Установка Django 4 на Windows

Python уже должен быть установлен.

В командной строке:

pip list - список установленных программ, в глобальном окружении

В Total Commander

Создаем папку Project/ bboard - здесь будет вирт. окружение. Нажать - F7 - создать каталог (новую папку) или в командной строке - $ mkdir bboard

Project/bboard $ python -m venv bboardvenv- создаем вирт. окруж. с именем bboardvenv

cd bboardvenv

pip list - смотрим на папки scripts, include, pip и т.д.

Переходим в scripts - там есть activate.bat - он и будет запускать нашу конфигурацию.

Запуск виртуального окружения

Переходим в папку Project/bboard

В командной строке:

$ . bboardvenv/scripts/activate

Out:

(bboardvenv\_1)$ pip list - видим пакеты pip, setuptools

(bboardvenv\_1)$ python --version - смотрим версию питона в вирт. окруж. Выбираем нужную версию.

$ deactivate bboardvenv

Создание Django проекта с виртуальным окружением через Pycharm

Структура папок

Эти папки уже должны быть созданы, **папка mus\_venv - пустая**

Music\_training # это папка проекта

|

mus\_venv # папка вирт. окружения

В Pycharm - new project (в папке проекта)

- (выбрать адрес вирт. окруж.) Music\_training / mus\_venv

‘OK’

В терминале активизируется само виртуальное окружение.

Установка Django в виртуальное окружение

В Pycharm - экран: кнопка “Open”, переходим в Project/Django, кнопка “OK”

Активизируем Django в этом вирт. окруж.

в терминале:

settings - project - add interpreter - добавить (открывается новое окно). Там:

слева сверху: нажать “Виртуальное окружение”, new новое, (existing существующее), и в строке видим Виртуальное окружение (адрес) <ОК> . В терминале активизируется само виртуальное окружение.

Установка Django

> pip install django==5.0.4

> pip list

> django-admin - смотрим список команд

> django-admin startproject sitewomen (лучше использовать доменное имя сайта)

> cd sitewomen

> python manage.py runserver

==================================================================

Pycharm настройка конфигурации

Это для того, чтобы иметь возможность запуска не через Pycharm/terminal, набирая каждый раз > python manage.py runserver

а через собственную конфигурацию, нажимая знак Треугольник, Паук. При этом появляется возможность делать точки останова в проекте. - это режим отладки.

В Pycharm, в открытом проекте справа сверху Current file - Edit Configurations (add configurations)

открывается окно - (+) - Python у нас Shell Script открывается окно с полями, которые заполняем:

name: sitewomen

script path: Project/sitewomen/manage.py

Python interpreter: Project/django/djvenv/scripts/python.exe

environment variables: PYTHONPATH

working directory: Django/Projects/django/sitewomen

Parameters: runserver

У нас вместо Parameters:

Script text: python manage.py runserver

Если ругается и не установлена environment variables, то:

Используйте виртуальную среду

Когда вы создаете виртуальную среду с помощью **venv** или **virtualenv** , она автоматически устанавливает PYTHONPATH для включения каталога site-packages виртуальной среды. Вот пример:

$ python -m venv my\_virtualenv

$ source my\_virtualenv/bin/activate

ИЛИ

Используйте файл .env

Вы можете использовать файл .env для установки переменных среды, включая PYTHONPATH, для вашего проекта. Сначала установите пакет **python-dotenv,** используя

pip install python-dotenv

Затем создайте файл .env в корневом каталоге вашего проекта и добавьте следующую строку, чтобы установить PYTHONPATH:

PYTHONPATH=/path/to/my/modules/

Наконец, загрузите переменные среды из файла .env в свой скрипт Python –

from dotenv import load\_dotenv

import os

load\_dotenv()

sys.path.insert(0, os.getenv("PYTHONPATH"))

Отключить тестовый сервер. Теперь можно включить собственную конфигурацию.

Ставим точки останова. треугольником или пауком запускам режим отладки, Смотрим окно браузера - там в окне загрузки кружок крутится без остановки - значит точка останова сработала.

Смотрим значение переменных в коде.

Выделяем курсором переменную. Внизу окна Pycharm есть окно Debug. В верхній строке окна Debug есть значок, похожий на калькулятор. Нажимаем. Смотрим.

================================================================

В корне проекта создать файл: README.md

### HOW ACTIVATE VIRTUAL ENVIRONMENT  
### PS E:\Django\_Projects\bboard\bboard> cd ..  
### PS E:\Django\_Projects\bboard> bboardvenv\Scripts\activate  
### (bboardvenv) PS E:\Django\_Projects\bboard>  
  
### HOW TO RUN  
### (bboardvenv) PS E:\Django\_Projects\bboard\bboard>  
### Запуск локального сервера - Окно запуска - конфигурация Django\_Project\_virt\_1

В корне проекта создать файл: requirements-dev.txt

$ pip freeze > requirements-dev.txt

В корне проекта создать файл:

В корне проекта создать файл .gitignore, в нем:

/requirements-dev.txt  
/.idea  
/.gitignore  
/bboardvenv

/bboard/my\_virtualenv

создание репозитория на компьютере

$ git init - создание репозитория на компьютере

$ git status

$ git add . - добавить все

$ git rm --cached # --cached - удалить только из памяти

$ git commit -am “First commit”

Загрузить проект с Pycharm на GitHub

**Вариант 1: Когда на GitHub нет репозитория**

**на GitHub создать репозиторий bboard**

В терминале в папке проекта:

(bboardvenv) PS E:\Django\_Projects\bboard>

$ git init - создание репозитория на компьютере

$ git add . - добавить все

$ git commit -am “first commit”

$ git branch -M main

$ git branch -a - показать ветки

$ git remote add origin https://github.com/Evchap/bboard

$ git push -f origin main

если ругается:

$ git pull - подтянуть изменения в ветках

$ git push -f origin main

У НАС

ругается на git remote add origin <https://github.com/Evchap/bboard>

# GitHub "fatal: remote origin already exists" - источник уже существует

Сделал через Pycharm:  
Git - GitHub - Share project no GitHub

создался репозиторий на ГитХабе

Проверки:

$ git --v

$ git status

$ git branch -a - показать ветки

$ cat .git/config - конфигурация. Смотреть, репозиторий на Гите

**Вариант 2: Когда на GitHub есть репозиторий**

В Git Bash:

$ git remote add origin https://github.com/Evchap/bboard

$ git branch -M main

$ git push -u origin main - эта команда может не пройти и запросить token

Token

На своей странице в GitHub справа вверху - выпадающий список (там где мои репозитории) внизу списка - настройки. В новом окне слева в самом низу - настройки разработчика Developer settings - Personal access token - generate new token …

Переходим в терминал Pycharm:

$ git push -u origin main

логин: ввести

пароль: ввести токен

Загрузить проект с Pycharm на GitHub

**Вариант 3: через** Pycharm

В папке проекта Git bush here

git log - просмотр веток в Гите в виде дерева