PyCharm Community Edition. Пример страниц сайта:



Вторая страница:

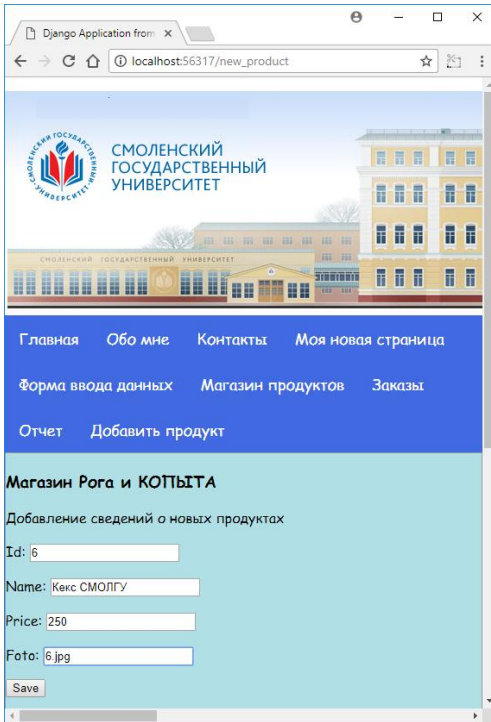


## Лабораторная работа 6-7 " Администрирование веб-приложения и базы данных средствами Django "

**Задание 1 .** Подключите к вашему многостраничному сайту на платформе Django новую страницу вывода информации из таблицы базы данных (Мои продукты). Пример подключения страницы product.html, выполняющей вывод данных таблицы Product базы данных SQLite:

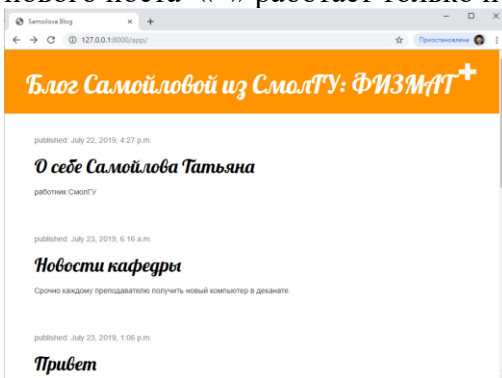


**Задание 2.** Подключите к вашему многостраничному сайту на платформе Django новую страницу ввода информации в таблицу Product:

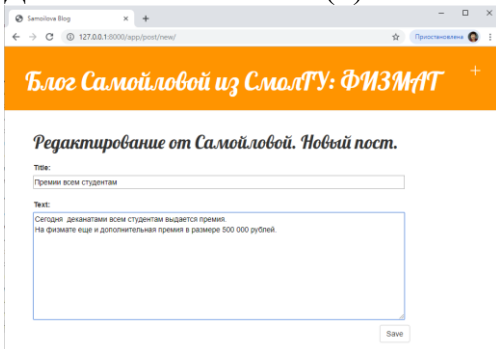


### Лабораторная работа 8-9 «Реализация проекта Блог в среде PyCharm»

**Задание .** Разработать средствами фреймворка Django приложение «Блог», содержащее три страницы: главную со списком тем постов, страницу добавления нового поста и страницу редактирования поста (для авторизованного администратором пользователя). Использовать библиотеку компонентов веб-интерфейса **Bootstrap**. Примерный вид страниц (знак добавления нового поста «+» работает только после авторизации с использованием admin):



Добавляем новый пост (+):



### Лабораторная работа 10-11 «Реализация проекта Опросы избирателей»

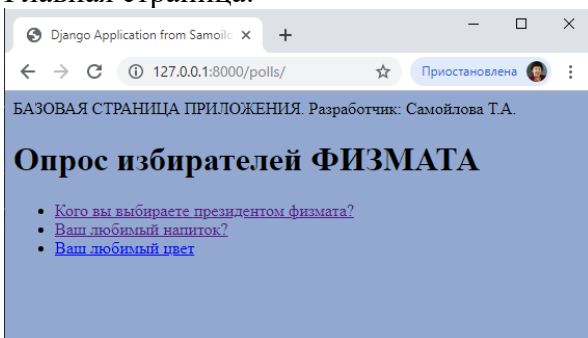
**Задание 1.** Разработать средствами фреймворка Django веб-приложение "Опросы", содержащее три страницы. Главная страница выдает список текстов вопросов с возможностью выбора конкретного для голосования. Вторая страница содержит все варианты ответов на выбранный



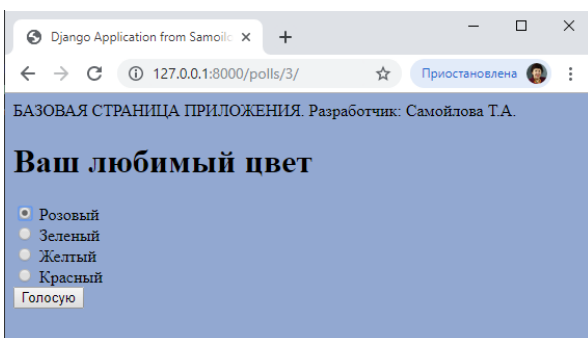
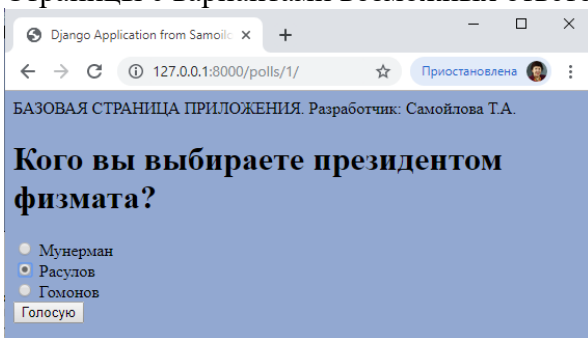
пользователем вопрос с возможностью проголосовать за предложенный вариант ответа. Третья страница содержит результаты подсчета голосов. База данных приложения должна содержать две таблицы: тексты опроса и варианты ответов на каждый вопрос с итогом по каждому варианту.

### Примерный вид страниц сайта

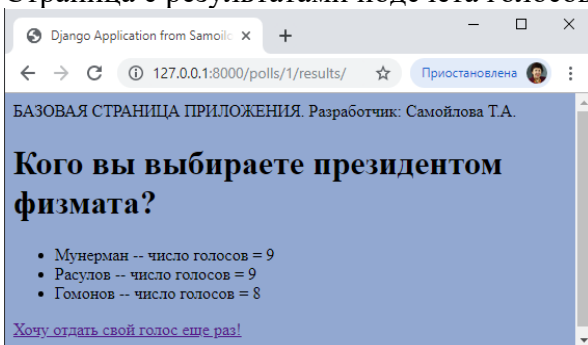
Главная страница:



Страницы с вариантами возможных ответов:



Страница с результатами подсчета голосов:



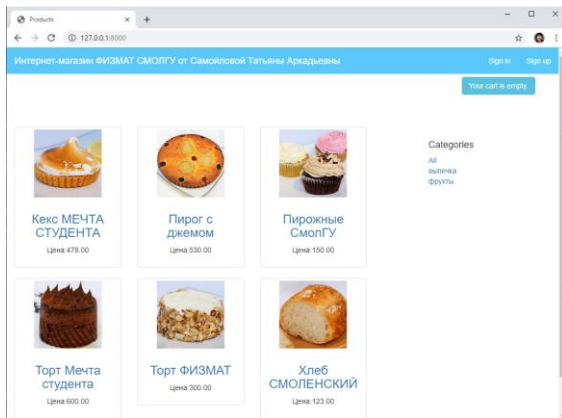
### Лабораторная работа 12-13 «Реализация проекта Интернет – магазин в среде PyCharm»

**Задание.** Разработать средствами фреймворка Django "Интернет – магазин" .

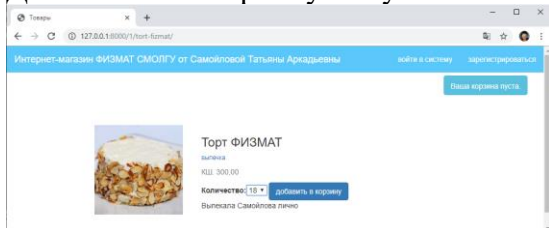
Часть 1. Создание каталога продуктов (таблицы Category, Product)

Часть 2. Реализация корзины для покупок (использование сессий для хранения корзины)

Пример веб - страниц приложения. Каталог продуктов:



#### Добавление в корзину покупок:



Задания для лабораторных работ, размещены в системе дистанционного обучения СмолГУ ([www.moodle.smolgu.ru](http://www.moodle.smolgu.ru)). На занятиях для каждой работы задание предоставляется студентам в электронном виде.

### Самостоятельная работа

Текущая самостоятельная работа студента направлена на углубление и закрепление знаний студентов, развитие практических умений. Она заключается в работе с лекционными материалами, поиске и обзоре литературы и электронных источников, информации по заданным темам курса, опережающей самостоятельной работе, в изучении тем, вынесенных на самостоятельную проработку, подготовке к лабораторным занятиям.

Самостоятельная внеаудиторная работа студентов включает:

- проработку лекционного материала, составление конспекта лекций по темам, вынесенным на самостоятельное изучение;
- выполнение домашних заданий;
- подготовку к защите лабораторных работ.

### Темы для самостоятельного изучения

1. Встроенный интерфейс администрирования в Django.
2. Среды разработки Django - программ.
3. Доступ веб - служб к PostgreSQL.
4. Аjax в Django. Асинхронная отправка данных на сервер и вывод ответа сервера пользователю в браузер.
5. Защита от атак в Django.

Консультирование студентов осуществляется в индивидуальном порядке на занятиях и во внеурочное время. Выполнение самостоятельной работы оценивается по электронным материалам, подготовленным студентами. Результаты деятельности накапливаются в индивидуальных портфолио студентов.

### Учебно-методическое обеспечение для самостоятельной работы

1. Интернет-Университет Информационных Технологий, <http://www.intuit.ru>
2. <https://tutorial.djangogirls.org/ru/>
3. <https://developer.mozilla.org/ru/docs/Learn/Server-side/Django/>

#### 4. Tutorial\_local\_library\_website

### 6. Критерии оценивания результатов освоения дисциплины (модуля) 6.1. Оценочные средства и критерии оценивания для текущей аттестации

#### Теоретические вопросы

Теоретические вопросы по основным темам курса предложены к каждому лабораторному занятию.

1. Основы языка Python. Сферы применения, парадигма. Синтаксис. Числовые, булевы, строковые литералы. Основные операции над простыми типами. Простые управляющие конструкции.
2. Организация кода в модули и пакеты. Коллекции: списки, строки, словари, кортежи, множества.
3. Объектно-ориентированное программирование, элементы функционального программирования. Работа с файлами и базой данных SQLite. Системные библиотеки math, time, os, shutil, sys, random.
4. Архитектура веб-приложений. Модель MVC. Сеть интернет. OSI-модель. Эволюция web-приложений. HTTP-протокол.
5. Запрос, ответ, параметры, GET, POST запросы. Анализ GET и POST запросов с помощью средств разработки в браузерах.
6. Введение в фреймворк Django. Реализация концепции MVC в Django. Структура приложений на Django.
7. Установка и настройка Django. Работа с URL в Django. Базовые модели, view, шаблоны. Класс HttpResponse.
8. Модели в Django. Реляционные модели. ORM-подход. Сравнение подходов.
9. Класс models.Model, реализация основных полей в Django. Использование миграций и fixtures.
10. Работа с формами в Django. Модельные формы.
11. Разработка веб-сервисов в Django. Клиентские приложения для UWP и Android. Эмуляторы мобильных устройств для отладки и тестирования.
12. Шаблоны в Django. Основы языка шаблонов в Django. Подключение шаблонов. Пути к шаблонам. Сбор шаблонов.
13. Основные операции в шаблонах. Наследование шаблонов. Передача параметров в шаблоны. Фильтры.
14. Представления в Django. Представления как реализация контроллера в MVC-модели.
15. Обработка запроса через urls.py, вызов представления. Обработка POST и GET данных. Библиотека компонентов веб-интерфейса Bootstrap.
16. Параметр request. Метод render, класс HttpResponseRedirect. Представления, основанные на классах.
17. Механизмы сессий, авторизации и cookie. Авторизация, аутентификация, регистрация. Принцип работы сессий и куки.
18. Механизмы авторизации в Django. Права доступа. Декораторы. Доступ к сессии и cookie с помощью request.
19. Работа со static и media файлами. Языки HTML, CSS, JS. Подключение static-файлов, сбор static-файлов, пути их поиска.
20. Загрузка файлов, пути к медиа-файлам. Работа с версткой и JS. AJAX-запрос.

#### Критерии оценивания теоретических вопросов

##### 1. Нормы оценивания ответов на теоретические вопросы

№ п/п	Теоретический вопрос	Количество баллов (*)
1	Дан краткий ответ на поставленный вопрос	1 балл
2	Дан развернутый ответ на вопрос с	2 балла

	анализом результатов	
--	----------------------	--

(\*) Возможна градация в 0,25 балла.

2. Шкала оценивания. Оценка «зачтено» за ответы на теоретические вопросы выставляется, если набрано не менее 3 баллов при ответе на три вопроса, в противном случае выставляется «не зачтено».

### Задания для лабораторных занятий

Задачи по темам курса предложены к каждому лабораторному занятию.

Задания для лабораторных и самостоятельной работ, образцы решений основных типовых задач практики также размещены в системе дистанционного обучения СмолГУ ([www.moodle.smolgu.ru](http://www.moodle.smolgu.ru)).

### Критерии оценивания выполнения лабораторных работ

1. Нормы оценивания каждой лабораторной работы:

№п/п	Структурная часть работы	Количество баллов (*)
1	Ответ на теоретические вопросы по теме лабораторной работы	1 балл
2	Демонстрация выполнения конкретного задания, предложенного для самостоятельного решения к лабораторной работе	2 балла

(\*) с возможностью градации до 0,25 балла.

2. Шкала оценивания. Оценка «зачтено» за лабораторную работу выставляется, если набрано не менее 2 баллов, в противном случае за работу выставляется «не зачтено».

## 6.2. Оценочные средства и критерии оценивания для промежуточной аттестации

### Зачетная работа (пример задания)

1. В каталог продуктов созданного веб - приложения добавьте новый продукт и его фото.
2. В таблицу базы данных вашего веб - приложения добавьте новое поле типа int.

### Критерии оценивания зачетной работы

1. Нормы оценивания работы

№ п/п	Структурная часть контрольной работы	Количество баллов (*)
1	Правильно реализован каждый метод решения	1 балл
2	Анализ результатов	2 балла

(\*) Возможна градация в 0,25 балла.

2. Шкала оценивания работы:

п/п	Оценка	Количество баллов
1	Отлично	4,75-5
2	Хорошо	3,75-4,5
3	Удовлетворительно	3-3,5
4	Неудовлетворительно	менее 3

### Критерий получения зачета

Зачет выставляется по результатам работы студента в течение семестра.

Для получения зачета студент должен:

- уметь отвечать на теоретические вопросы, рассмотренные на лекциях;
- уметь решать задачи, предложенные на лабораторных занятиях;
- уметь решать задачи, предложенные на зачетной контрольной работе.

## 7. Перечень основной и дополнительной учебной литературы

### 7.1. Основная литература

1. Чернышев, С. А. Основы программирования на Python : учебное пособие для вузов / С. А. Чернышев. — Москва : Издательство Юрайт, 2022. — 286 с. — (Высшее образование). — ISBN 978-5-534-14350-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/496893>
2. Черткова, Е. А. Программная инженерия. Визуальное моделирование программных систем : учебник для вузов/ Е. А. Черткова. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2022. — 147 с. — (Профессиональное образование). — ISBN 978-5-534-09823-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/493226>.

### 7.2. Дополнительная литература

1. Саммерфилд, М. Программирование на Python 3. Подробное руководство / М.: Символ, 2022. - 608 с.
2. У. Чан, П. Биссекс, Д. Форсье. Django. Разработка веб-приложений на Python / пер. с англ. А. Киселёв. — СПб.: Символ-Плюс, 2020. — 456 с.
3. Головатый А., Каплан-Мосс Дж. Django. Подробное руководство Django. / пер. с англ. — СПб.: Символ-Плюс, 2021. — 560 с.

### 7.3. Перечень ресурсов информационно-телекоммуникационной сети «Интернет»

1. Система дистанционного обучения СмолГУ (moodle.smolgu.ru).
2. Интернет-Университет Информационных Технологий, <http://www.intuit.ru>
3. <https://tutorial.djangogirls.org/ru/>
4. [https://developer.mozilla.org/ru/docs/Learn/Server-side/Django/Tutorial\\_local\\_library\\_website](https://developer.mozilla.org/ru/docs/Learn/Server-side/Django/Tutorial_local_library_website)
5. <https://github.com/ojii/django-shop-example>
6. <https://pocoz.gitbooks.io/django-v-primerah/content/glava-7-sozдание-internet-magazina.html>

## 8. Материально-техническое обеспечение

**Учебная аудитория для проведения занятий лекционного типа**, оснащенная стандартной учебной мебелью, интерактивной доской, мультимедиапроектором, ноутбуком и колонками.

**Учебная аудитория для проведения занятий семинарского типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации** - компьютерная аудитория с выходом в Интернет.

**Помещение для самостоятельной работы** – компьютерный класс с доступом к сети «Интернет» и ЭИОС СмолГУ.

## 9. Программное обеспечение

Kaspersky Endpoint Security для бизнеса Стандартный АО «Лаборатория Касперского», лицензия 1FB6-161215-133553-1-6231.

Microsoft Open License, лицензия 49463448 в составе: Microsoft Windows Professional 7 Russian; Microsoft Office 2010 Russian.

Python 3.9; PyCharm Pro

ДОКУМЕНТ ПОДПИСАН  
ЭЛЕКТРОННОЙ ПОДПИСЬЮ

Сертификат: 03B6A3C600B7ADA9B742A1E041DE7D81B0  
Владелец: Артеменков Михаил Николаевич  
Действителен: с 04.10.2021 до 07.10.2022