

Тема 2.4.3. Создание первого проекта на Django

Шаг 1. Создать шаблон проекта Web_1.

В главном меню PyCharm выбрать File/New Project.

В открывшемся окне в строке Location задать имя проекта Web_1.

Нажать Create.

Шаг 2. Добавить в проект библиотеку Django

Выбрать в меню команду File опцию Setting для настройки параметров проекта.

В открывшемся окне выбрать опцию Project Interpreter.

Установить интерпретатор Python и библиотеку Django (в верхней строке поиска набрать Django и щелкнуть по кнопке Install Package).

Библиотека Django будет добавлена в проект.

В папке виртуальной среды проекта Web_1 в папке Scripts появятся два файла:

django-admin.exe

django-admin.py

Шаг 3. Создание проекта Django

Перейти в окно терминала PyCharm (Terminal в строке в нижней части окна).

В окне терминала набрать и выполнить команду:

django-admin startproject hello

В папке проекта Web_1 будет создана папка hello проекта Django.

Структура проекта Django

Наш проект содержит:

Web_1 - папка корневого каталога, контейнер проекта, будет содержать все приложения и файлы управления проектом

Web_1/hello - внешняя папка web-проекта

manage.py - файл для управления из командной строки

db.sqlite3 - база данных SQLite. Создается по умолчанию для хранения данных проекта

Web_1/hello/hello - внутренняя папка web-проекта

hello__init__.py – файл для указания и регистрации пакетов

hello/asgi.py – файл точки входа для ASGI-совместимых вебсерверов

WSGI ([англ. Web Server Gateway Interface](#)) — стандарт взаимодействия между [Python](#)-программой, выполняющейся на стороне сервера, и самим [веб-сервером](#)

[ASGI](#) – духовный наследник WSGI, добавляющий поддержку асинхронный приложений.

hello/setting.py – файл установок и конфигурации текущего проекта Django.

hello/urls.py – таблица контента Django-проекта (URL-декларации)

hello/wsgi.py - файл точки входа для WSGI-совместимых вебсерверов

Шаг 4. Запуск Django-проекта на исполнение

В окне Терминала PyCharm выполнить команды:

- **cd hello** (переход в папку hello)
- **python manage.py runserver** (запуск на компьютере отладочного веб-сервера с адресом <http://127.0.0.1:8000/>)
- по щелчку на ссылке веб-сервера откроется веб-браузер и в него будет загружена страница поздравления с успешной установкой Django.



Установка прошла успешно! Поздравляю!

Эта страница отображается, так как `DEBUG=True` находится в файле параметров и URL-адреса не настроены.



Документация Django
Темы, ссылки и инструкции



Учебник: приложение для опроса
Начало работы с Django



Сообщество Джанго
Подключайтесь, получайте помощь или вносите свой вклад

Программирование отклика приложения

Корректировка файла представлений views.py

по умолчанию файл выглядит так:

```
from django.shortcuts import render
```

```
#Create your views here
```

Функция `render` (предоставлять) используется для обработки входящих запросов.

Изменим на:

```
from django.shortcuts import render
```

```
from django.http import HttpResponse
```

```
#Create your views here
```

```
def index(request):
```

```
    return HttpResponse(«Hello world! Это мой первый проект на Django»)
```

Класс `HttpResponse` предназначен для текстового ответа, отправляемого пользователю.

Корректировка файла `urls.py`

Файл `urls.py` сопоставляет маршруты с представлениями.

по умолчанию файл выглядит так:

```
from django.contrib import admin
from django.urls import path
urlpatterns=[
    path( 'admin/', admin.site.urls),
]
```

Класс `AdminSite` дает возможность работы с интерфейсом администратора сайта.

Функция `path` дает возможность сопоставлять запросы пользователя (маршруты) с функцией (методом) их обработки.

Т.о. маршрут `'admin/'` будет обрабатываться методом `admin.site.urls`, который вернет HTML-страницу администратора.

Изменим файл `urls.py`:

```
from django.contrib import admin
from django.urls import path
from firstapp import views
urlpatterns=[
    path( ' ', views.index, name='home'),
    path( 'admin/', admin.site.urls),
]
```

Маршрут `' '` - это запрос к корневой странице сайта, дополнительно имя для этого маршрута `name='home'`.

При запросе этого маршрута будет вызвана функция `index` из файла `views.py`.

Запустим отладочный веб-сервер:

```
python manage.py runserver
```

В браузере перейдем по адресу <http://127.0.0.1:8000>

Браузер отобразит строку `Hello world! Это мой первый проект на Django.`