Introduction to Django

First steps to create Django project

Reference: https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django

Reference: https://docs.djangoproject.com/en/4.1/

Topic covered

- Framework – an introduction
- Commonly used Python frameworks
- Key features of django framework
- Client side documents (html, css, javascript)
- Create django project
- File structure in django project
- MVT architecture

What is Framework?

- Framework is set of structures (e.g. file/directory and storage structure), procedures and set of rules to implement solution (e.g. web site implementation for Django)
- Existing framework structure of files, directories and models can be efficiently and effectively used and implemented
- Django is Python-based web framework popular among large and popular web sites
- For client side, client application that supports HTML (Hyper Text Markup Language), CSS (Cascade Style Sheet) and JavaScript application development program. Browser supports all these three client-side tools
- In a Web Framework, there are standard common databases system for authentication and data storage, good framework support encryption as well and web framework also supports web services
- In cloud environment, the framework deployment is scalable, portable and highly available
- Web Framework supports clean, quick design and implementation
- For web application – we require web server, authentication data server and database system on the server side

Global users of Python/Django framework

Quick search on web results in the following companies using Django application framework for web services:

- Instagram
- National Geographic
- Mozilla
- Spotify
- Pinterest
- Disqus
- Bitbucket
- Eventbrite
- YouTube
- Dropbox

What are most common web-frameworks used for Python?

- AIOHTTP framework (Asynchronous framework) relies on Python 3.5+
- Bottle is Micro-framework which creates a single file for every application, as these frameworks were developed to develop APIs. Bottle is simple to learn web frameworks and prototyping
- CherryPy – It is Micro-framework, an open-source, object-oriented Python framework that follows minimalistic approach. CherrPy has its own embedded multi-threaded web server and runs on any OS with support for Python. Such app can be deployed anywhere ordinary Python app is able to run
- Flask – Micro-framework available under BSD license, it requires Jinja2 template and Werkzeug WSGI toolkit
- Falcon – Aimed at rapidly building web APIs. It is another widely used Python framework

Django Web Framework and Key highlights

Django is one of the top 10 Full-Stack Python web frameworks with large number of built-in features rather than offering them individual libraries. Django has its ORM for mapping objects to database tables

Key highlights:

✓A plethora of ready-to-use libraries

✓Authentication support

✓Database schema migrations

✓Object-relational mapper (ORM)

✓Support for web servers

✓Template engine

✓URL routing

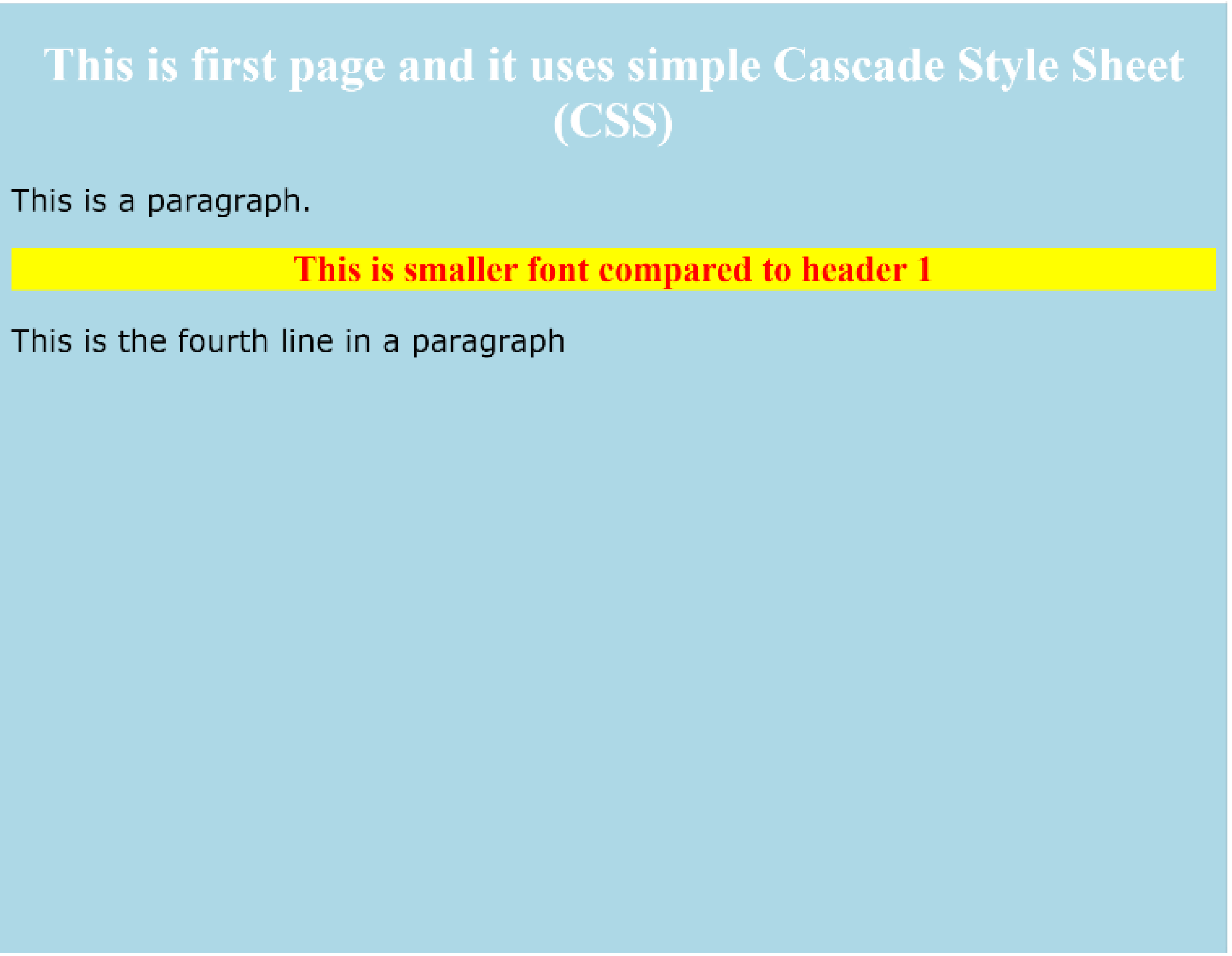
Django Framework (Server side features)

- Python code – requires use of Python libraries
- Phyton functions
- Classes
- Authentication services
- Databases (SQL) – SQLite File
  - Stores data
  - Other options are there as well (e.g. POSTGRES)

Client – Side or Front – End web application

- On the client side
  - Text/Images/Data/Information rendering is done through HTML
  - Consistency in rendering pages (e.g. colour, bold, Italics, font style and size) are rendered by Cascade Style Sheet (CSS)
  - Client side functionality, some level of client-side dynamism is achieved using JavaScript (e.g. text fields must have data in them, password must match with multiple fields)
  - Combination of all three creates front-end or client side application

Client-side page



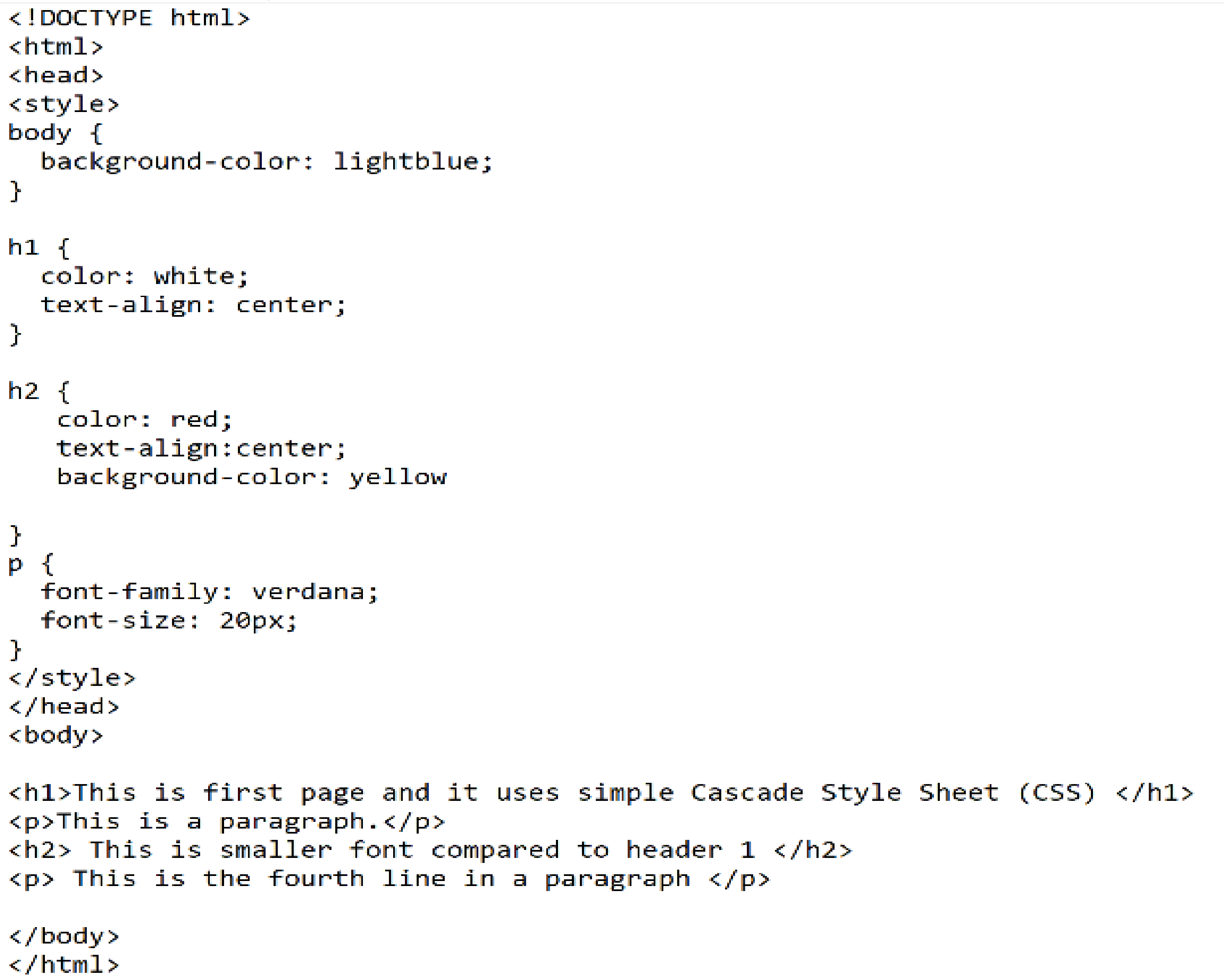
This is first page and it uses simple Cascade Style Sheet (CSS)

This is a paragraph.

This is smaller font compared to header 1

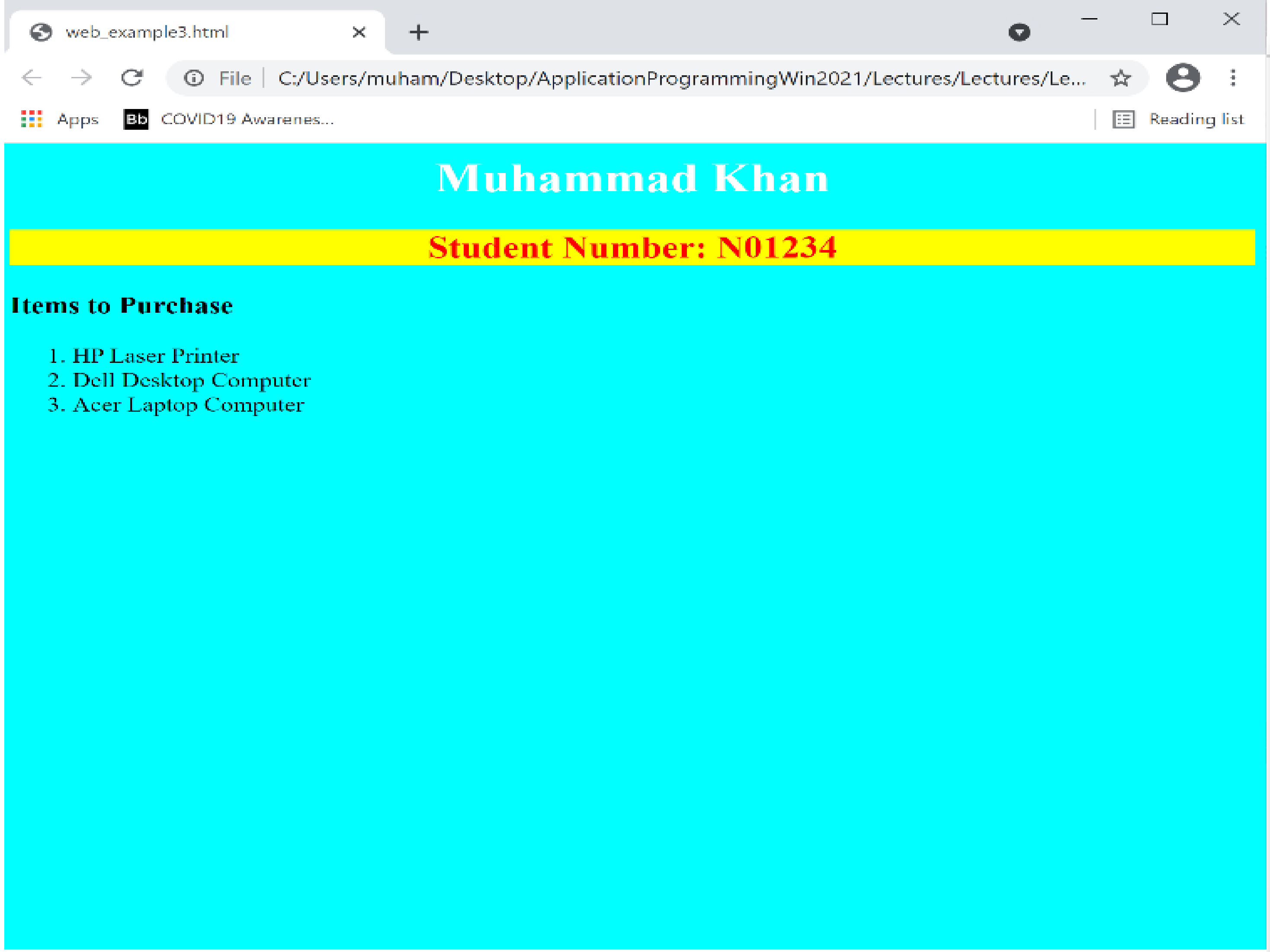
This is the fourth line in a paragraph

Script for Client-side page



```
<!DOCTYPE html>
<html>
<head>
<style>
body {
  background-color: lightblue;
}

h1 {
  color: white;
  text-align: center;
}

h2 {
   color: red;
   text-align:center;
   background-color: yellow

}
p {
  font-family: verdana;
  font-size: 20px;
}
</style>
</head>
<body>

<h1>This is first page and it uses simple Cascade Style Sheet (CSS) </h1>
<p>This is a paragraph.</p>
<h2> This is smaller font compared to header 1 </h2>
<p> This is the fourth line in a paragraph </p>

</body>
</html>
```

Rendering web page with stylesheet link

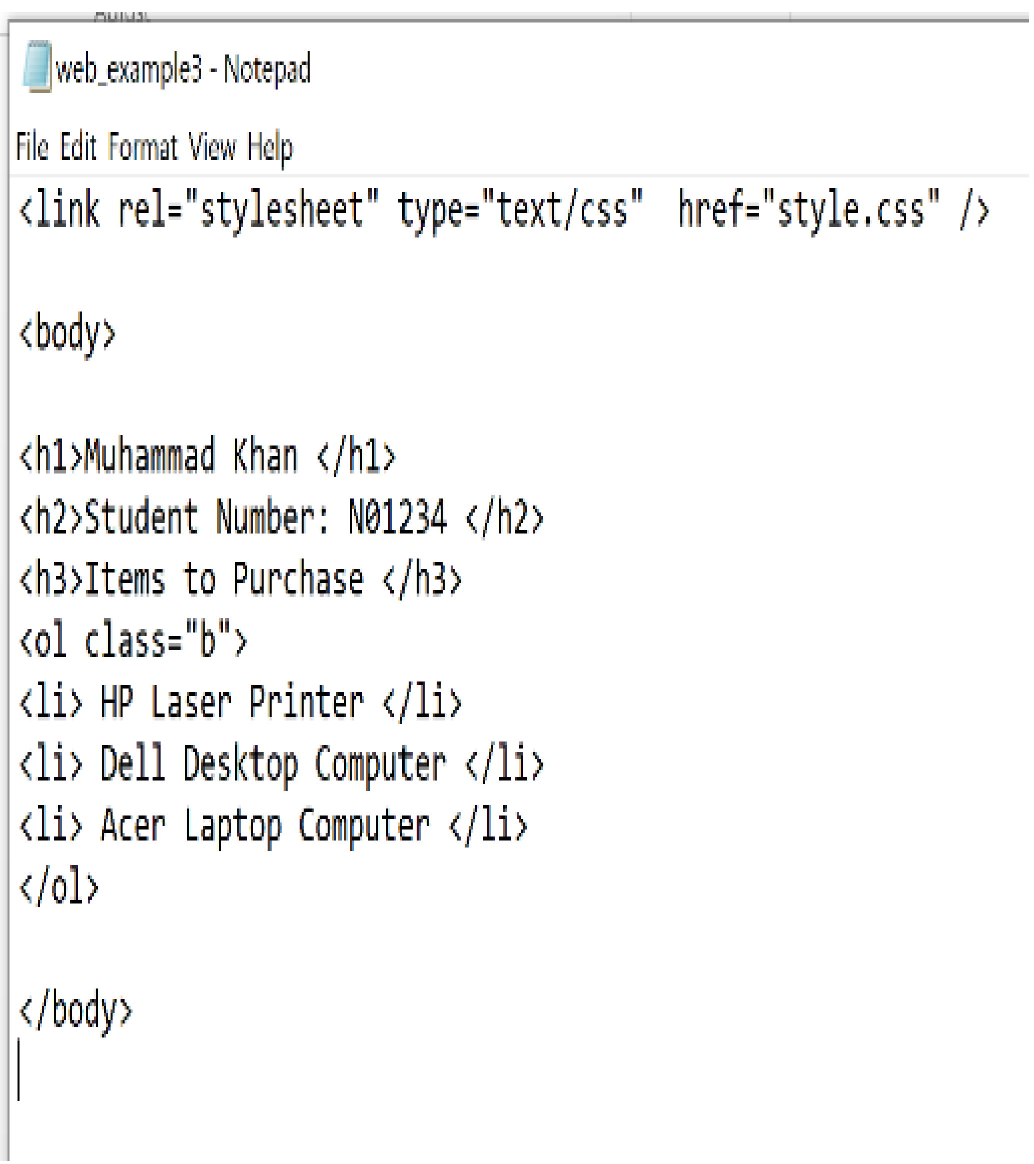

web_example3.html
File
C:/Users/muham/Desktop/ApplicationProgrammingWin2021/Lectures/Lectures/Le...
Apps
COVID19 Awarenes...
Reading list

Muhammad Khan

Student Number: N01234

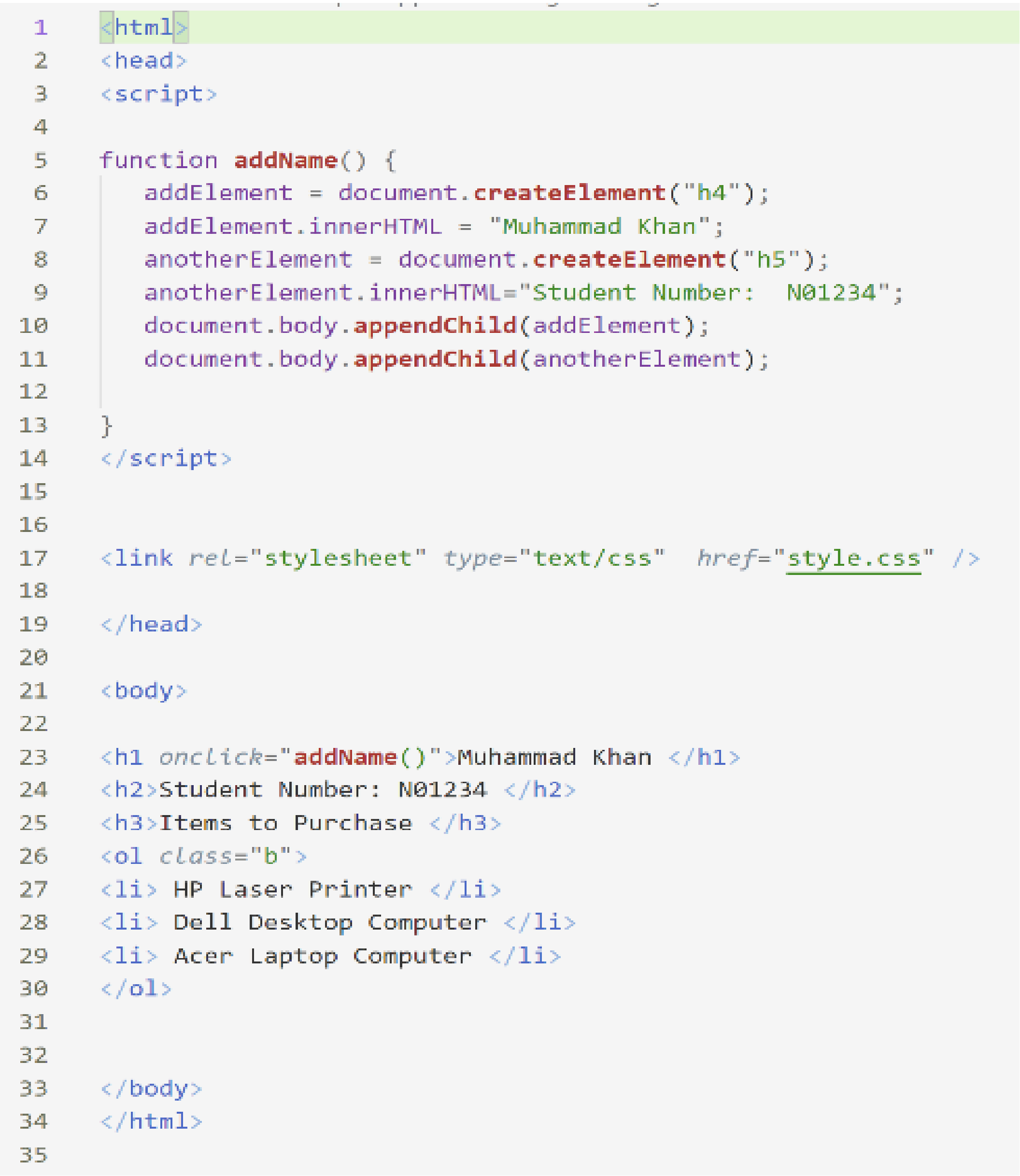
Items to Purchase

1. HP Laser Printer
2. Dell Desktop Computer
3. Acer Laptop Computer

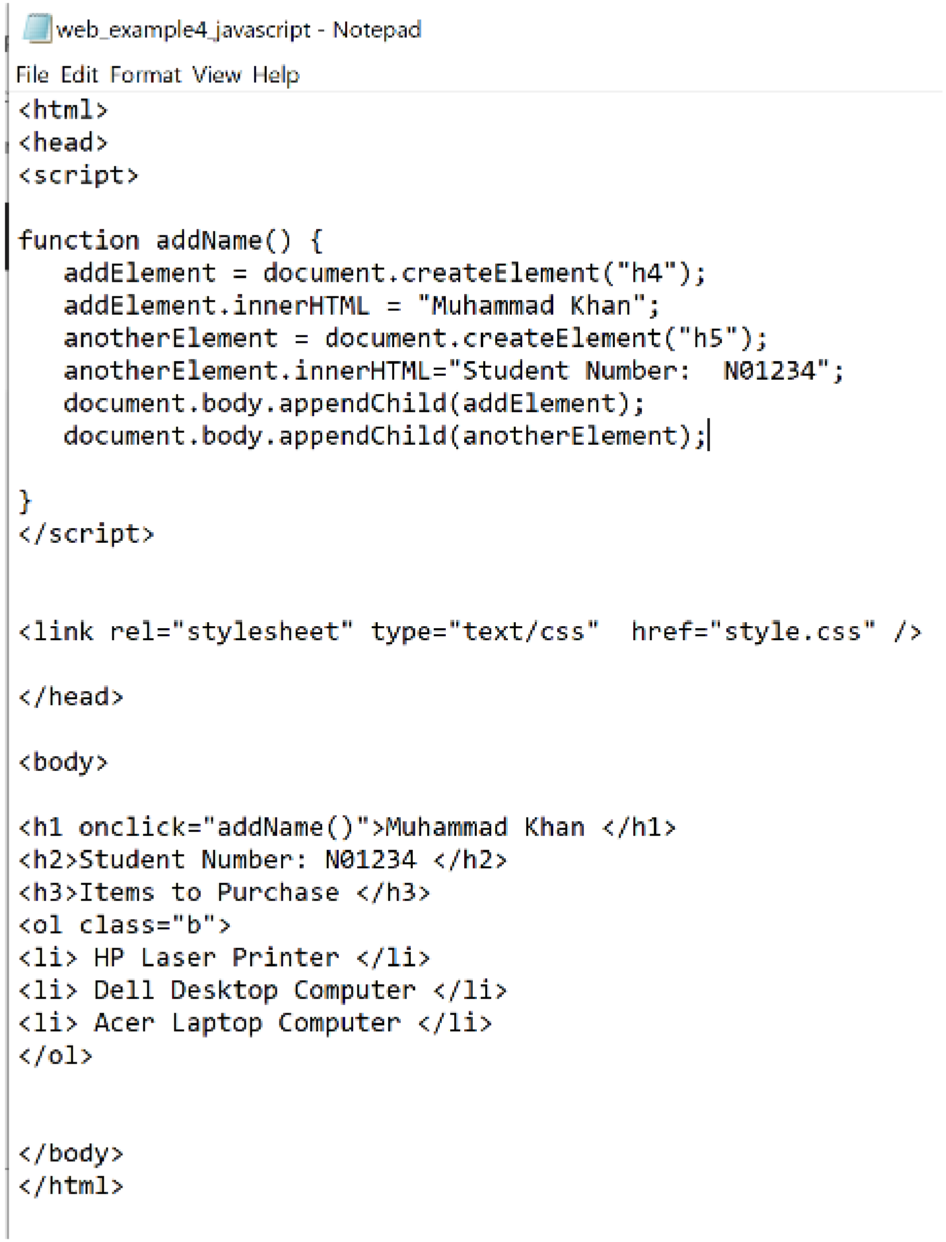


```
<link rel="stylesheet" type="text/css"  href="style.css" />

<body>

<h1>Muhammad Khan </h1>
<h2>Student Number: N01234 </h2>
<h3>Items to Purchase </h3>
<ol class="b">
<li> HP Laser Printer </li>
<li> Dell Desktop Computer </li>
<li> Acer Laptop Computer </li>
</ol>

</body>
```

JavaScript – an introduction to introduce client-side dynamism

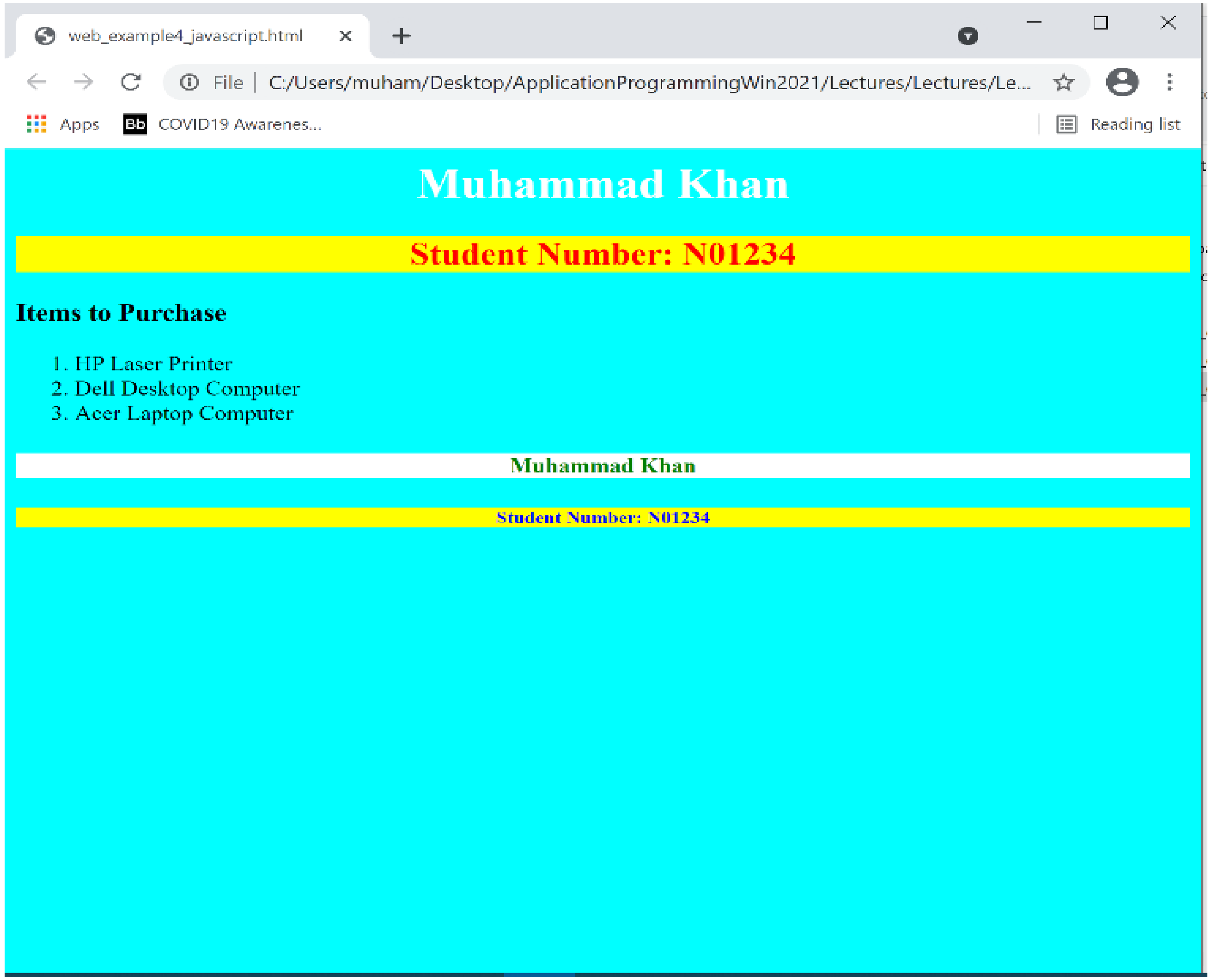


```
<html>
<head>
<script>

function addName() {
    addElement = document.createElement("h4");
    addElement.innerHTML = "Muhammad Khan";
    anotherElement = document.createElement("h5");
    anotherElement.innerHTML="Student Number:  N01234";
    document.body.appendChild(addElement);
    document.body.appendChild(anotherElement);

}
</script>


<link rel="stylesheet" type="text/css"  href="style.css" />

</head>

<body>

<h1 onclick="addName()">Muhammad Khan </h1>
<h2>Student Number: N01234 </h2>
<h3>Items to Purchase </h3>
<ol class="b">
<li> HP Laser Printer </li>
<li> Dell Desktop Computer </li>
<li> Acer Laptop Computer </li>
</ol>


</body>
</html>
```



```
<html>
<head>
<script>

function addName() {
   addElement = document.createElement("h4");
   addElement.innerHTML = "Muhammad Khan";
   anotherElement = document.createElement("h5");
   anotherElement.innerHTML="Student Number:  N01234";
   document.body.appendChild(addElement);
   document.body.appendChild(anotherElement);

}
</script>


<link rel="stylesheet" type="text/css"  href="style.css" />

</head>

<body>

<h1 onclick="addName()">Muhammad Khan </h1>
<h2>Student Number: N01234 </h2>
<h3>Items to Purchase </h3>
<ol class="b">
<li> HP Laser Printer </li>
<li> Dell Desktop Computer </li>
<li> Acer Laptop Computer </li>
</ol>


</body>
</html>
```

Web Rendering – with embedded JavaScript function


web_example4_javascript.html
File
C:/Users/muham/Desktop/ApplicationProgrammingWin2021/Lectures/Lectures/Le...
Apps
COVID19 Awarenes...
Reading list

Muhammad Khan

Student Number: N01234

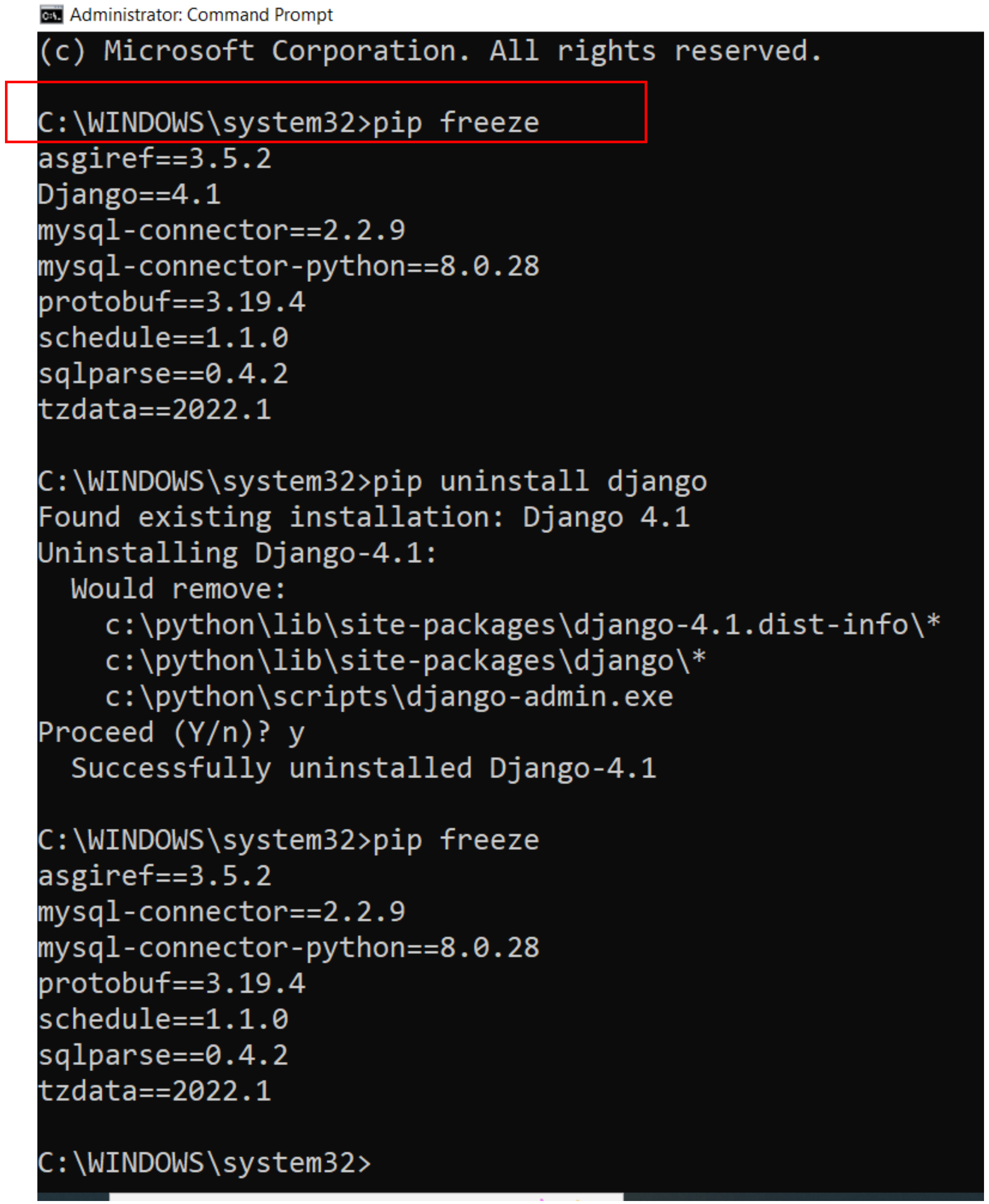
Items to Purchase

1. HP Laser Printer
2. Dell Desktop Computer
3. Acer Laptop Computer

Muhammad Khan

Student Number: N01234

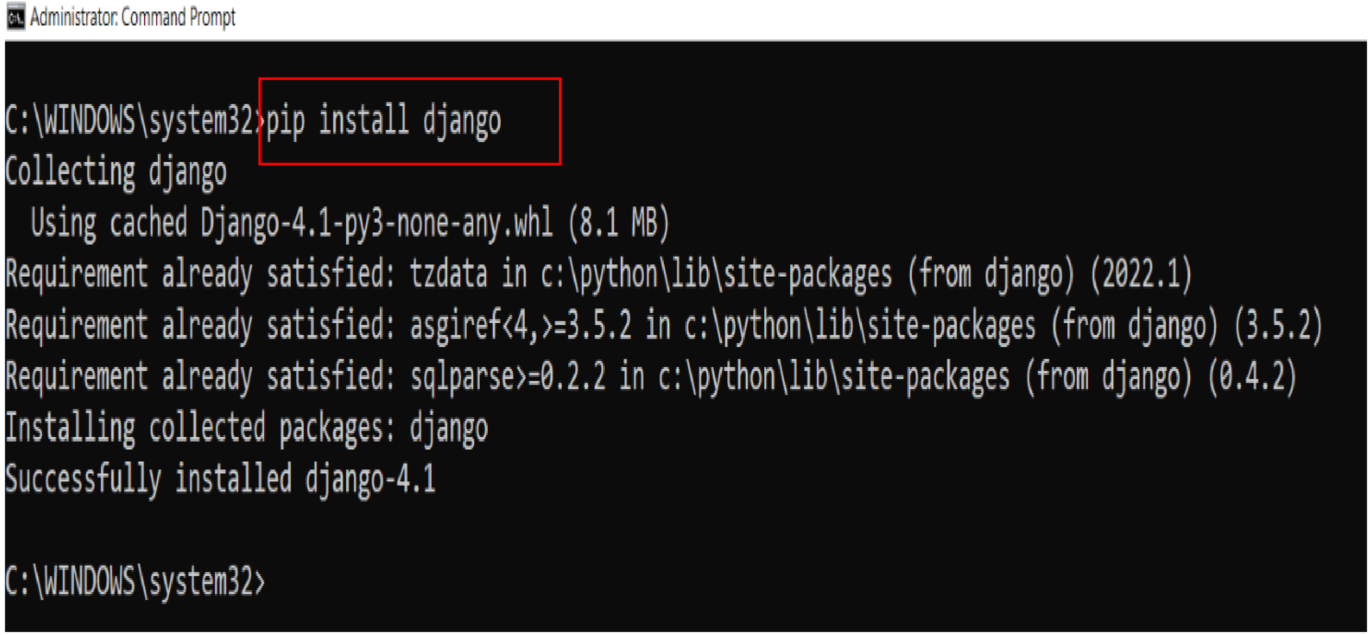
Check Python modules



Administrator: Command Prompt

```
(c) Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>pip freeze
asgiref==3.5.2
Django==4.1
mysql-connector==2.2.9
mysql-connector-python==8.0.28
protobuf==3.19.4
schedule==1.1.0
sqlparse==0.4.2
tzdata==2022.1

C:\WINDOWS\system32>pip uninstall django
Found existing installation: Django 4.1
Uninstalling Django-4.1:
  Would remove:
    c:\python\lib\site-packages\django-4.1.dist-info\*
    c:\python\lib\site-packages\django\*
    c:\python\scripts\django-admin.exe
Proceed (Y/n)? y
  Successfully uninstalled Django-4.1

C:\WINDOWS\system32>pip freeze
asgiref==3.5.2
mysql-connector==2.2.9
mysql-connector-python==8.0.28
protobuf==3.19.4
schedule==1.1.0
sqlparse==0.4.2
tzdata==2022.1

C:\WINDOWS\system32>
```

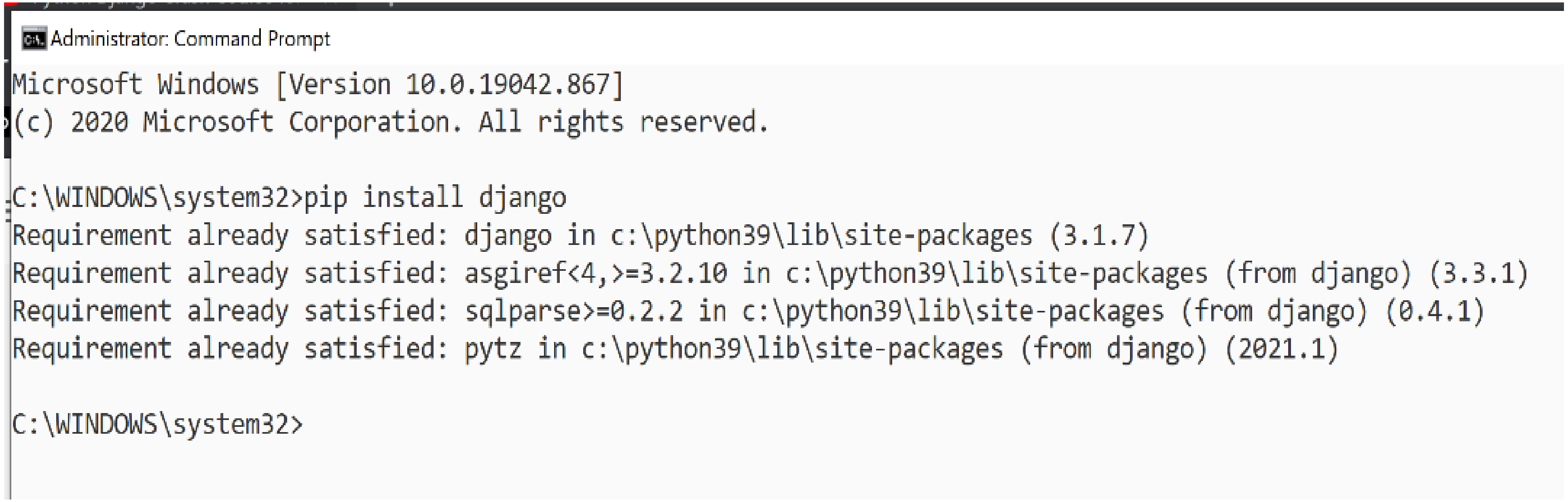
Install Django framework (pip install django)



```
C:\WINDOWS\system32>pip install django
Collecting django
  Using cached Django-4.1-py3-none-any.whl (8.1 MB)
Requirement already satisfied: tzdata in c:\python\lib\site-packages (from django) (2022.1)
Requirement already satisfied: asgiref<4,>=3.5.2 in c:\python\lib\site-packages (from django) (3.5.2)
Requirement already satisfied: sqlparse>=0.2.2 in c:\python\lib\site-packages (from django) (0.4.2)
Installing collected packages: django
Successfully installed django-4.1

C:\WINDOWS\system32>
```

django Install

- pip install django

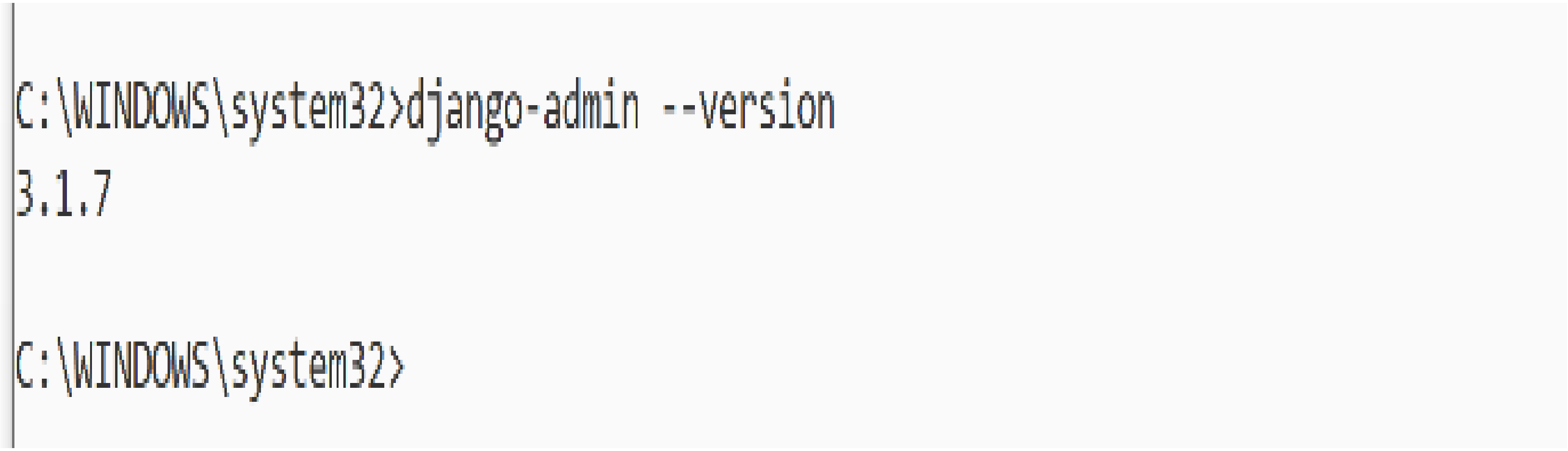


```
Microsoft Windows [Version 10.0.19042.867]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>pip install django
Requirement already satisfied: django in c:\python39\lib\site-packages (3.1.7)
Requirement already satisfied: asgiref<4,>=3.2.10 in c:\python39\lib\site-packages (from django) (3.3.1)
Requirement already satisfied: sqlparse>=0.2.2 in c:\python39\lib\site-packages (from django) (0.4.1)
Requirement already satisfied: pytz in c:\python39\lib\site-packages (from django) (2021.1)

C:\WINDOWS\system32>
```

Here message shows that django is already installed

django version test

- django-admin -- version

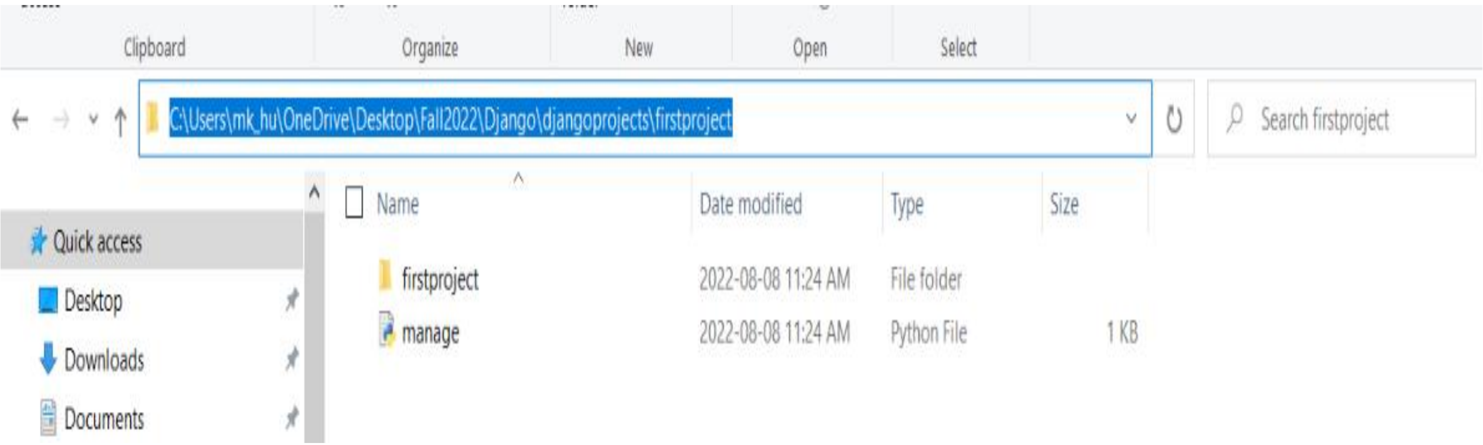


```
C:\WINDOWS\system32>django-admin --version
3.1.7

C:\WINDOWS\system32>
```

Steps (process) to create project

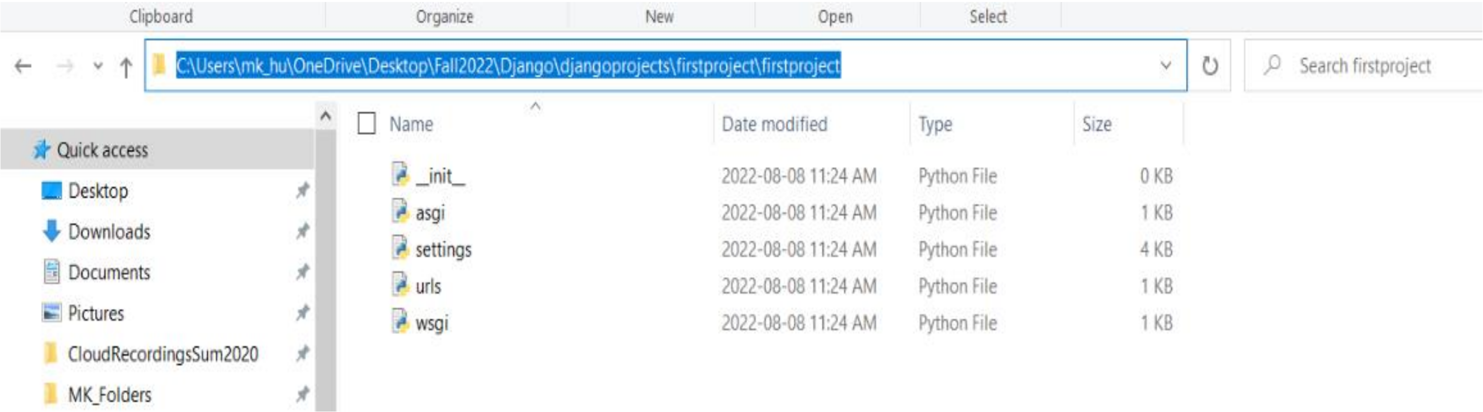
- Use ‘django-admin startproject project_name’
- Within the project (here project_name), create Python file ‘views.py’
- Create path in urls.py (setup route)
- Run server (python manage.py runserver)

Important files and folders in *firstproject*



C:\Users\mk_hu\OneDrive\Desktop\Fall2022\Django\djangoprojects\firstproject

Name	Date modified	Type	Size
firstproject	2022-08-08 11:24 AM	File folder	
manage	2022-08-08 11:24 AM	Python File	1 KB



C:\Users\mk_hu\OneDrive\Desktop\Fall2022\Django\djangoprojects\firstproject\firstproject

Name	Date modified	Type	Size
__init__	2022-08-08 11:24 AM	Python File	0 KB
asgi	2022-08-08 11:24 AM	Python File	1 KB
settings	2022-08-08 11:24 AM	Python File	4 KB
urls	2022-08-08 11:24 AM	Python File	1 KB
wsgi	2022-08-08 11:24 AM	Python File	1 KB

Project files created as project is initiated

- __init__.py

  An empty Python file that is created as project is created and is part of Python directed

- settings.py

  This is main settings file for the project. It points to URL location, Views location, settings for paths and routes, settings for database connectivity details, username password, settings for middleware and security and many more settings for web services

- urls.py

  This Python file contains the URL links to the corresponding Views to display web pages from the apps. This file may contain all URLs and links within site, but it can delegate URL links or routes to other files as well

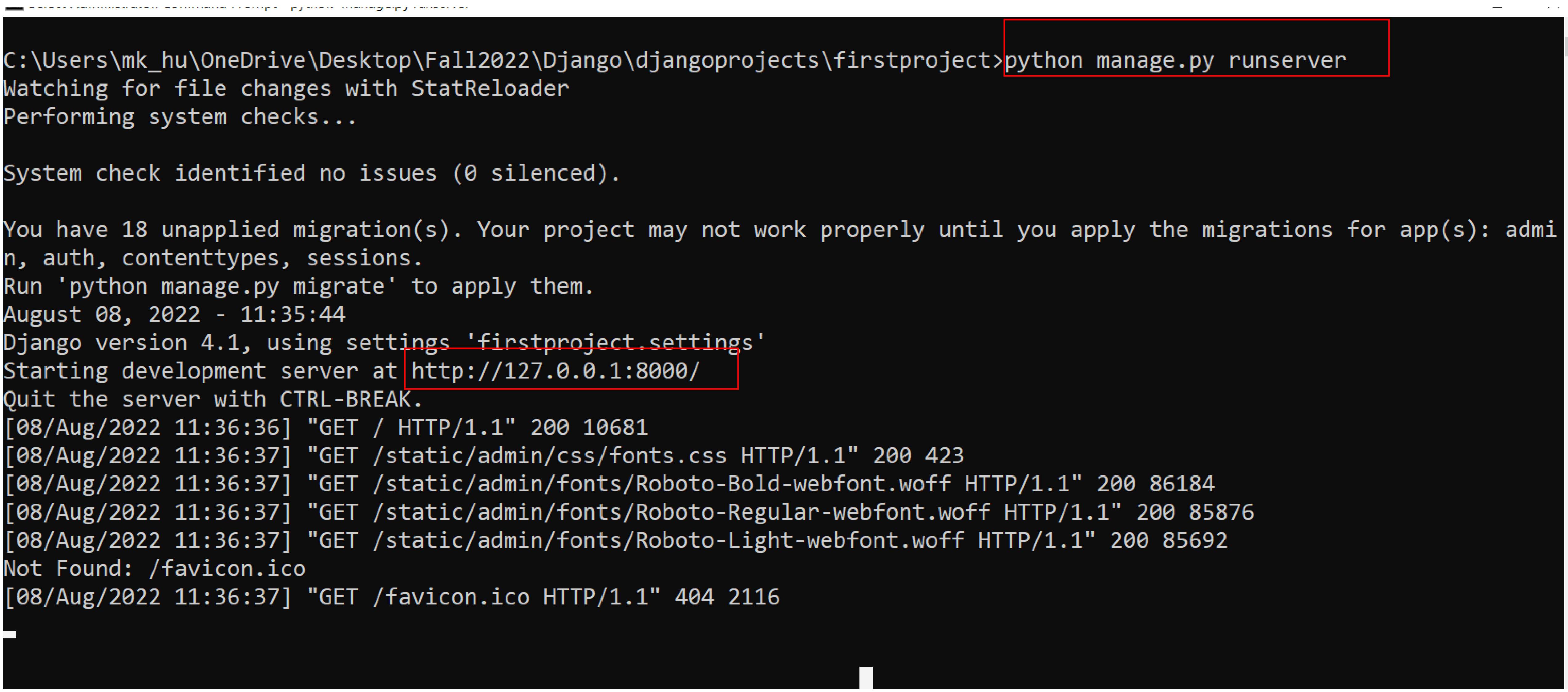
- wsgi.py (webserver gateway interface)

  This file initiates web service when it is called by the client. It refers to (in general) synchronous web services offered by the web server

- asgi.py (asynchronous server gateway interface)

  Successor to WSGI to support asynchronous web server apps and services

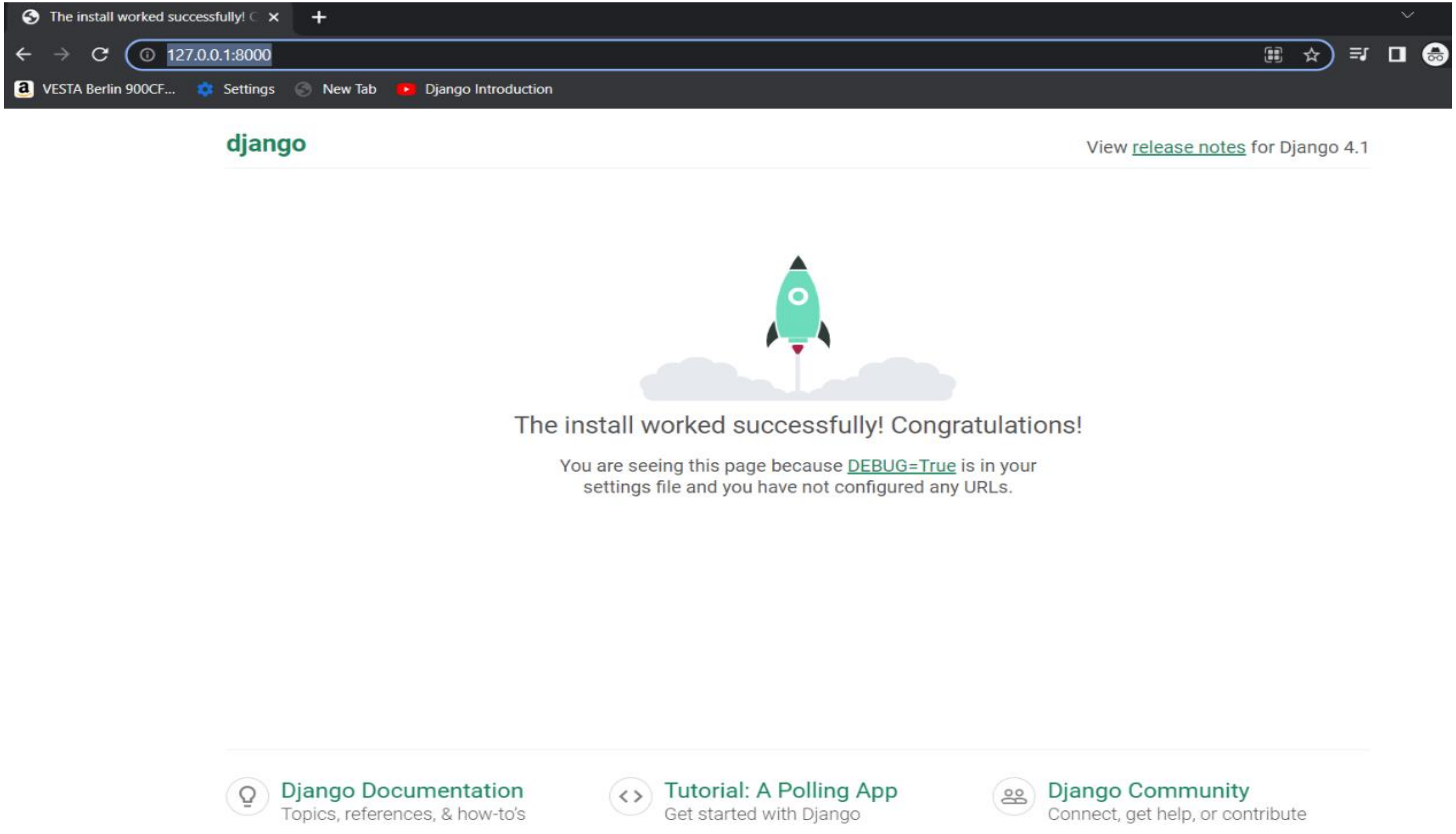
Run webserver in Django (python manage.py runserver)



```
C:\Users\mk_hu\OneDrive\Desktop\Fall2022\Django\djangoprojects\firstproject>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).

You have 18 unapplied migration(s). Your project may not work properly until you apply the migrations for app(s): admi
n, auth, contenttypes, sessions.
Run 'python manage.py migrate' to apply them.
August 08, 2022 - 11:35:44
Django version 4.1, using settings 'firstproject.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
[08/Aug/2022 11:36:36] "GET / HTTP/1.1" 200 10681
[08/Aug/2022 11:36:37] "GET /static/admin/css/fonts.css HTTP/1.1" 200 423
[08/Aug/2022 11:36:37] "GET /static/admin/fonts/Roboto-Bold-webfont.woff HTTP/1.1" 200 86184
[08/Aug/2022 11:36:37] "GET /static/admin/fonts/Roboto-Regular-webfont.woff HTTP/1.1" 200 85876
[08/Aug/2022 11:36:37] "GET /static/admin/fonts/Roboto-Light-webfont.woff HTTP/1.1" 200 85692
Not Found: /favicon.ico
[08/Aug/2022 11:36:37] "GET /favicon.ico HTTP/1.1" 404 2116
```

Open browser and run server

- Here default port number is 8000



django

View release notes for Django 4.1

The install worked successfully! Congratulations!

You are seeing this page because DEBUG=True is in your settings file and you have not configured any URLs.

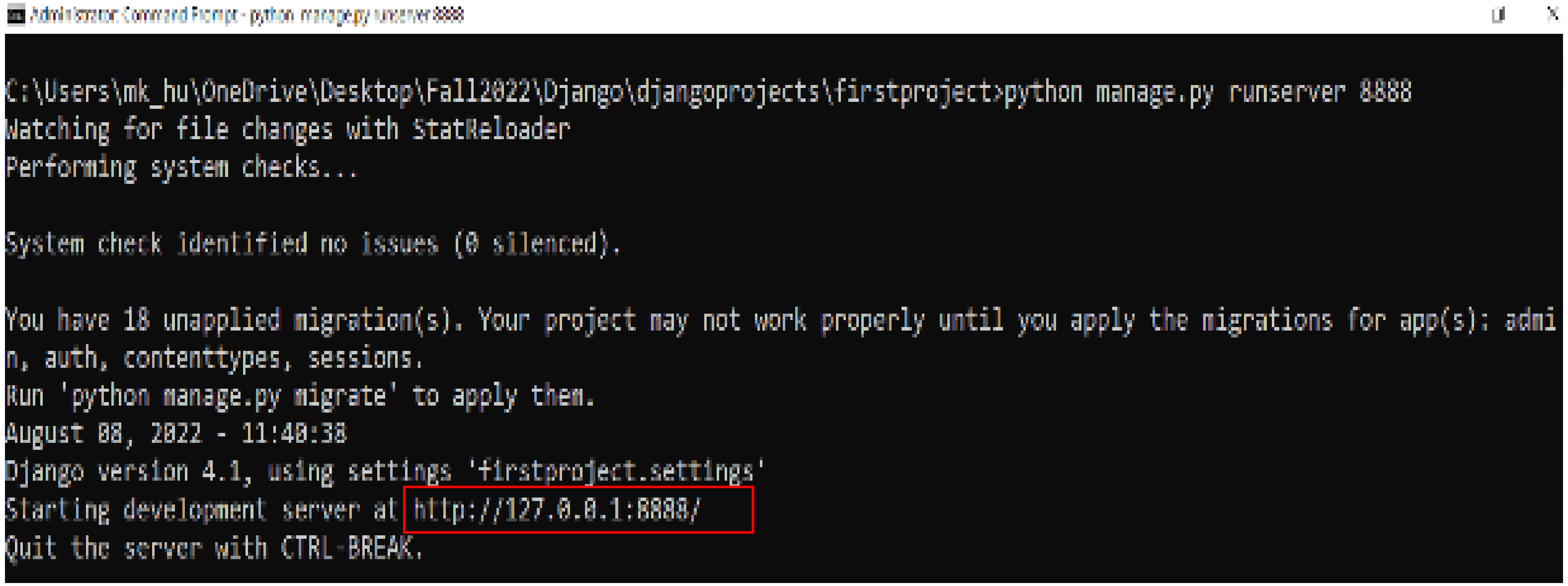
Django Documentation
Topics, references, & how-to's

Tutorial: A Polling App
Get started with Django

Django Community
Connect, get help, or contribute

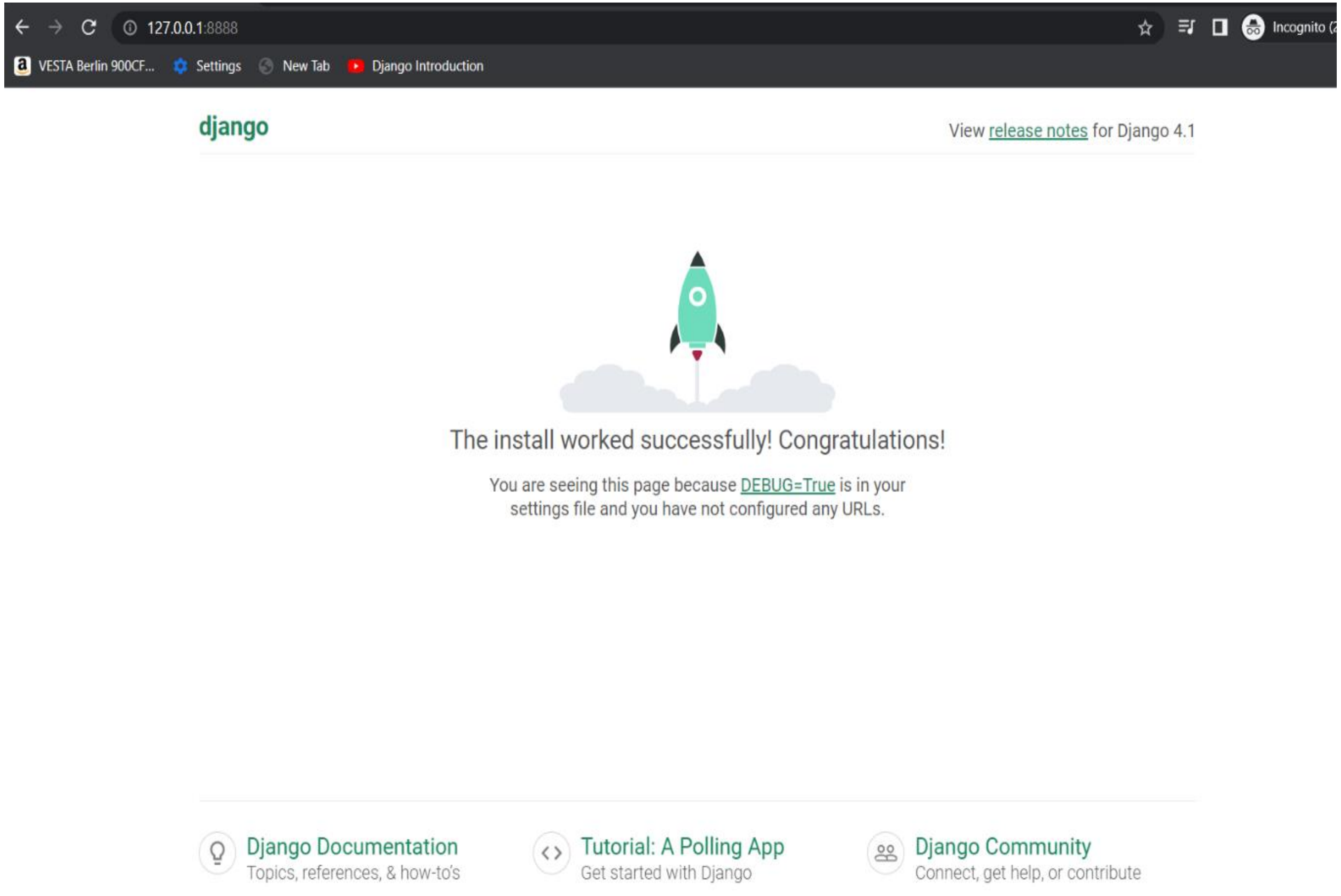
Changing default port number

- The port number is now 8888


Administrator: Command Prompt - python manage.py runserver 8888

```
C:\Users\mk_hu\OneDrive\Desktop\Fall2022\Django\djangoprojects\firstproject>python manage.py runserver 8888
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).

You have 18 unapplied migration(s). Your project may not work properly until you apply the migrations for app(s): admi
n, auth, contenttypes, sessions.
Run 'python manage.py migrate' to apply them.
August 08, 2022 - 11:40:38
Django version 4.1, using settings 'firstproject.settings'
Starting development server at http://127.0.0.1:8888/
Quit the server with CTRL-BREAK.
```

Open browser and watch web server run on port 8888



django

View release notes for Django 4.1

The install worked successfully! Congratulations!

You are seeing this page because DEBUG=True is in your settings file and you have not configured any URLs.

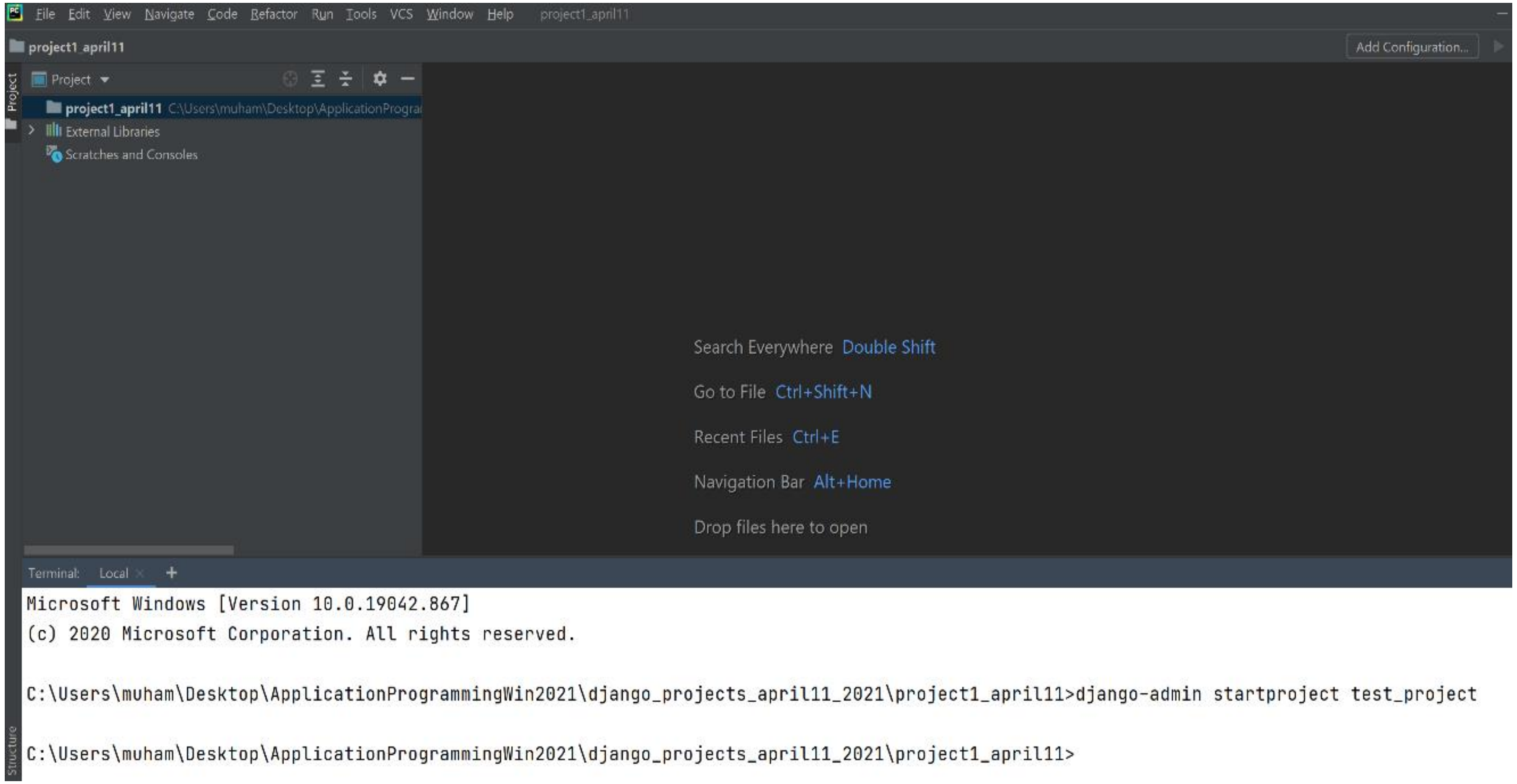
Django Documentation
Topics, references, & how-to's

Tutorial: A Polling App
Get started with Django

Django Community
Connect, get help, or contribute

Creating django project

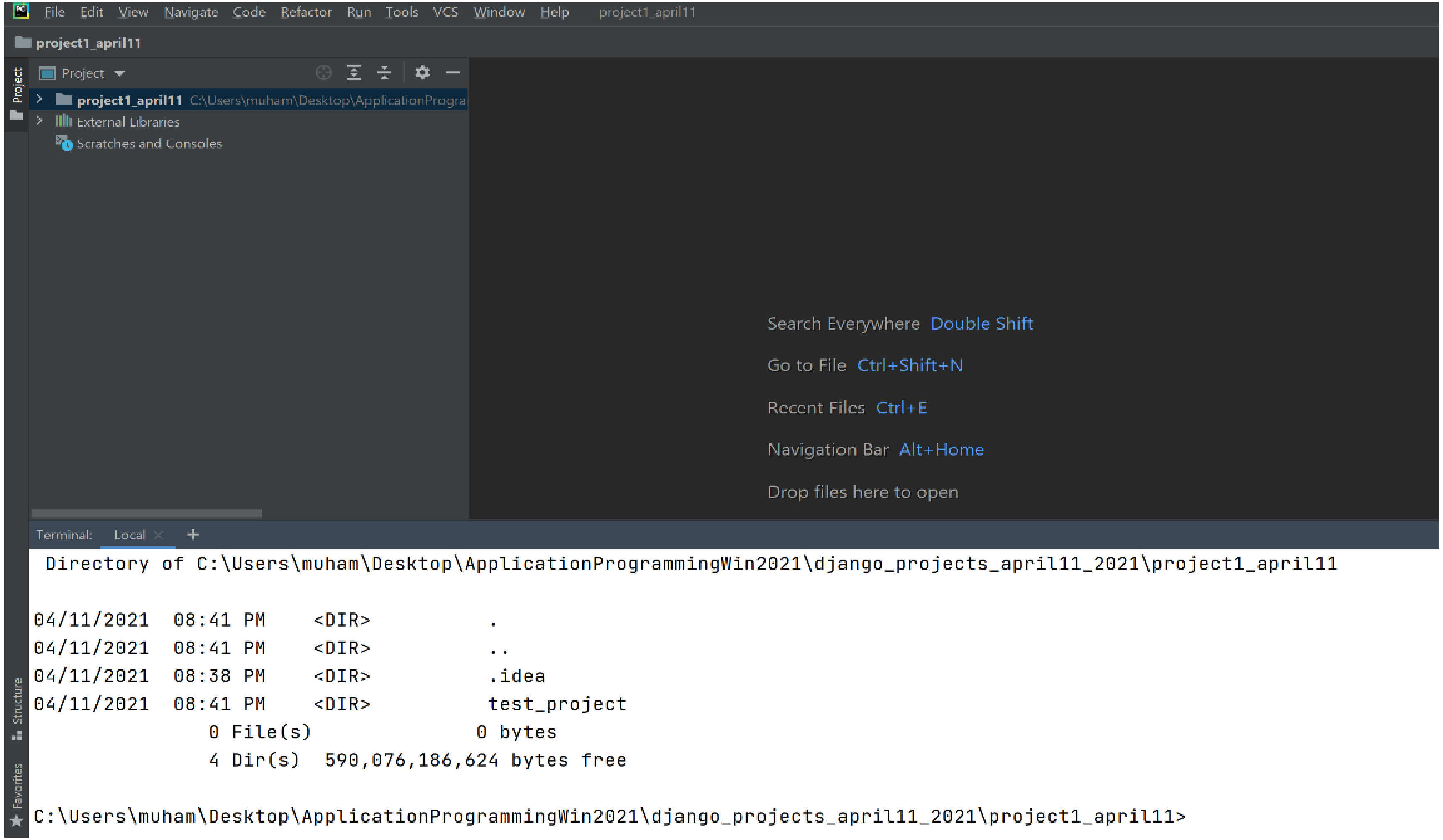
- In Pycharm – access terminal and use the command:
- django-admin startproject <project name goes here>



```
Microsoft Windows [Version 10.0.19042.867]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\muham\Desktop\ApplicationProgrammingWin2021\django_projects_april11_2021\project1_april11>django-admin startproject test_project

C:\Users\muham\Desktop\ApplicationProgrammingWin2021\django_projects_april11_2021\project1_april11>
```

test_project is created in the folder

test_project folder is created with another test_project folder

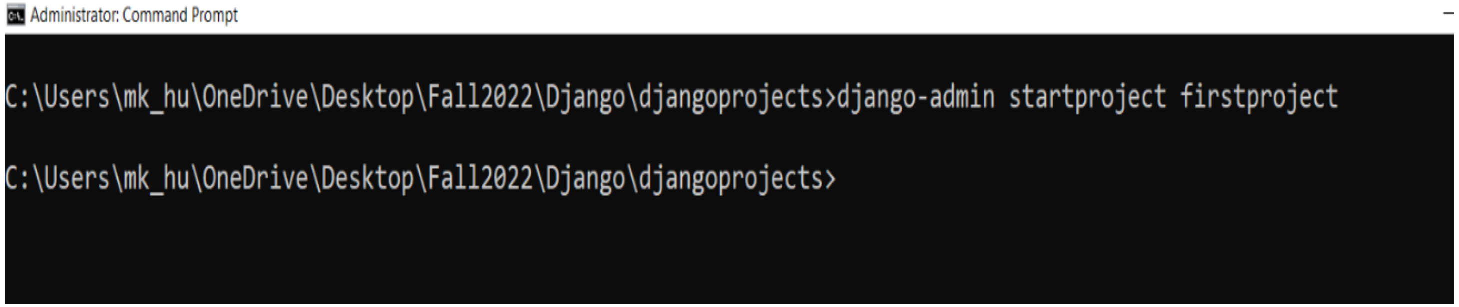


```
 Directory of C:\Users\muham\Desktop\ApplicationProgrammingWin2021\django_projects_april11_2021\project1_april11

04/11/2021  08:41 PM    <DIR>          .
04/11/2021  08:41 PM    <DIR>          ..
04/11/2021  08:38 PM    <DIR>          .idea
04/11/2021  08:41 PM    <DIR>          test_project
               0 File(s)              0 bytes
               4 Dir(s)  590,076,186,624 bytes free

C:\Users\muham\Desktop\ApplicationProgrammingWin2021\django_projects_april11_2021\project1_april11>
```

Create django project

- Project can be created using the command:

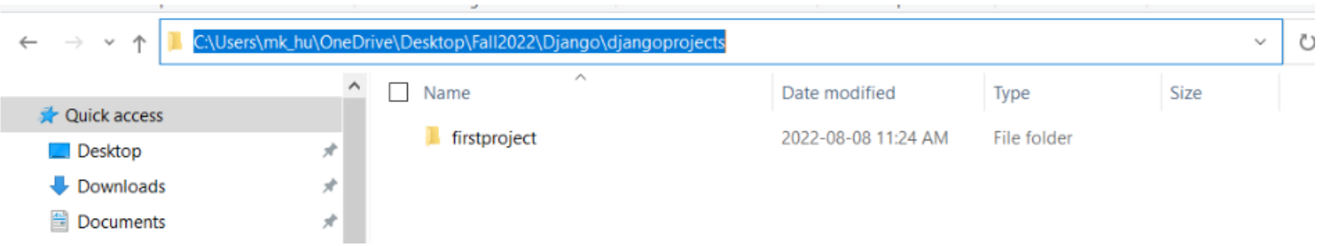
django-admin startproject projectname



Administrator: Command Prompt

```
C:\Users\mk_hu\OneDrive\Desktop\Fall2022\Django\djangoprojects>django-admin startproject firstproject

C:\Users\mk_hu\OneDrive\Desktop\Fall2022\Django\djangoprojects>
```

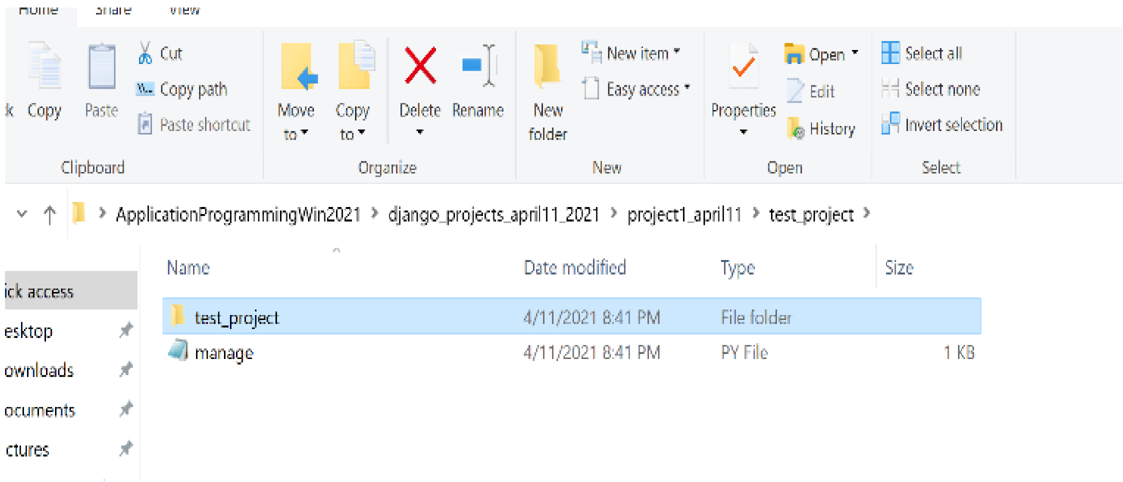
Here firstproject is created (as a folder with number of files in it)



C:\Users\mk_hu\OneDrive\Desktop\Fall2022\Django\djangoprojects

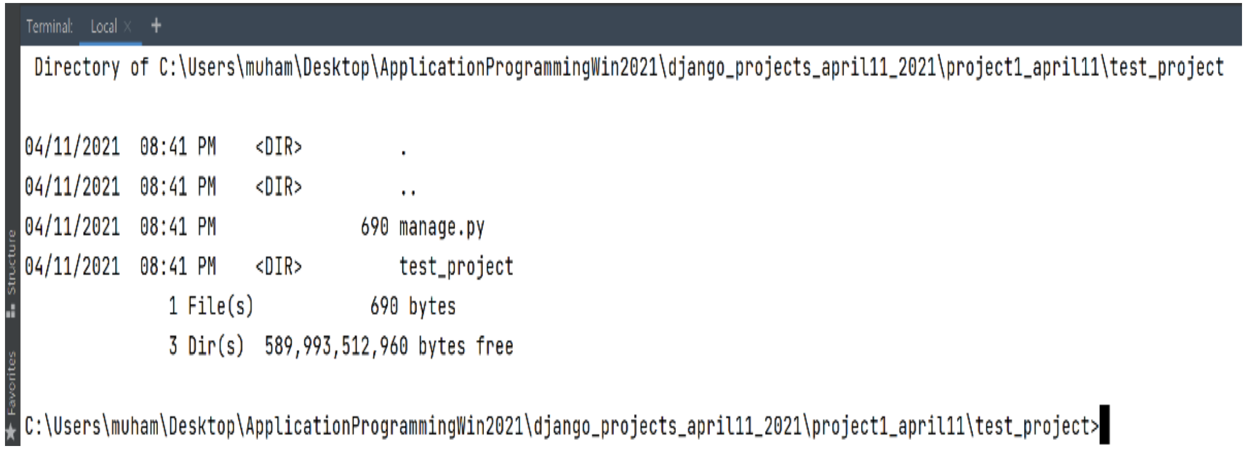
Name	Date modified	Type	Size
firstproject	2022-08-08 11:24 AM	File folder	

Contents of test_project folder



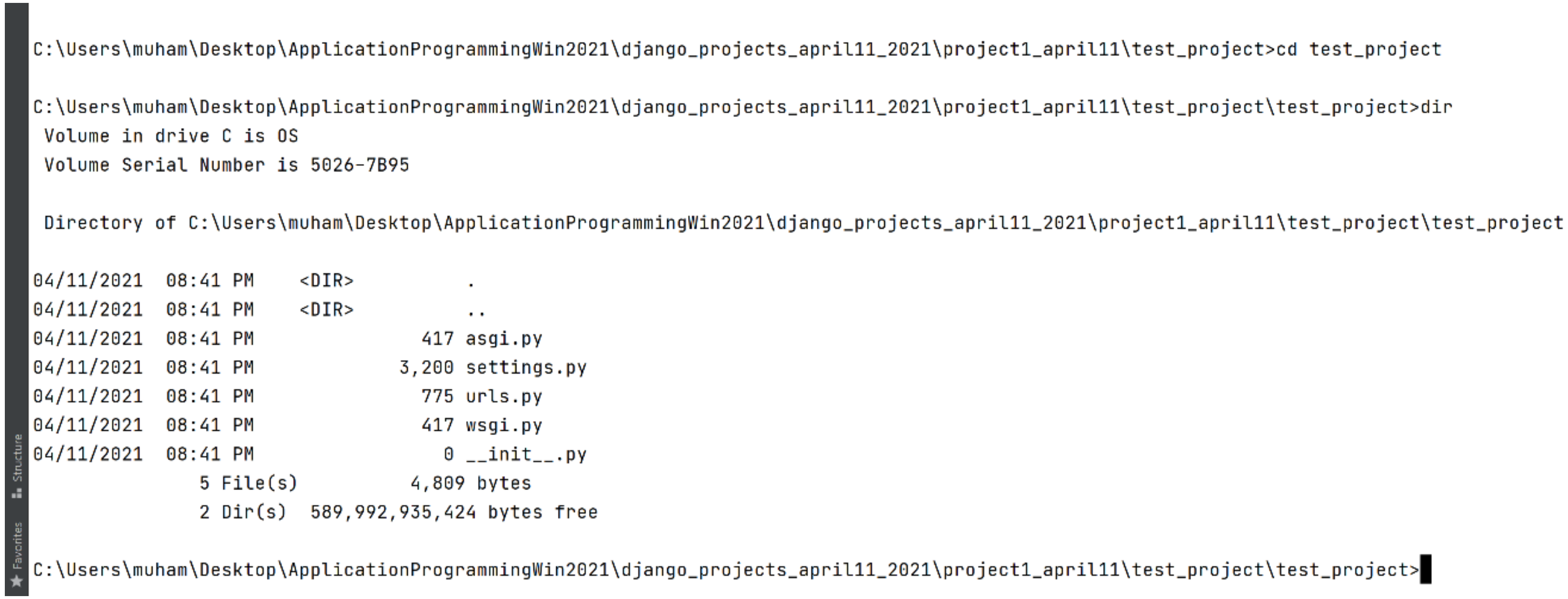
Name	Date modified	Type	Size
test_project	4/11/2021 8:41 PM	File folder	
manage	4/11/2021 8:41 PM	PY File	1 KB

manage.py is an important Python file to start web server

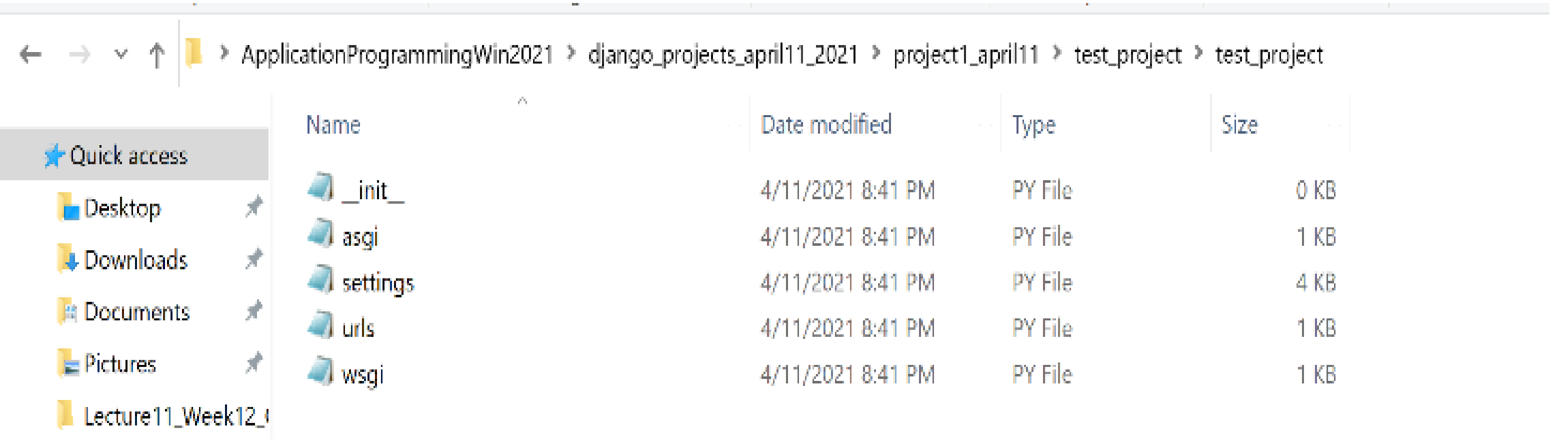


```
 Directory of C:\Users\muham\Desktop\ApplicationProgrammingWin2021\django_projects_april11_2021\project1_april11\test_project


04/11/2021  08:41 PM    <DIR>          .
04/11/2021  08:41 PM    <DIR>          ..
04/11/2021  08:41 PM               690 manage.py
04/11/2021  08:41 PM    <DIR>          test_project
               1 File(s)            690 bytes
               3 Dir(s)  589,993,512,960 bytes free

C:\Users\muham\Desktop\ApplicationProgrammingWin2021\django_projects_april11_2021\project1_april11\test_project>
```

Contents of sub-folder (test_project)



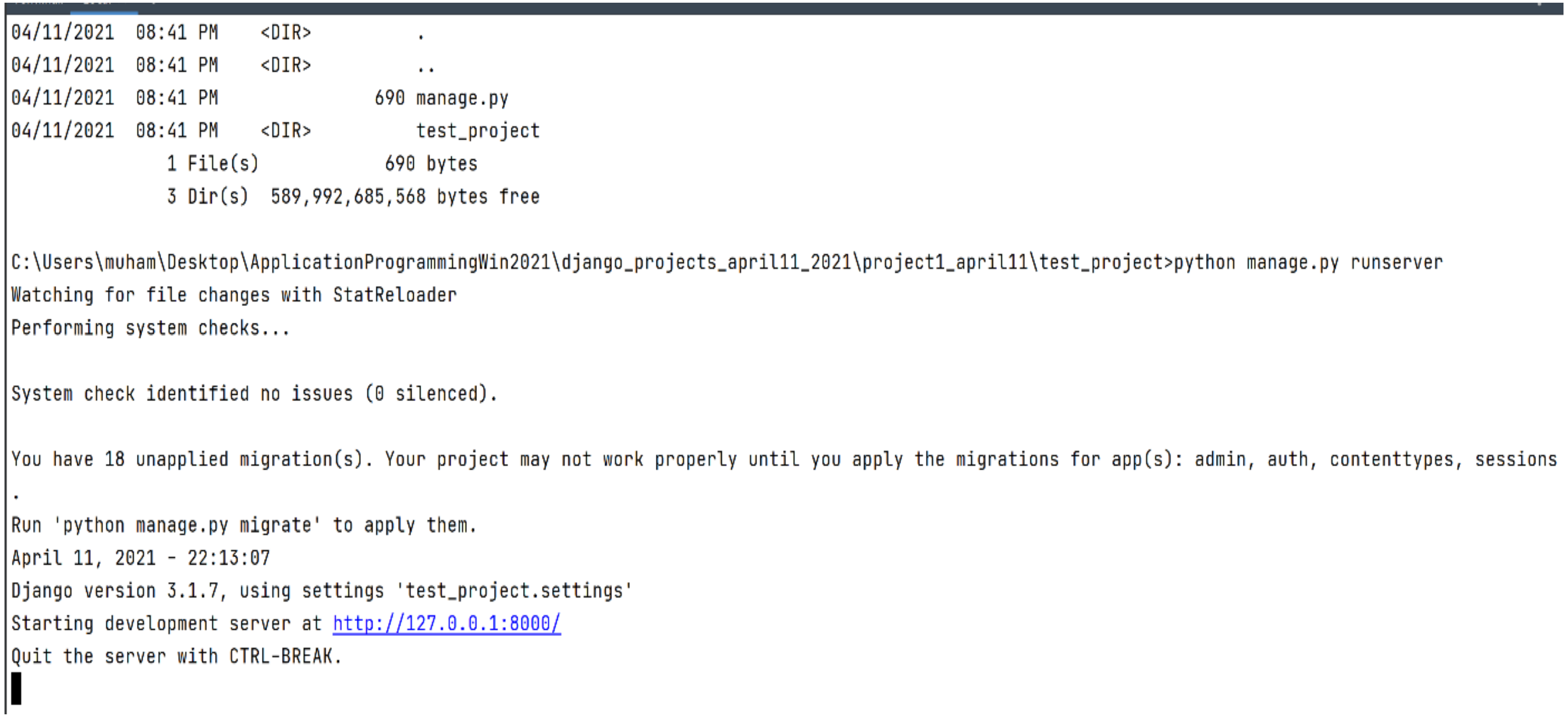
```
C:\Users\muham\Desktop\ApplicationProgrammingWin2021\django_projects_april11_2021\project1_april11\test_project>cd test_project

C:\Users\muham\Desktop\ApplicationProgrammingWin2021\django_projects_april11_2021\project1_april11\test_project\test_project>dir
 Volume in drive C is OS
 Volume Serial Number is 5026-7B95

 Directory of C:\Users\muham\Desktop\ApplicationProgrammingWin2021\django_projects_april11_2021\project1_april11\test_project\test_project

04/11/2021  08:41 PM    <DIR>          .
04/11/2021  08:41 PM    <DIR>          ..
04/11/2021  08:41 PM               417 asgi.py
04/11/2021  08:41 PM             3,200 settings.py
04/11/2021  08:41 PM               775 urls.py
04/11/2021  08:41 PM               417 wsgi.py
04/11/2021  08:41 PM                 0 __init__.py
               5 File(s)          4,809 bytes
               2 Dir(s)  589,992,935,424 bytes free

C:\Users\muham\Desktop\ApplicationProgrammingWin2021\django_projects_april11_2021\project1_april11\test_project\test_project>
```



ApplicationProgrammingWin2021 › django_projects_april11_2021 › project1_april11 › test_project › test_project
Quick access
Desktop
Downloads
Documents
Pictures
Lecture11_Week12_

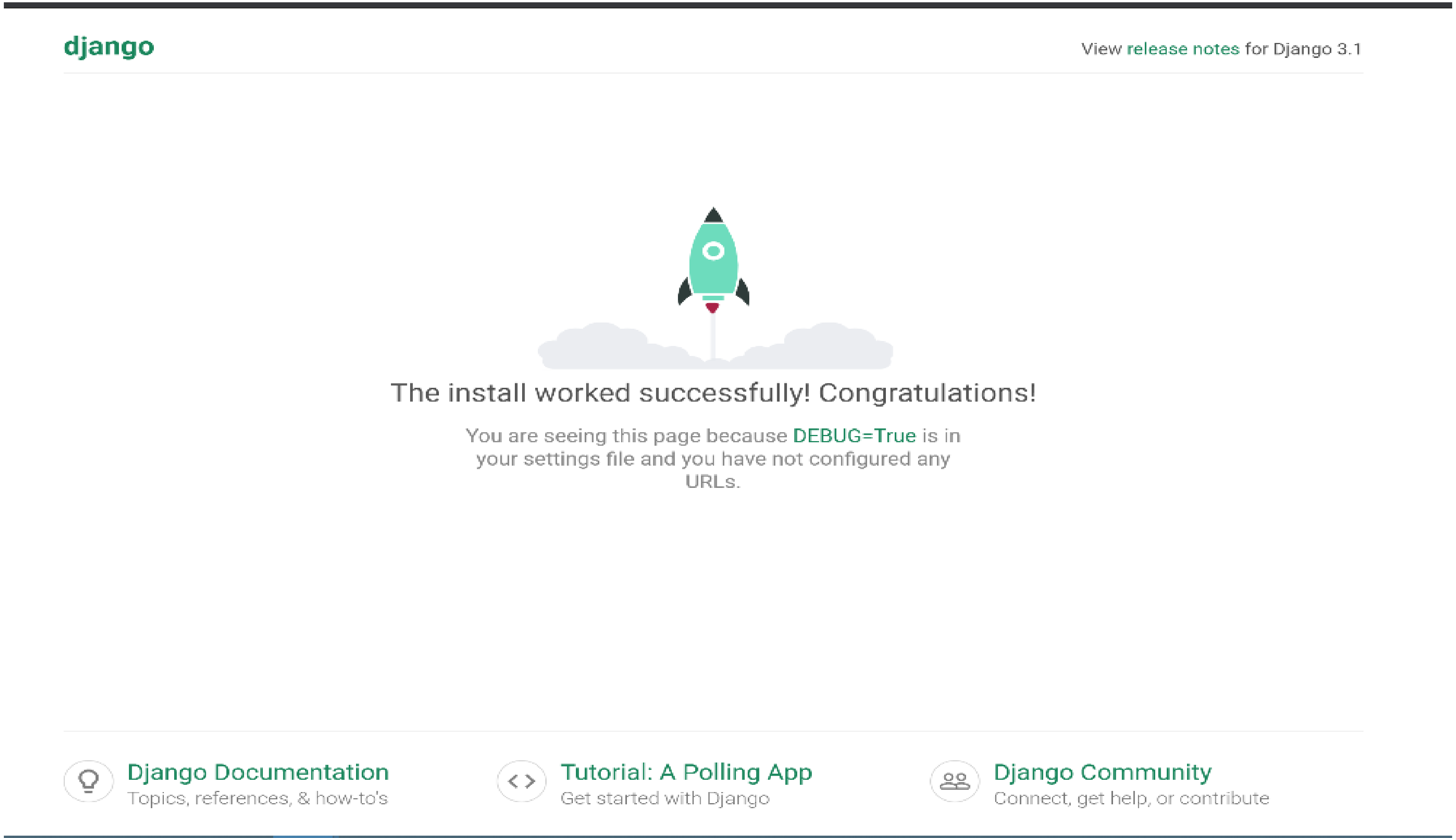
Name	Date modified	Type	Size
init	4/11/2021 8:41 PM	PY File	0 KB
asgi	4/11/2021 8:41 PM	PY File	1 KB
settings	4/11/2021 8:41 PM	PY File	4 KB
urls	4/11/2021 8:41 PM	PY File	1 KB
wsgi	4/11/2021 8:41 PM	PY File	1 KB

Run web server (python manage.py runserver)



```
04/11/2021  08:41 PM    <DIR>          .
04/11/2021  08:41 PM    <DIR>          ..
04/11/2021  08:41 PM               690 manage.py
04/11/2021  08:41 PM    <DIR>          test_project
               1 File(s)            690 bytes
               3 Dir(s)  589,992,685,568 bytes free

C:\Users\muham\Desktop\ApplicationProgrammingWin2021\django_projects_april11_2021\project1_april11\test_project>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).

You have 18 unapplied migration(s). Your project may not work properly until you apply the migrations for app(s): admin, auth, contenttypes, sessions
.
Run 'python manage.py migrate' to apply them.
April 11, 2021 - 22:13:07
Django version 3.1.7, using settings 'test_project.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

Web-Server display



django

View release notes for Django 3.1

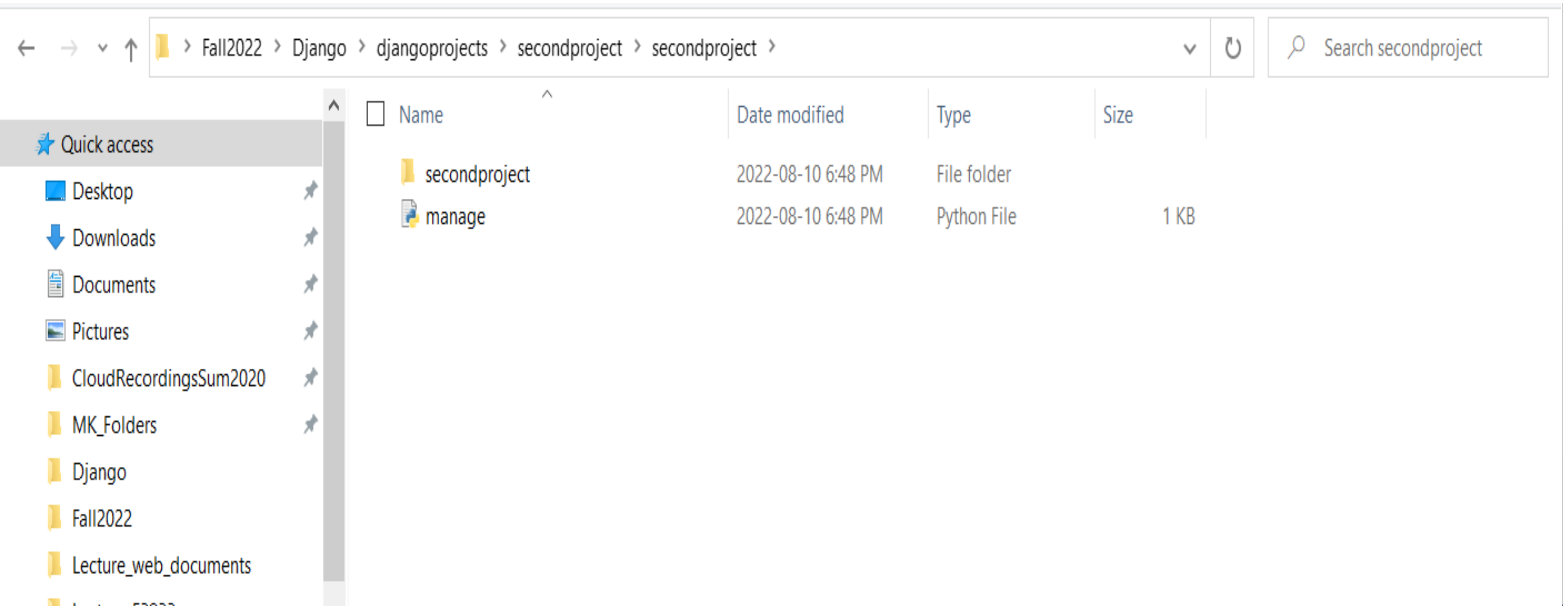
The install worked successfully! Congratulations!

You are seeing this page because DEBUG=True is in your settings file and you have not configured any URLs.

Django Documentation
Topics, references, & how-to's

Tutorial: A Polling App
Get started with Django

Django Community
Connect, get help, or contribute

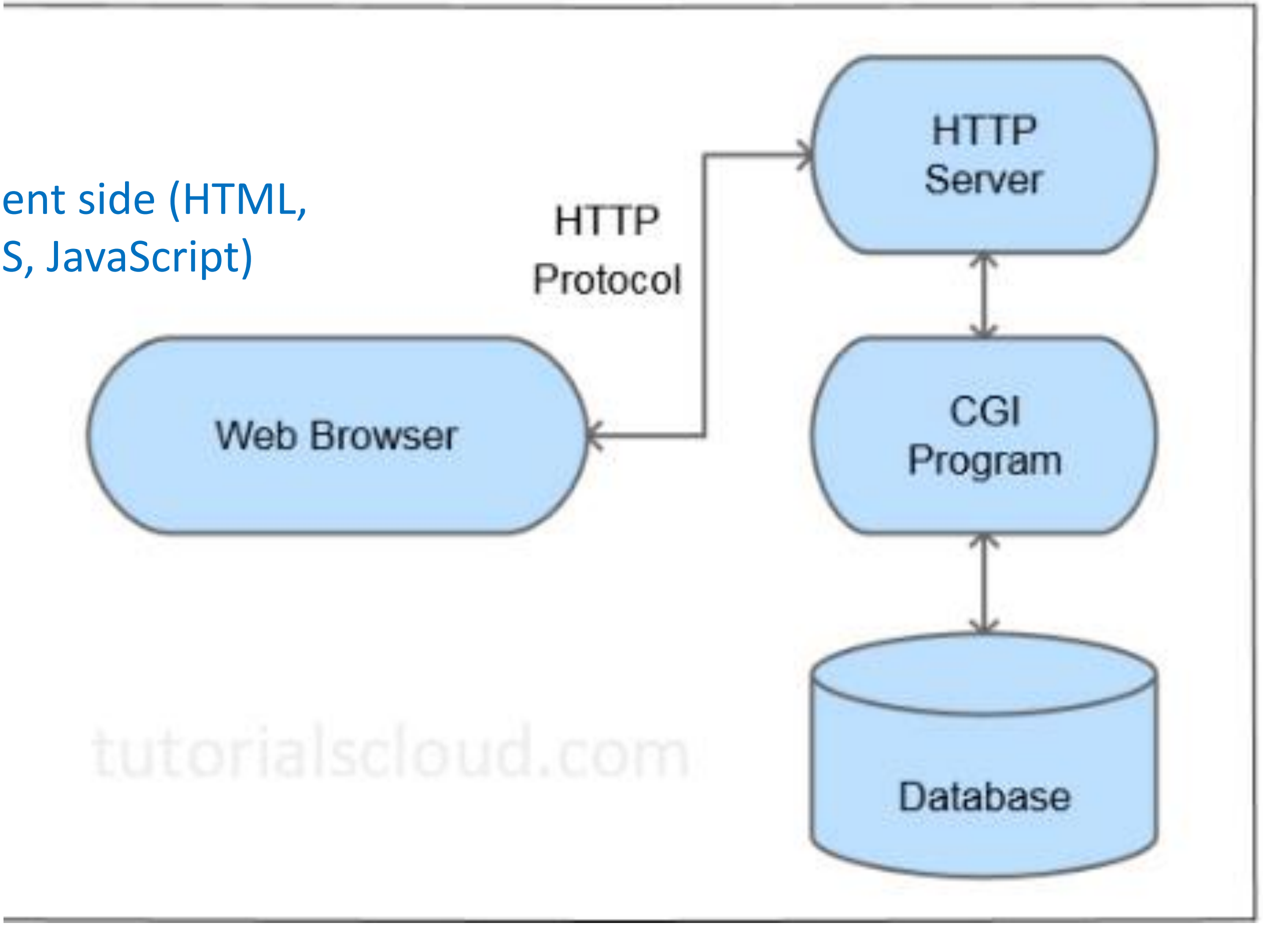


Name	Date modified	Type	Size
secondproject	2022-08-10 6:48 PM	File folder	
manage	2022-08-10 6:48 PM	Python File	1 KB

Traditional client – server model

A networking tool (e.g. Browser and network infrastructure) requests access to web server and is authenticated by Authentication server

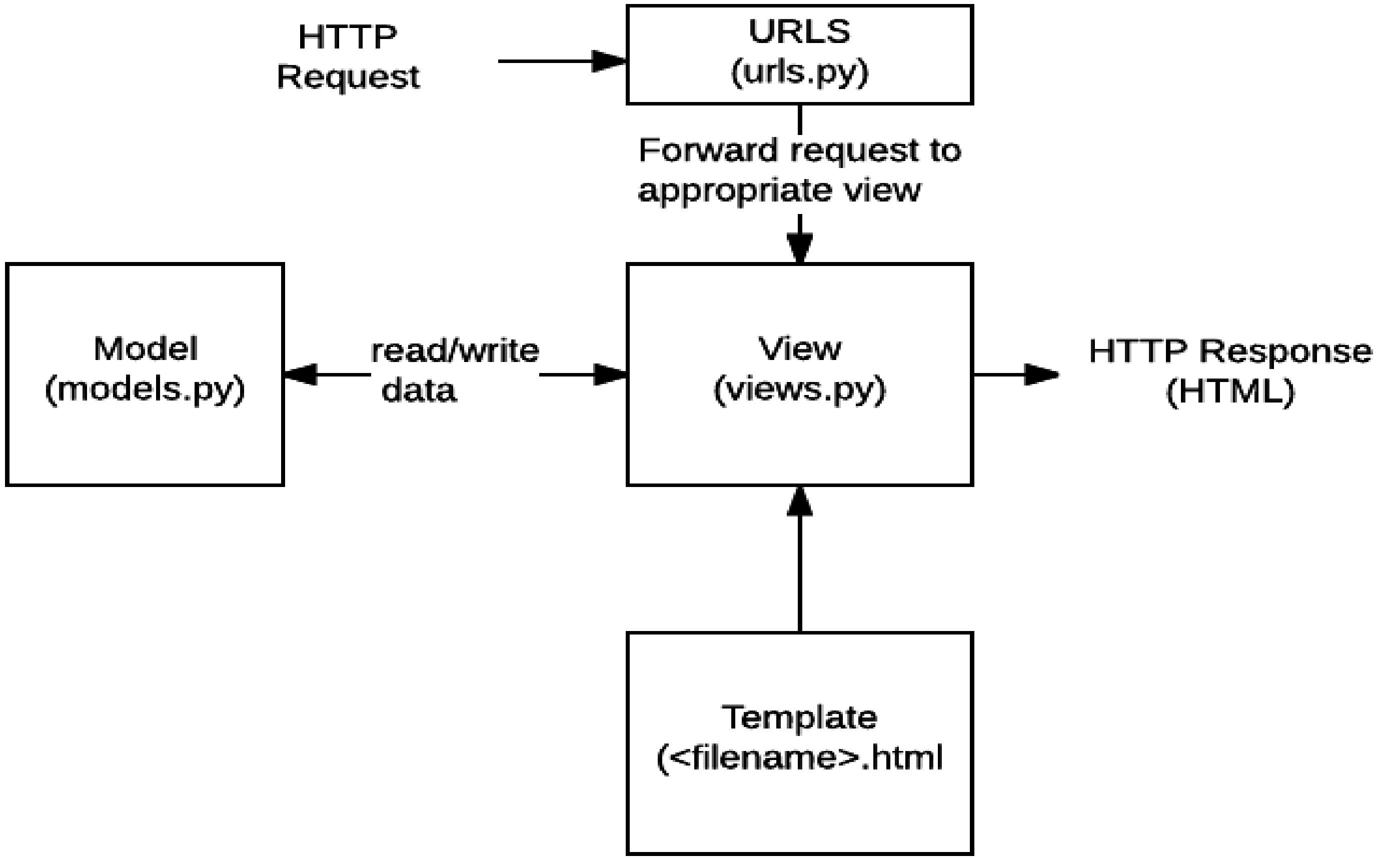
Client side (HTML, CSS, JavaScript)



Web Server (Apache, Tomcat, IIS – may be part of LAMP or XAMP)

Authentication server
Database server

django based MVT pattern



Reference: https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django
Reference: https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django/Introduction

Summary

- Framework – an introduction
- Commonly used Python frameworks
- Key features of django framework
- Client side documents (html, css, javascript)
- Create django project
- File structure in django project
- MVT architecture