### WEEK-9

1. **What is Django, what is the framework that is used to implement Django.**

**DJANGO**

Django is a high-level Python web framework that encourages rapid development and clean, pragmatic design. Built by experienced developers, it takes care of much of the hassle of web development, so you can focus on writing your app without needing to reinvent the wheel. It’s free and open source,Ridiculously fast.

Django was designed to help developers take applications from concept to completion as quickly as possible reassuringly secure.Django takes security seriously and helps developers avoid many common security mistakes Exceedingly scalable Some of the busiest sites on the web leverage Django’s ability to quickly and flexibly scale.

FRAMEWORK

Rich ecosystem

Read Django like a system, developers say. What they mean is that there are many third-party applications that come with Django. These applications can be integrated depending on project requirements. To imagine this better, think of Legos. There are many different Lego blocks. In app development, an authorization “block” or email sending “block” is present in almost every project. Django consists of many applications — such as for authorization and sending emails — that can easily be plugged into a system.

Maturity. Django has been around for 11 years and has gone through stages of significant improvement. A lot of things have been brought to perfection and many new things have been added. Most importantly, when you’re trying to figure out how something should work in Django, you can usually find the answer. Thousands of people must have already solved any issue you’re dealing with, and you can find a solution provided by the passionate Django community.

Admin panel by default

Admin panels are designed to help you manage your application. A Django admin panel is generated automatically from Python code, whereas creating an admin panel manually would take a lot of time and be absolutely pointless.

Python is famous for having human-readable code, and that’s an advantage if you want your site to rank high in search results. With Django, you can generate readable website URLs and links using the most relevant keywords and search engine optimization (SEO) best practices.

After all, a domain name is just a “human-readable” string that maps to a “computer friendly” set of numbers, known as an IP address. People fixate on getting the right domain name, but tend to neglect what the URL slug is—Django can fix that Pluggable

Django is pluggable by nature and can be extended with plugins. Plugins are software components that allow developers to add a specific feature to an app, leaving a lot of scope for customization. There are hundreds of packages to help you add Google Maps, create complex permissions, or connect to Stripe to process payments. And if you need to scale your project in the future, you can unplug some components and replace them with others that meet your current needs Libraries

Every programming language comes with its own set of libraries for solving common tasks. A software library includes prewritten code, classes, procedures, scripts, configuration data, and more. As a rule, a library is added to a program to provide more functionality or to automate a process without manually writing new code. This reduces time to market.

Django allows developers to use libraries when building any project. Some popular libraries include the Django REST framework, which is responsible for building application programming interfaces (APIs); Django CMS, which is designed to manage website content; and

Django-allauth, which is an integrated set of Django applications for authentication, registration, account management, and third-party (social) account authentication ORM.Django is valued for its object-relational mapper that helps developers interact with databases.An object-relational mapper (ORM) is a library that automatically transfers data stored in databases such as PostgreSQL and MySQL into objects commonly used in application code.

1. **What are the prerequisites to implement Django.**

There are many prerequisites to learn Django: The basic syntax of the Python

Regular Expression (RE) object-oriented concepts Learn REST APIs and JSON

Database Management and SQL queries

1. **How to install and run Django**

Install Django¶

Django can be installed easily using pip.

In the command prompt, execute the following command: pip install django. This will download and install Django.

After the installation has completed, you can verify your Django installation by executing django-admin --version in the command prompt.

Use the Django admin console

Create a superuser. You will be prompted to enter a username, email, and password. python manage. py createsuperuser.

Start a local web server: python manage. py runserver.

Log in to the admin site using the username and password you used when you ran createsuperuser .

1. **Write a very basic program using Django.** django\_admin startproject hello\_world\_project hello\_world\_project/

manage.py helloworld\_project/

init .py

settings.py urls.py asgi.py

migrating and testing project cd hello\_world\_project python manage.py migrate python manage.py runserver

1. **What is a database.Which database is by default in Django.**

A database is an organized collection of structured information, or data, typically stored electronically in a computer system. A database is usually controlled by a database management system (DBMS). Together, the data and the DBMS, along with the applications that are associated with them, are referred to as a database system, often shortened to just database.

Data within the most common types of databases in operation today is typically modeled in rows and columns in a series of tables to make processing and data querying efficient. The data can then be easily accessed, managed, modified, updated, controlled, and organized. Most databases use structured query language (SQL) for writing and querying data.By default, the configuration uses SQLite. If you’re new to databases, or you’re just interested in trying Django, this is the easiest.

1. **What is the framework and advantages of Django.**

Rich ecosystem

Read Django like a system, developers say. What they mean is that there are many third-party applications that come with Django. These applications can be integrated depending on project requirements. To imagine this better, think of Legos. There are many different Lego blocks. In app development, an authorization “block” or email sending “block” is present in almost every project. Django consists of many applications — such as for authorization and sending emails — that can easily be plugged into a system.

Maturity. Django has been around for 11 years and has gone through stages of significant improvement. A lot of things have been brought to perfection and many new things have been added. Most importantly, when you’re trying to figure out how something should work in Django, you can usually find the answer. Thousands of people must have already solved any

issue you’re dealing with, and you can find a solution provided by the passionate Django community.

Admin panel by default

Admin panels are designed to help you manage your application. A Django admin panel is generated automatically from Python code, whereas creating an admin panel manually would take a lot of time and be absolutely pointless.

There’s a lot of room for customization in the Django admin panel thanks to third-party applications. Additionally, Django allows you to modify the interface with third-party wrappers and add dashboards unique to your needs.

Good for SEO

Python is famous for having human-readable code, and that’s an advantage if you want your site to rank high in search results. With Django, you can generate readable website URLs and links using the most relevant keywords and search engine optimization (SEO) best practices.

After all, a domain name is just a “human-readable” string that maps to a “computer friendly” set of numbers, known as an IP address. People fixate on getting the right domain name, but tend to neglect what the URL slug is—Django can fix that.

Pluggable

Django is pluggable by nature and can be extended with plugins. Plugins are software components that allow developers to add a specific feature to an app, leaving a lot of scope for customization. There are hundreds of packages to help you add Google Maps, create complex permissions, or connect to Stripe to process payments. And if you need to scale your project in the future, you can unplug some components and replace them with others that meet your current needs.

Libraries

Every programming language comes with its own set of libraries for solving common tasks. A software library includes prewritten code, classes, procedures, scripts, configuration data, and more. As a rule, a library is added to a program to provide more functionality or to automate a process without manually writing new code. This reduces time to market.

Django allows developers to use libraries when building any project. Some popular libraries include the Django REST framework, which is responsible for building application programming interfaces (APIs); Django CMS, which is designed to manage website content; and

Django-allauth, which is an integrated set of Django applications for authentication, registration, account management, and third-party (social) account authentication.

ORM

Django is valued for its object-relational mapper that helps developers interact with databases. An object-relational mapper (ORM) is a library that automatically transfers data stored in databases such as PostgreSQL and MySQL into objects commonly used in application code.

1. **How to create an environment to start a new project in Django.**
2. Install Package. First, install python3-venv package by using the following command.
3. Create a Directory. $ mkdir djangoenv. ...
4. Create Virtual Environment. $ python3 -m venv djangoenv. ...
5. Activate Virtual Environment.

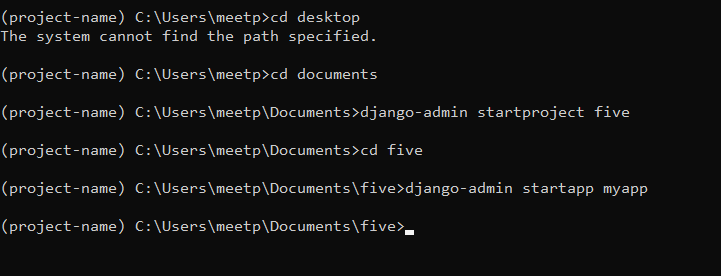
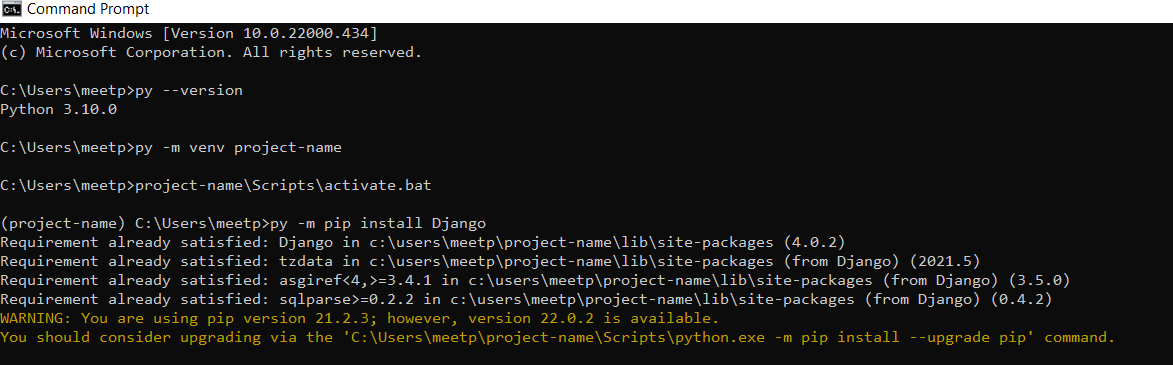
**8. What is the difference between project and app.**

A *project* refers to the entire application and all its parts.

An *app* refers to a submodule of the project. It's self-sufficient and not intertwined with the other apps in the project such that, in theory, you could pick it up and plop it down into another project without any modification. An *app* typically has its own *models.py* (which might actually be empty). You might think of it as a standalone python module. A simple project might only have one app.

For your example, the *project* is the whole website. You might structure it so there is an *app* for articles, an *app* for ranking tables, and an *app* for fixtures and results. If they need to interact with each other, they do it through well-documented public classes and accessor methods.

The main thing to keep in mind is this level of interdependence between the *apps*. In practice it's all one *project*, so there's no sense in going overboard, but keep in mind how co-dependent two apps are. If you find one app is solving two problems, split them into two apps. If you find two apps are so intertwined you could never reuse one without the other, combine them into a single app.



### WEEK-10

**1.Aim: To create forms using django. procedure:**

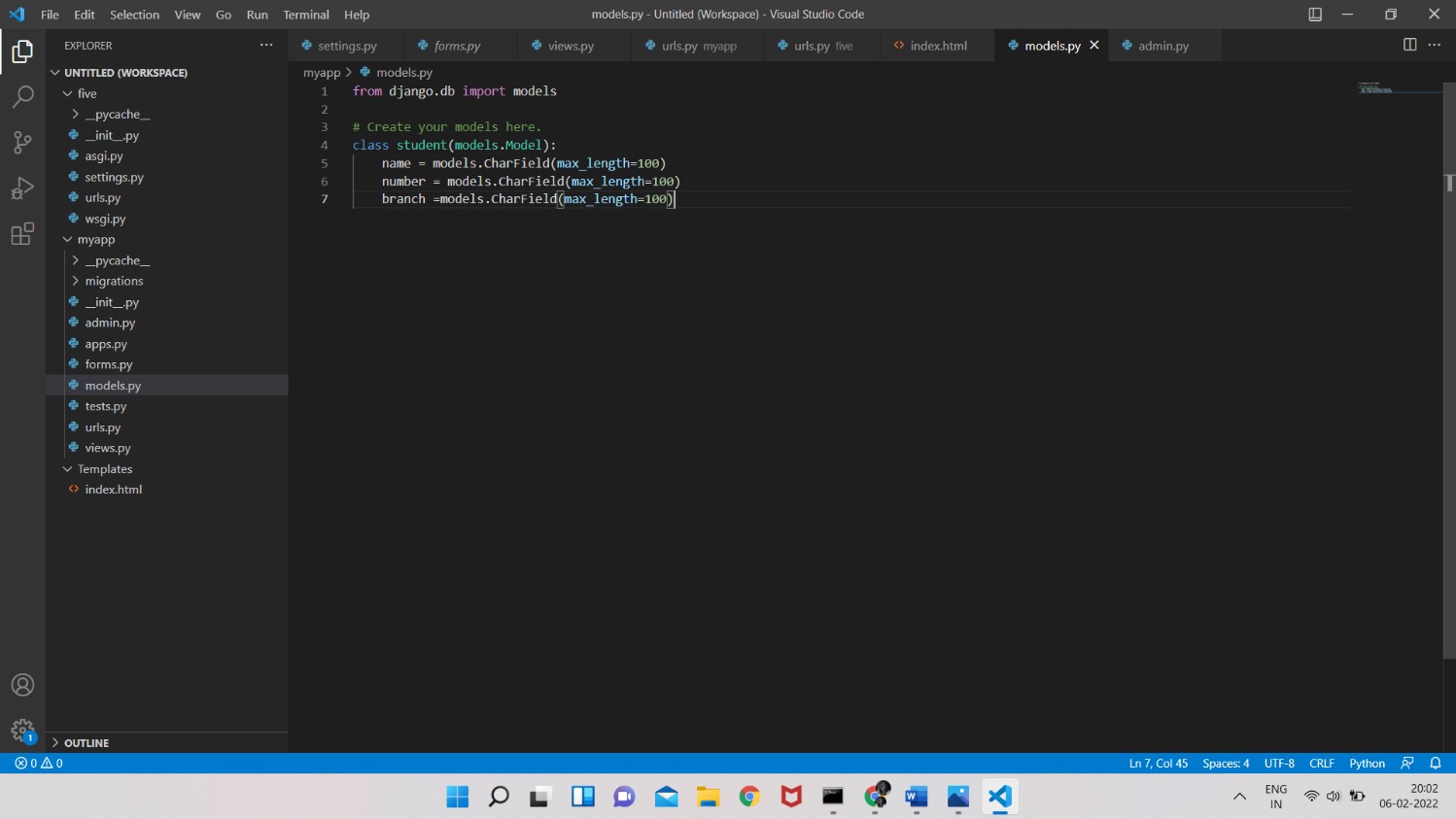
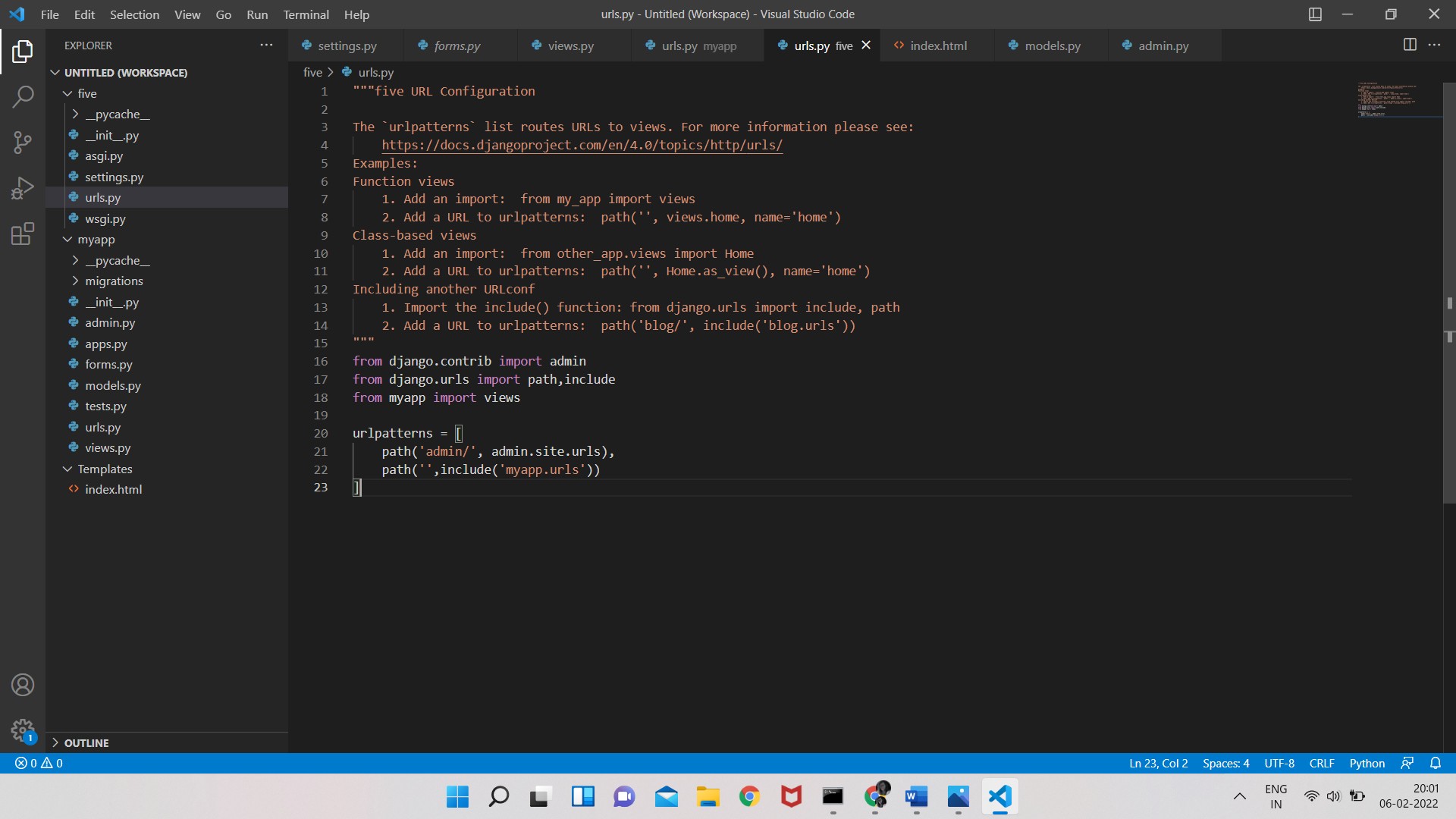
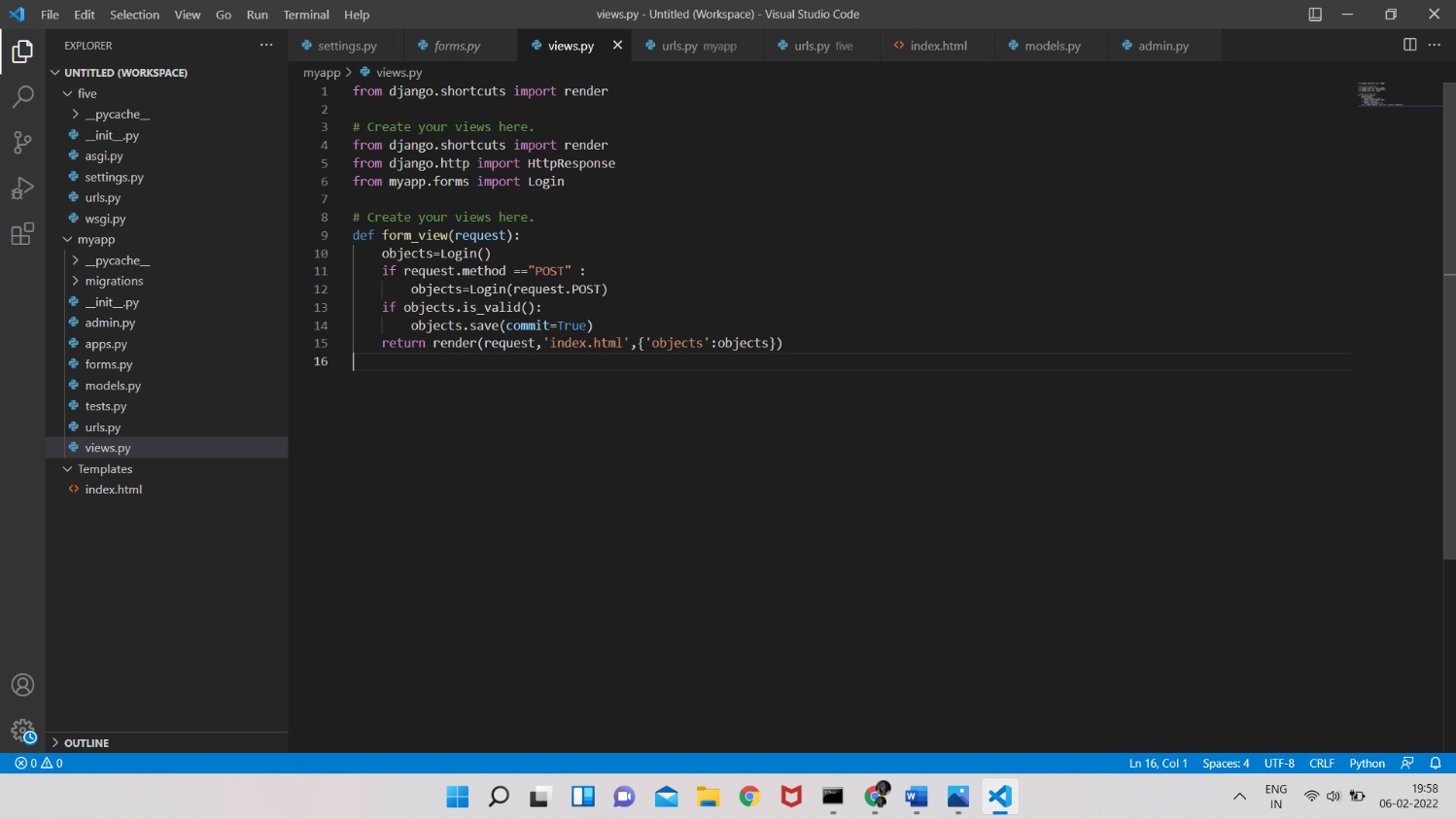
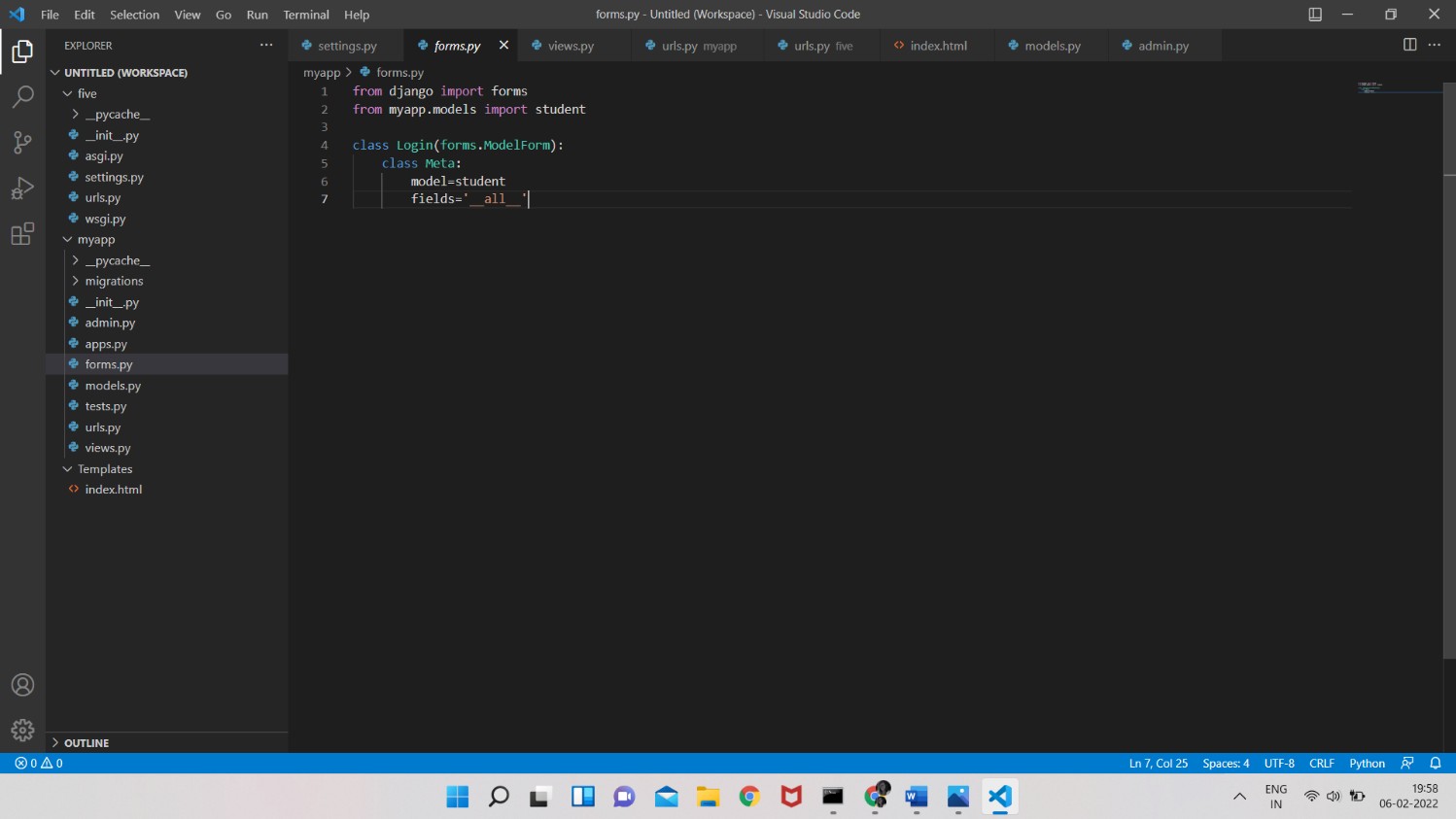
If that file name ex: index.html is not present , then right click on Templates and select New file and name it as index.html. You can do the same for all others that are missing.

Once the code in one page is done ..do save it.(Ctrl+s)

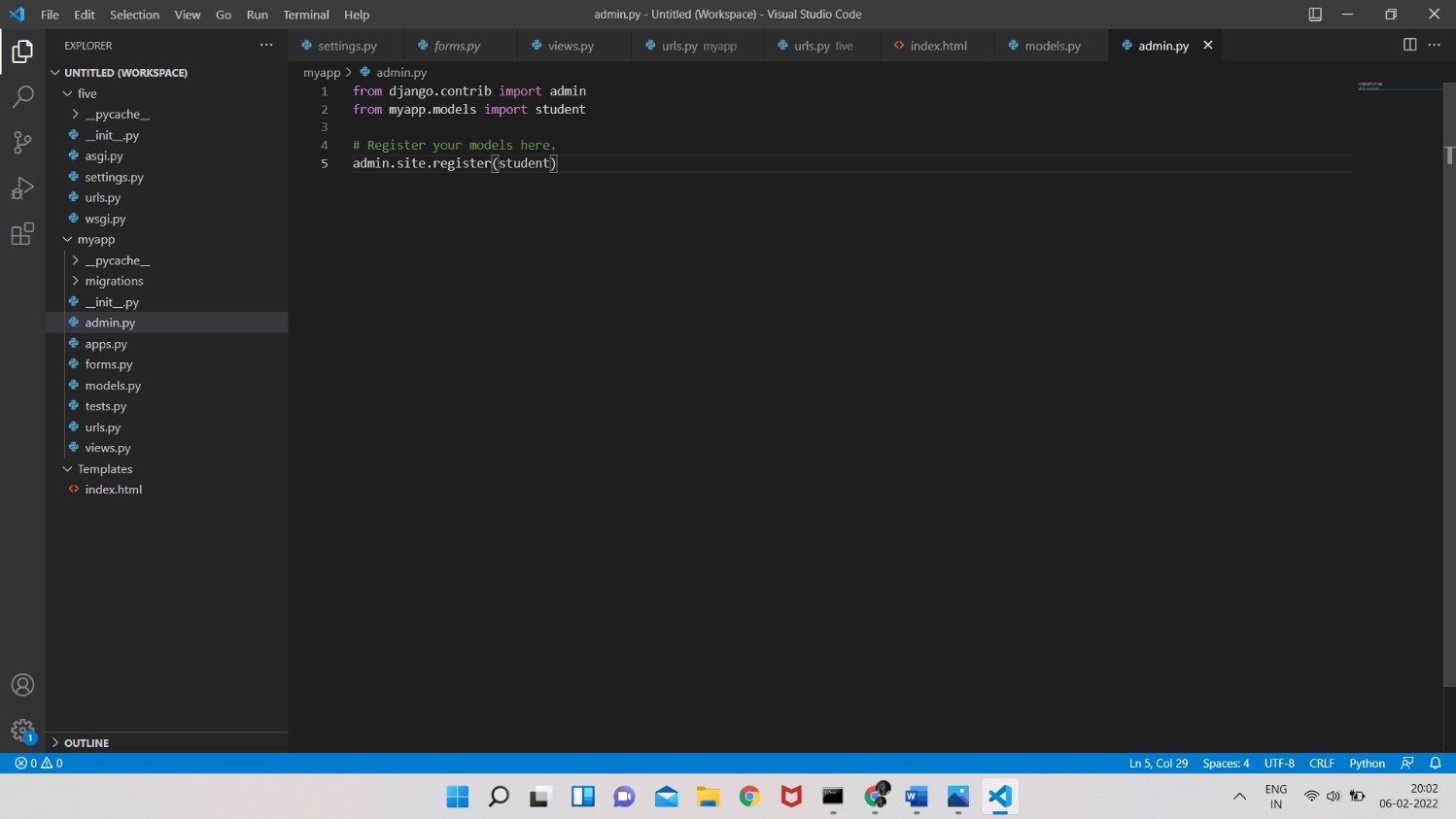
**Step : 1 install django in command prompt**

**Step : 2 create an environment**

**Step : 3 open visual studio , folder five (as per me ) in visual studio ,files will be uploaded Type the following codes.**



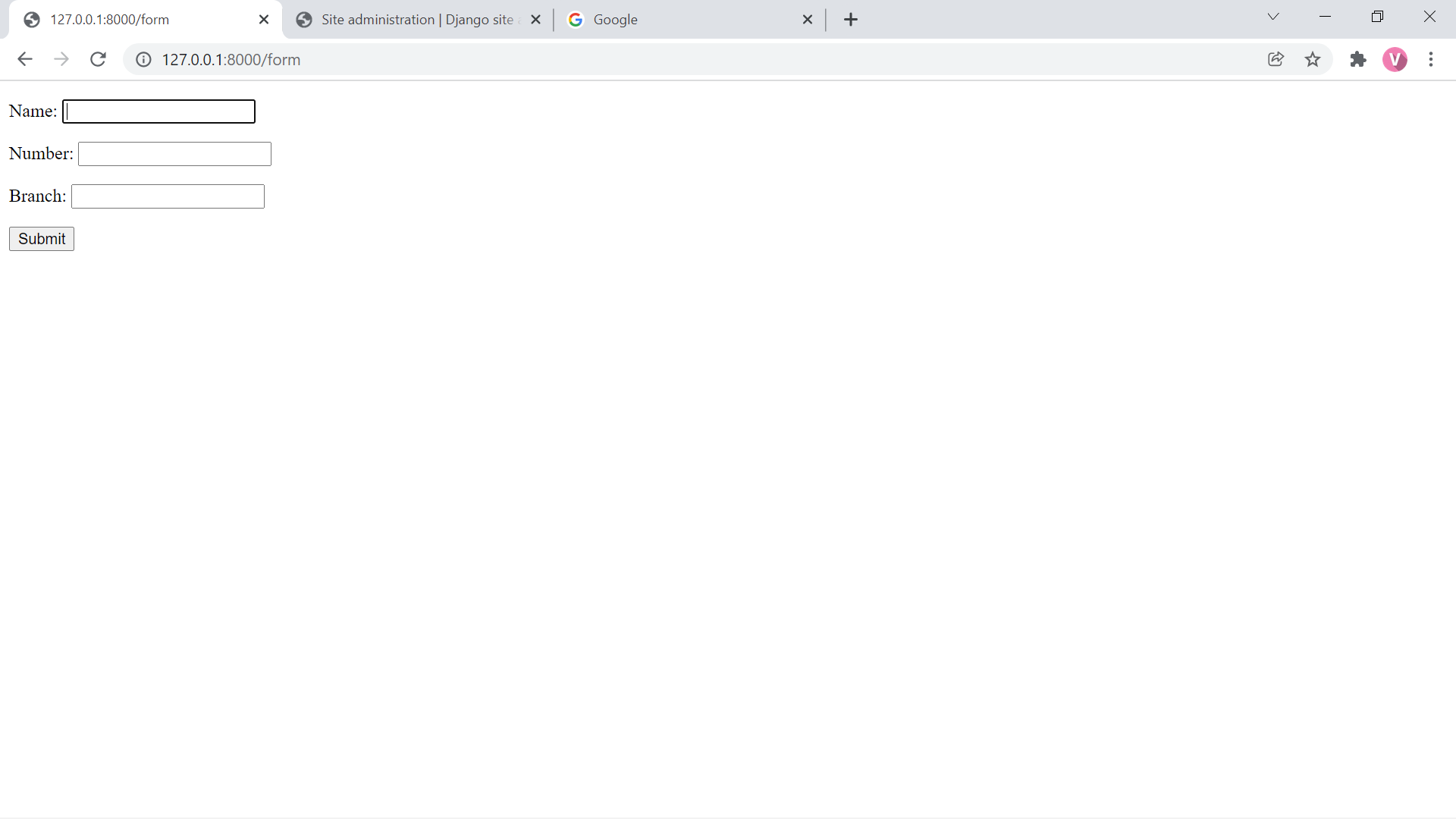
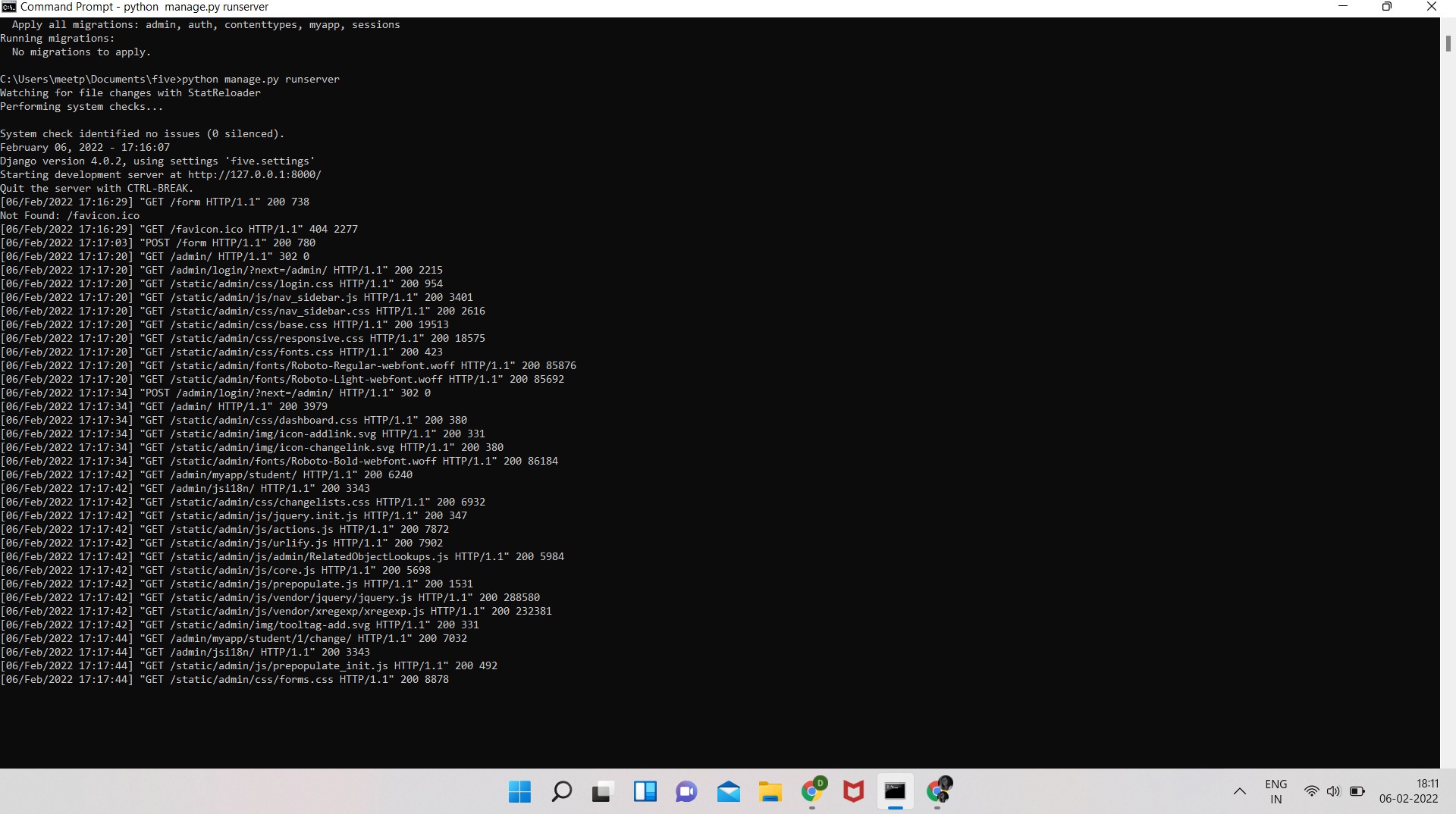
Step : 4 For execution of the form go to command prompt type: python manage.py makemigrations



thereafter type: python manage.py migrate after type:python manage.py runserver

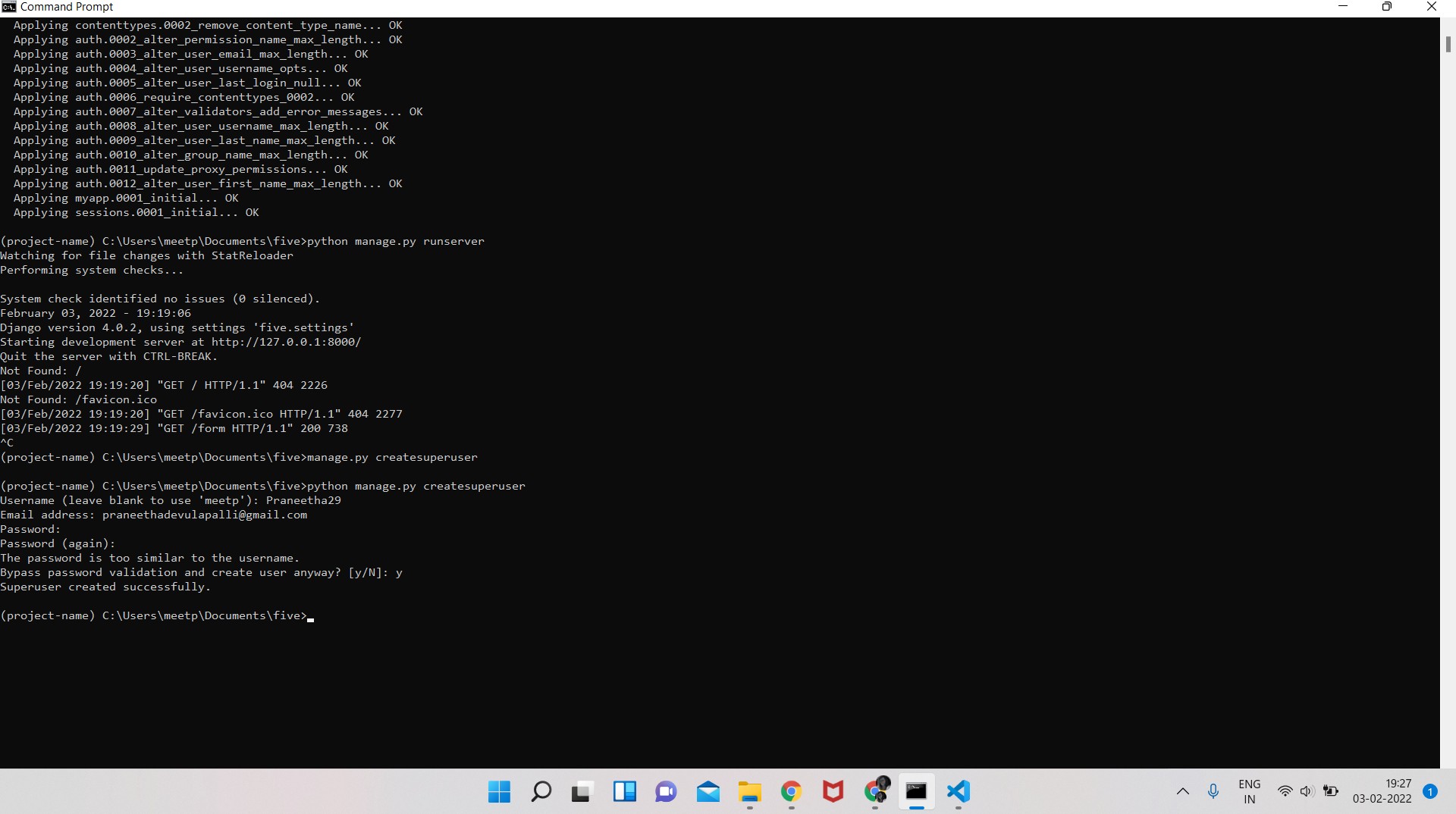
we will get an ip address

copy that and paste it in chrome add : /form to the address http://127.0.0.1:8000/form



Step : 5 We will get a form like this thereafter create a user id for dijango

by typing in cmd propmt : python manage.py createsuperuser



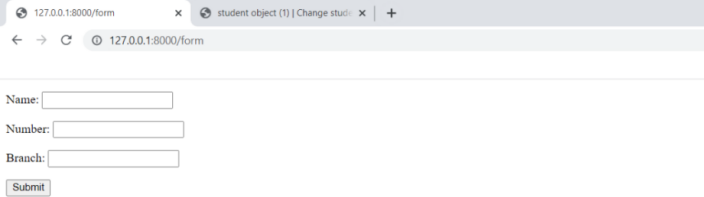
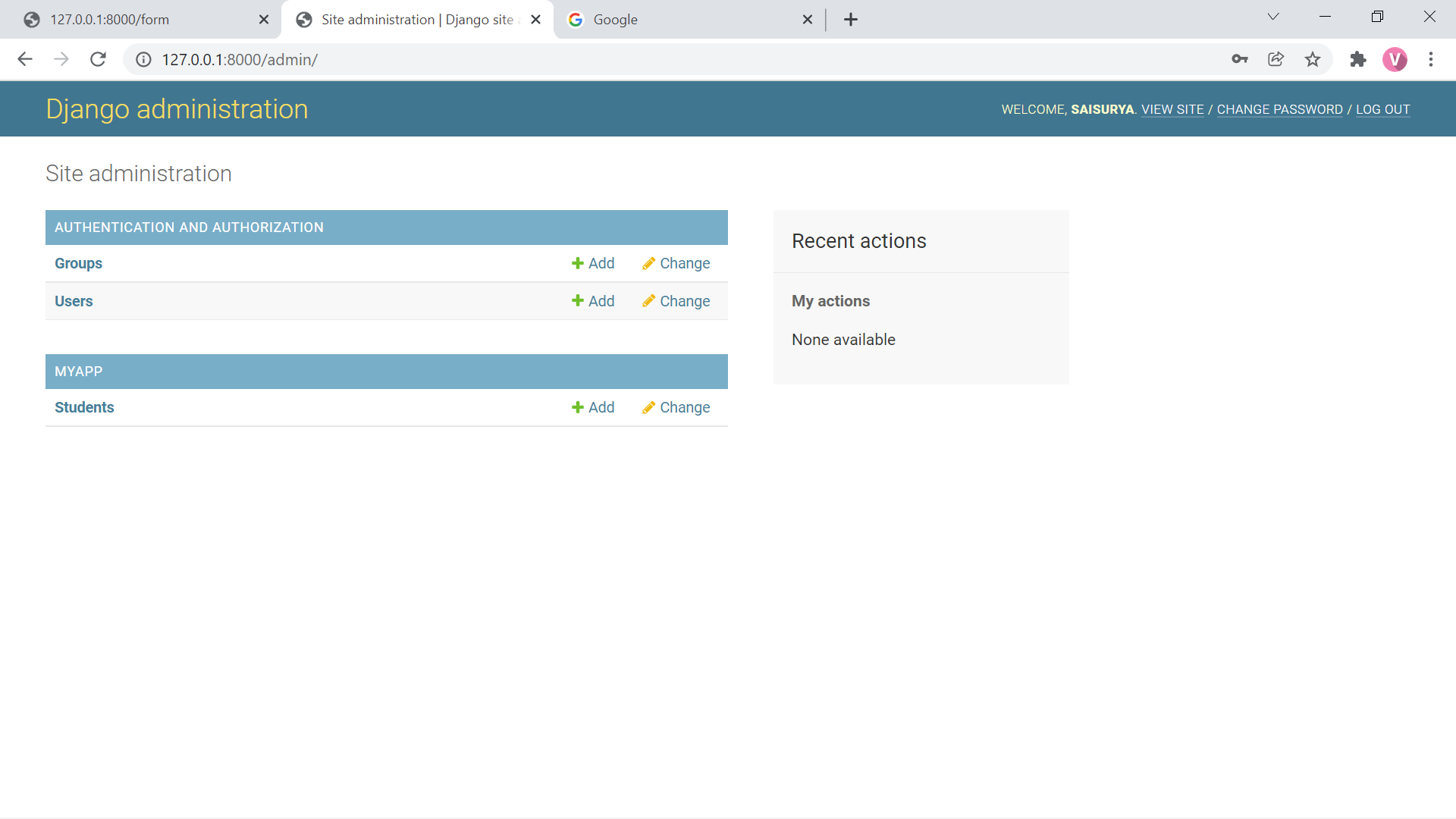
Step : 6 http://127.0.0.1:8000/admin with/admin

again go to chrome and paste the address

will get a site consist of username and passowed will get an Django administration site

login

in student option we can cretae as many as student objects and we can delete too



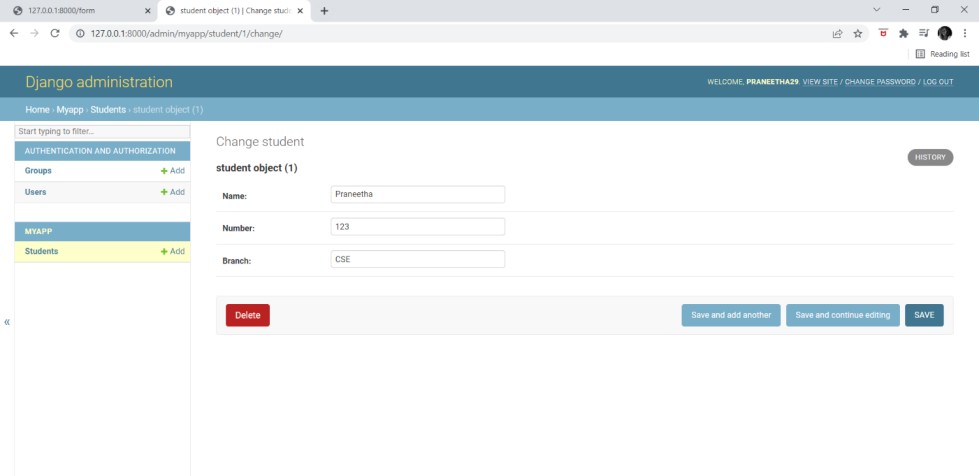
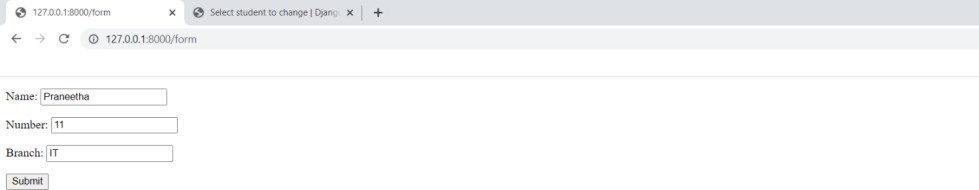
### WEEK-11

1. CRUD operations on the form.

CRUD Meaning: CRUD is an acronym that comes from the world of computer programming and refers to the four functions that are considered necessary to implement a persistent storage application: **create, read, update and delete.**

|  |  |  |
| --- | --- | --- |
| **Letter** | **Operation** | **Insert** |
| C | Create | Insert |
| R | Read | Select |
| U | Update | Edit |
| D | Delete | Delete |

**CREATE:**

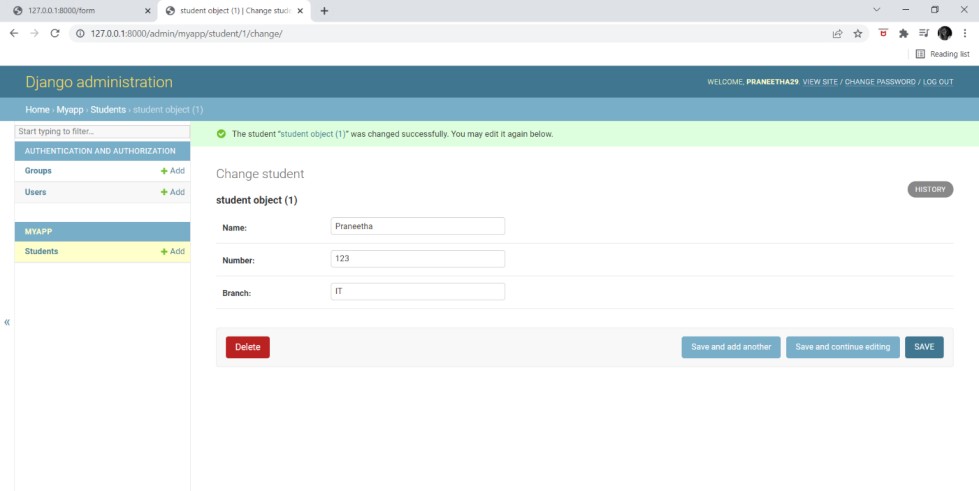
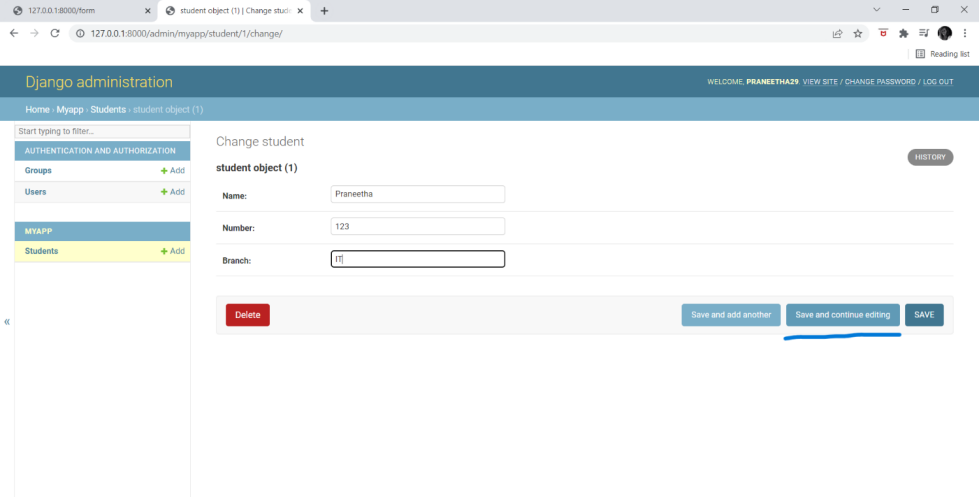


**READ:**

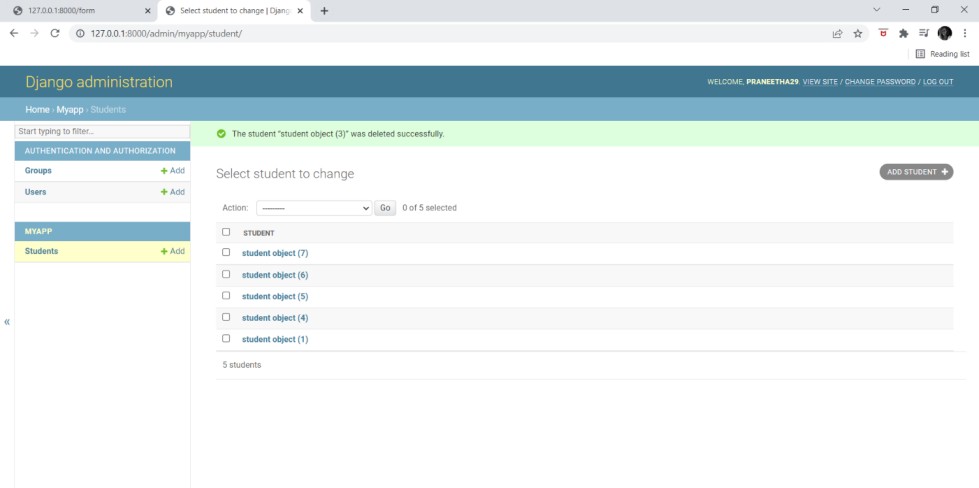
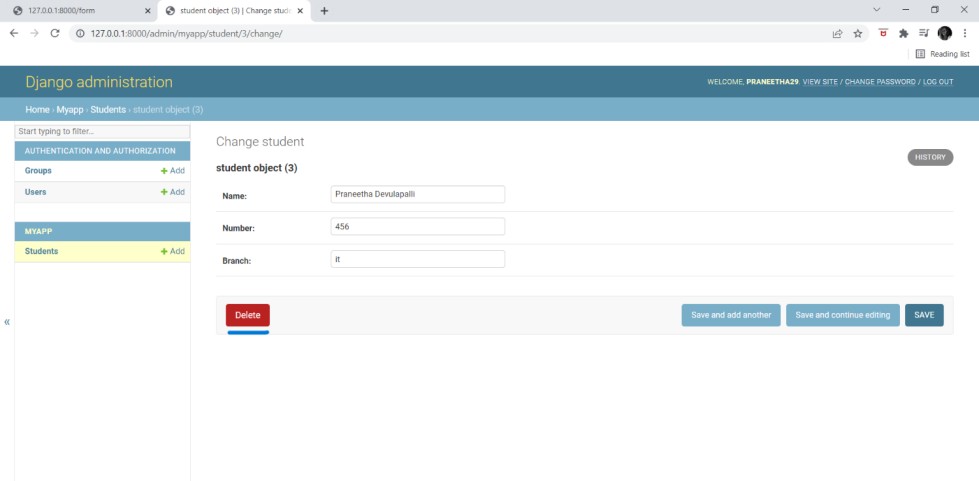
**Update:**

Step 1: Select and open one of the objects created.(say student object (1))

Step 2: Select the Save and continue editing button to update the object**.**



Step 3: The change is updated successfully.



**DELETE:**

Step 1: Select the student object to delete. (say student object(3))

Step 2: The student “student object(3)” was deleted successfully.