

# Введение в Django

**№ урока:** 1 **Курс:** Python&Django

**Средства обучения:** Персональный компьютер с установленным PyCharm , Python и Django.

## Обзор, цель и назначение урока

Целью данного урока является познакомить учащегося с фреймворком Django , с его преимуществами и недостатками. Познакомить с пакетным менеджером pip. Научить устанавливать Django и виртуальное окружение на ПК. Продемонстрировать написание и запуск первого проекта. Познакомить с понятием виртуальное окружение, научить создавать свои модули, а также подключать и использовать их.

## Изучив материал данного занятия, учащийся сможет:

- Установить виртуальное окружение virtualenv.
- Установить Django.
- Понимать, что такое модуль .
- Писать модуль и включать их в проект Django.

## Содержание урока

1. Что такое Django?
2. Сравнительная характеристика известных фреймворков Flask vs Django.
3. Установка виртуального окружения, установка Django.
4. Рассмотрение понятия приложения в проекте Django.
5. Создание и подключение своего первого модуля.

## Резюме

- Django — свободный фреймворк для веб-приложений на языке Python, использующий шаблон проектирования MVC. Проект поддерживается организацией Django Software Foundation. Сайт на Django строится из одного или нескольких приложений, которые рекомендуется делать отчуждаемыми и подключаемыми. Один из основных принципов фреймворка — DRY ( *Don't repeat yourself* ).
- Архитектура Django похожа на «Модель-Представление-Контроллер» (MVC). Контроллер классической модели MVC примерно соответствует уровню, который в Django называется Представление (View), а презентационная логика Представления реализуется в Django уровнем Шаблонов (Template). Из-за этого уровневую архитектуру Django часто называют «Модель-Шаблон-Представление» (MTV)
- Разработчики фреймворка:
  1. Расселл Кейт-Маги
  2. Адриан Головатый
  3. Саймон Виллисон,
  4. Джейкоб Каплан-Мосс
  5. Уилсон Майнер
- Данный фреймворк имеет ряд преимуществ:
  - встроенный интерфейс администратора с возможностью перевода на многие языки;
  - диспетчер URL на основе регулярных выражений;
  - расширяемая система шаблонов с тегами и наследованием;
  - авторизация и аутентификация пользователей;

- система фильтров для построения дополнительных обработчиков запросов;
- система тегов для вывода информации в шаблон;
- Модули(приложения) – это базовые строительные блоки для создания приложений на фреймворке Django.
- Чтобы проект был изолированным от других Django-проектов, нужно создать виртуальное окружение с помощью команды `python -m virtualenv '<name virtualenviroment>'` и активировать его, зайдя в папку Scripts, и вызвать скрипт activate.
- Приложение состоит из основных управляемых файлов: `models.py` , `views.py` , `urls.py`.
- Для подключения модуля необходимо в файле настроек проекта (`settings.py`) в массиве `INSTALLED_APPS` подключить приложение.
- Для создания нового модуля необходимо выполнить команду в консоли :`python manage.py startapp '<name application>'`.
- Для того, чтобы запустить приложение, нужно выполнить команду : `python manage.py runserver`.

### Закрепление материала

- Что такое Django?
- Что такое модуль?
- Зачем нужно виртуальное окружение?
- Каким образом происходит подключение модуля?
- Как происходит запуск проекта?

### Дополнительное задание

Задание

Создать проект Django, создать и подключить приложение в проект, запустить сервер.

### Самостоятельная деятельность учащегося

Задание №1

Выучите основные термины, рассмотренные на уроке.

Задание №2

Создайте модуль `firstapp`, добавьте в него файл `urls.py`, создайте `urlpatterns` типа `url(r'^$',)`, подключите к этому `url` адресу функцию обработчик `hellodjango`, реализуйте функцию `hellodjango`, чтобы она с помощью метода `HttpResponse` выводила в браузер фразу "Hello Django!".

### Рекомендуемые ресурсы

Документация:

<https://docs.djangoproject.com/en/1.11/releases/1.11/>