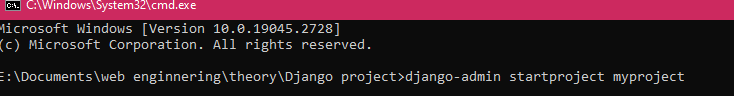
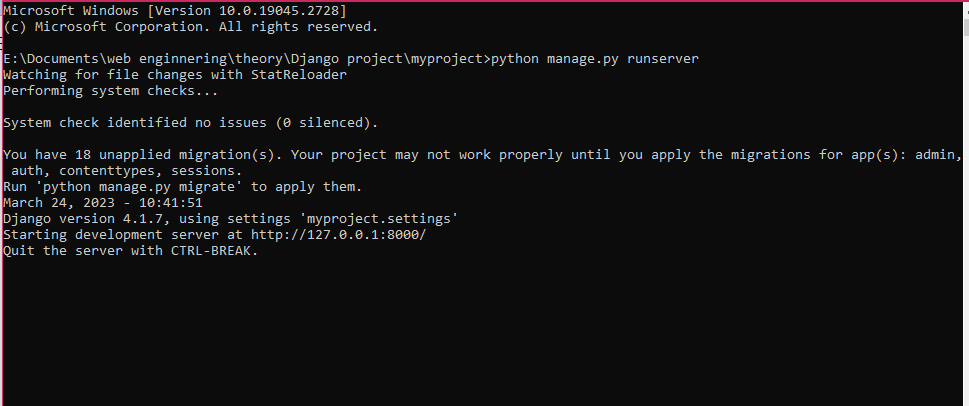
Python version 🡪 Python 3.11.2

Django version 🡪-4.1.7

Create project:



Manage command:



Command run on google local host :

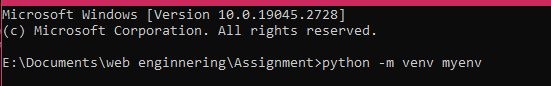
<http://127.0.0.1:8000/>

Assignment no1:

First create a folder then open the cmd on this folder location and add command

myenv

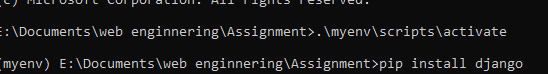
* py -m venv (name of env)



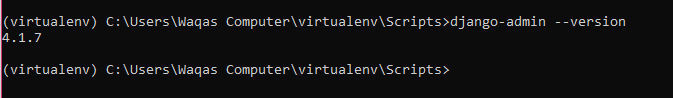
for activate this env go to the virtualenv file and then go to scripts file and run command:

* activate.bat

after activation install django



Django version:



(note) For deactivate the virtual env simple cpmmand is

* deactivate

1. **Now create new project by using Django:**



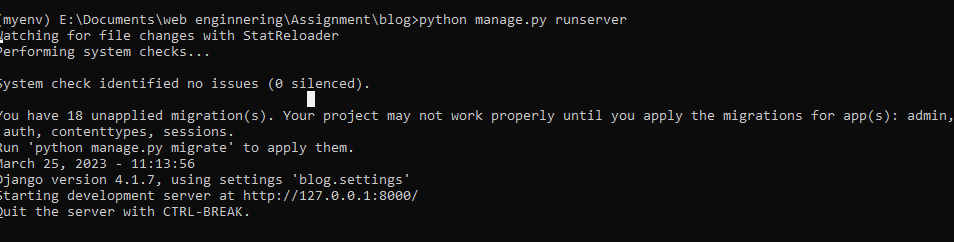
(Note) If open VS code in project simple command is

* code .

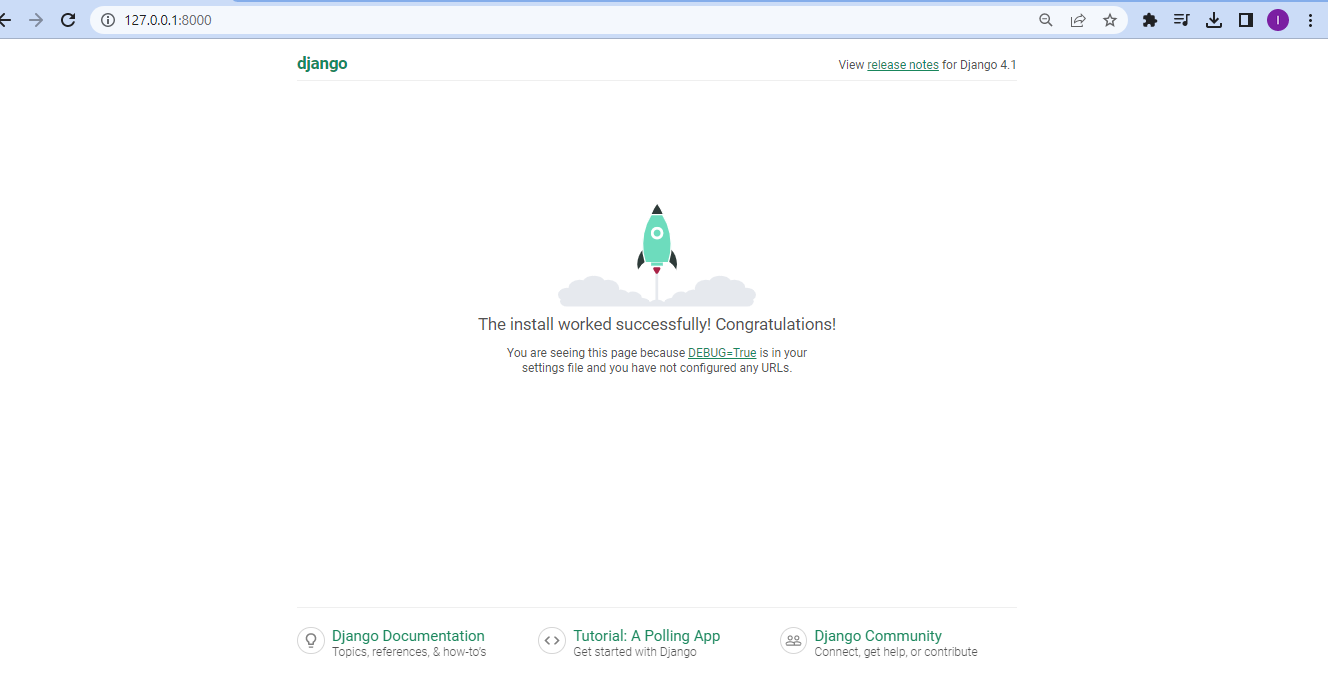
now create link for google local host (development server)

command: python manage.py runserver

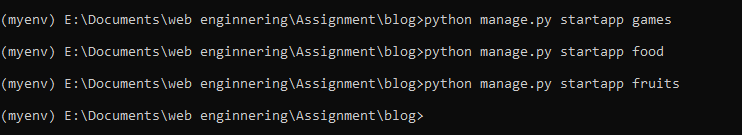
* http://127.0.0.1:8000/



Output (front page):

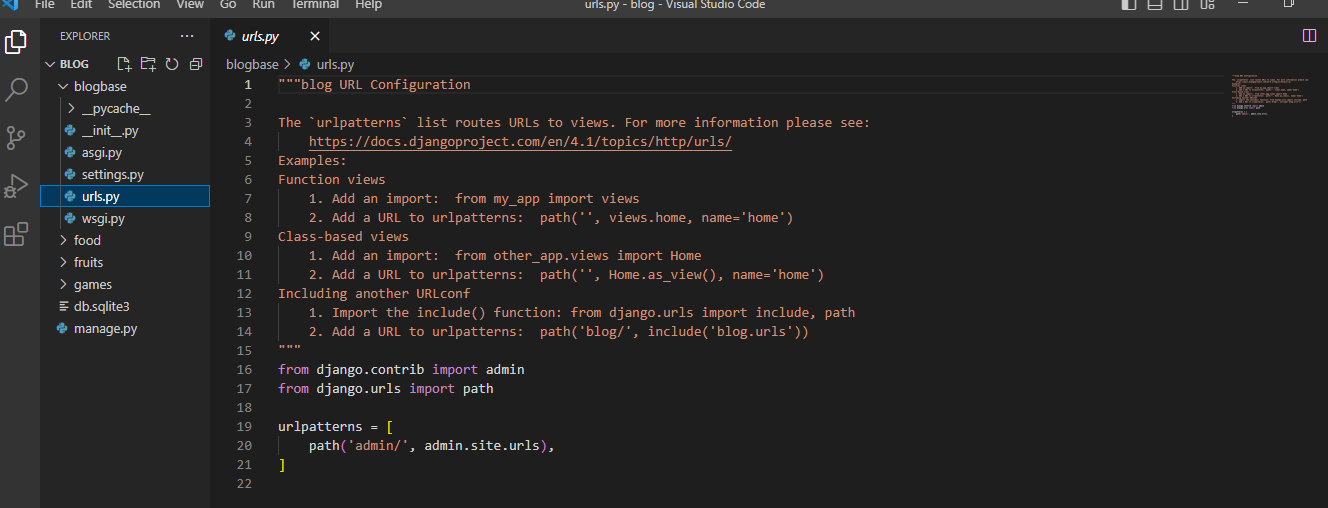


1. **Now create 3 apps in it:**
2. Games
3. Food
4. Fruits



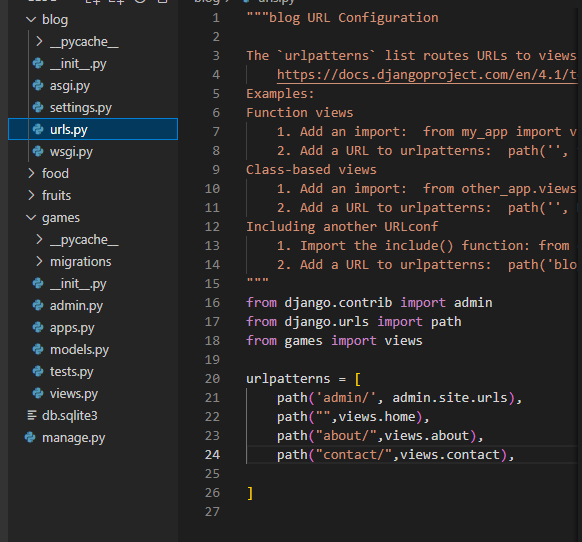
1. **View files:**

Now we chanage the front page by using urls of the baseapp (blogbase).

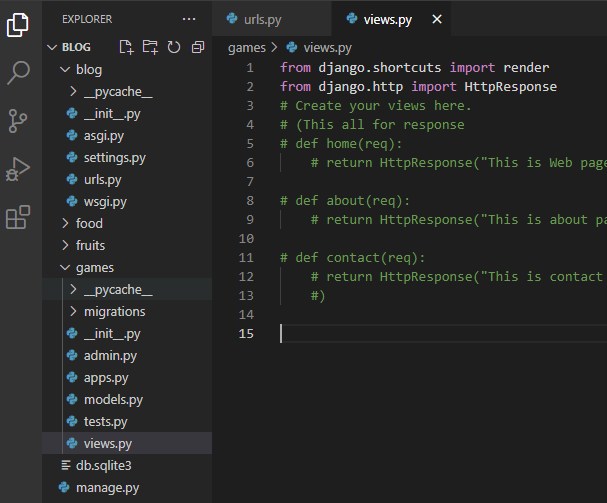


* Now add rotmers in the code.

Base files urls change such as:



From app files changes occur in views files such as:



1. **Now add templates in files:**

Create file in app(games ,fruits, food) name **template.**

After this in templates create html file **(home.html)**

**Template formate:**

**Import render function in view file for template**

If we add simple data in page we use dictionary in views files such as

**View files:**

Dictionary formate

def home(req):

    return render(req,"home.html",{'data':"lorem"}) #dictionary

if we add some thing else simple add same format but change the key index

def home(req):

    return render(req,"home.html",{'data':"lorem",'age':25})

#for passing list in it

def home(req):

    return render(req,"home.html",{'data': [1,2,3,4,5]})

**templates html files:**

!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

</head>

<body>

    <h2>This is Home Page.</h2>

#for simple data entry use this format for displaying values

    {{data}}

    {{age}}

#for calling list data different ways

 {% for i in data%}

   {{i}}

   {% endfor %}

#Also show in order list such as

<ul>

    {% for i in data%}

    <li> {{i}} </li>

    {% endfor %}

</ul>

#for even numbers print

 <ul>

    {% for i in data%}

    {% if i|divibleby:|2  %}

    <li> {{i}} </li>

    {% endfor %}

   </ul>

</body>

</html>

**For default tables in sqlite3:**

Simply use command: python manage.py migrate

If you want to change tables on other files use **migration** keyword.

* **Create super user:**

PS E:\Documents\web enginnering\Assignment> cd blog

PS E:\Documents\web enginnering\Assignment\blog> python manage.py createsuperuser

Username (leave blank to use 'waqascomputer'): admin

Email address: iqra123@gmail.com

Password: iqra123

Password (again): iqra123

Superuser created successfully.

* **Dynamic urls:**

**Import view file in urls**

**3 types of data type used in urls:**

* **Int**
* **Str**
* **Slug (hello-in-world)**

**Urls file:**

**Dynamic url**

**/<datatype:courseid>,**  Function name used in view file



**Note: If you don’t know the data type then simply add the parameter name it will show all types of data.**



**View file:**

**Import render function in view file first**

Function calling

Pass parameter that used in url after colon sign



**How to pass Data from Django view to template:**

**Create models(tables) in apps:**

Go to the models file of the app and create classes such as:

class games(models.Model):

#field name

    games\_icon=models.CharField(max\_length=50)

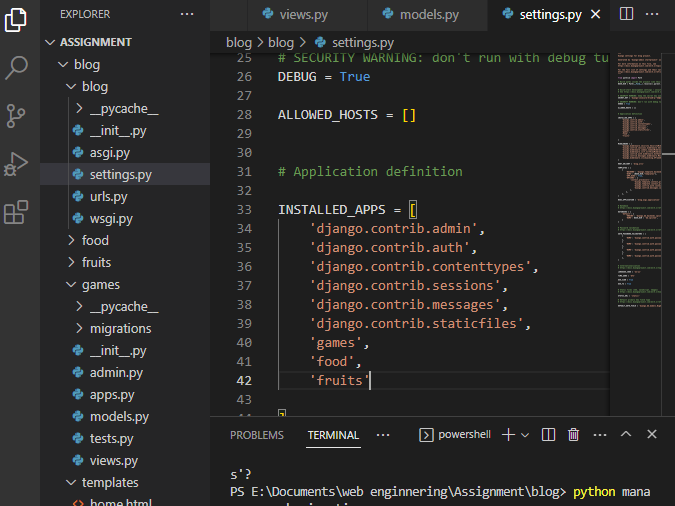
    games\_tiltle=models.CharField(max\_length=50)

    games\_description=models.TextField()

then go project setting and add your app name in installed apps:

after adding this information run command:

* python manage.py migrations



For converting model into tables we use command:

* Python manage.py migrate

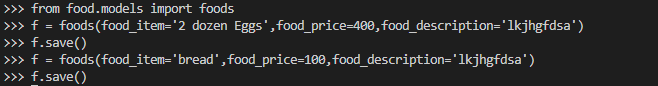
**Add data in table by use of python:**

* Off the server by pressing ctrl c
* After this run these commands:

py manage.py shell 🡪 for open python shell

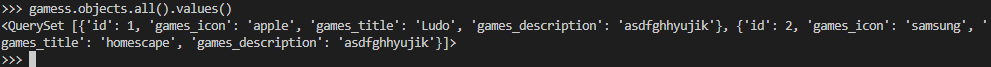
* then add your table in query and run server

>>> from games.models import games

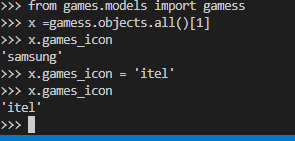


**for view data use this command:**





**Update data in table:**



**2 task:**

**Username (leave blank to use 'waqascomputer'): iqra**

**Email address: myblog123@gmail.com**

**Password: 123**

**Password (again): 123**