

### **How to install Django:-**

Python should be available in machine.

```
C:\Windows\System32>py --version
```

Python 3.10.5

-->Install Django by using pip

```
C:\Windows\System32>pip install django==4.1
```

### **To check the django version:**

```
C:\Windows\System32>python -m django --version
```

4.0

### **IDE:-**

ATOM/Pycharm/Vscode

### **Django Project vs Django Application:**

-->A django project is a collection of applications and configurations which forms a full web application.

### **Ex:**

Banking Project

-->A django application is responsible to perform a particular task in our entire web application.

Ex: loan app, registration app, polling app

Project: Several Applications + Configuration Information

### **Note:-**

1.The django applications can be plugged into other projects. i.e these are re-usable.(Pluggable Django Applications)

2.Without existing Django project there is no chance of existing Django application. Before creating any application first we have to create project.

## How to create Django project?

```
D:\>mkdir Django_19MAR_7PM
```

```
D:\>cd Django_19MAR_7PM
```

```
D:\Django_19MAR_7PM>django-admin startproject firstproject
```

or

```
D:\Django_19MAR_7PM>py -m django startproject firstproject
```

```
D:\Django_19MAR_7PM>tree /f
```

```
D:.
```

```
├──firstproject
│   |  manage.py
│   |
│   └──firstproject
│       asgi.py
│       settings.py
│       urls.py
│       wsgi.py
│       __init__.py
```

### **1. \_\_init\_\_.py:**

It is a blank python script. Because of this special name, Django treated this folder as python package.

### **2. settings.py:**

In this file we have to specify all our project settings and configurations like installed apps, middleware configuration, database configuration etc.....

### **3.urls.py:**

Here we have to store all our url-patterns of our project.

For every view(web page), we have to define separate url-pattern. End user can use url-patterns to access our web pages.

#### **4.wsgi.py:**

wsgi-->web server gateway interface

We can use this file while developing our application in production on online server.

#### **5.asgi.py:**

asgi--> Asynchronous server gateway interface

#### **6.manage.py:**

The most commonly used python script is manage.py

It is a commandline utility to interact with Django project in various ways like to run development server, run tests, create migrations etc.....

#### **How to run Django development server:**

-->We have to move to manage.py file location and we have to execute.

D:\Django\_19MAR\_7PM\firstproject>py manage.py runserver

Now server will be started.

#### **How to send the first request:**

-->Open browser and send request:

http://127.0.0.1:8000/

**Ex:**

a = [0,1,2,3]

for a[-1] in a:

print(a[-1])