Module – 5 (DB and Python Framework)

Q 1.) Why Django should be used for web-development? Explain how you can create a project in Django?

- Django is a high-level Python web framework that enables rapid development of secure and maintainable websites. Built by experienced developers, Django 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 is free and open source, has a thriving and active community, great documentation, and many options for free and paid-for support.
    - **Complete**
        - Django follows the "Batteries included" philosophy and provides almost everything developers might want to do "out of the box". Because everything you need is part of the one "product", it all works seamlessly together, follows consistent design principles, and has extensive and up-to-date documentation.
    - **Versatile**
        - Django can be (and has been) used to build almost any type of website — from content management systems and wikis, through to social networks and news sites. It can work with any client-side framework, and can deliver content in almost any format (including HTML, RSS feeds, JSON, and XML).
    - **Secure**

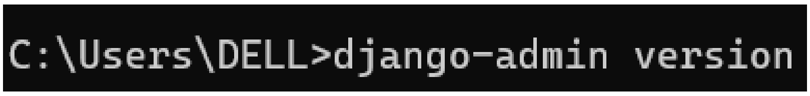
    - Django helps developers avoid many common security mistakes by providing a framework that has been engineered to “do the right things” to protect the website automatically. For example, Django provides a secure way to manage user accounts and passwords, avoiding common mistakes like putting session information in cookies where it is vulnerable (instead cookies just contain a key, and the actual data is stored in the database) or directly storing passwords rather than a password hash.

- **Scalable**
    - Django uses a component-based “shared-nothing” architecture (each part of the architecture is independent of the others, and can hence be replaced or changed if needed). Having a clear separation between the different parts means that it can scale for increased traffic by adding hardware at any level: caching servers, database servers, or application servers. Some of the busiest sites have successfully scaled Django to meet their demands (e.g. Instagram and Disqus, to name just two).
- **Maintainable**
    - Django code is written using design principles and patterns that encourage the creation of maintainable and reusable code. In particular, it makes use of the Don’t Repeat Yourself (DRY) principle so there is no unnecessary duplication, reducing the amount of code. Django also promotes the grouping of related functionality into reusable “applications” and, at a lower level, groups related code into modules (along the lines of the Model View Controller (MVC) pattern).

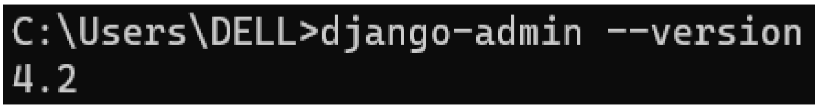
- **Portable**
  - Django is written in Python, which runs on many platforms. That means that you are not tied to any particular server platform, and can run your applications on many flavors of Linux, Windows, and macOS. Furthermore, Django is well-supported by many web hosting providers, who often provide specific infrastructure and documentation for hosting Django sites.

- After creating virtual environment for version control and activating it in the folder for development of project, the commands to be typed in CMD for creating new project is
  - CMD: **cd myenv** #(going into the virtual environment setup folder)
  - CMD: **cd Scripts** #(opening the script folder inside myenv)
  - CMD: **activate** #(activating the virtual environment)
  - CMD: **cd .. X 2** #(returning to the root folder where new project to be started)
  - CMD: **django-admin startproject "Projectname"**

Q 2.) How to check installed version of django?

- 

```
C:\Users\DELL>django-admin version
```

- 

```
C:\Users\DELL>django-admin --version
4.2
```

- Current version available

Q 3.) Explain what does django-admin.py make messages command is used for?

- django-admin is Django’s command-line utility for administrative tasks. This document outlines all it can do.
- In addition, manage.py is automatically created in each Django project. It does the same thing as django-admin but also sets the DJANGO_SETTINGS_MODULE environment variable so that it points to your project’s settings.py file.
- The django-admin script should be on your system path if you installed Django via pip. If it’s not in your path, ensure you have your virtual environment activated.
- Generally, when working on a single Django project, it’s easier to use manage.py than django-admin. If you need to switch between multiple Django settings files, use Django-admin with DJANGO_SETTINGS_MODULE or the --settings command line option.
- The command-line examples throughout this document use django-admin to be consistent, but any example can use manage.py or python -m django just as well.

Q 4.) What is Django URLs?make program to create django urls.

- Django URL. URL is a path through which a specific web-based application and one particular page in that web application can be reached. So for any web-oriented application setting these url paths is a very key necessity.

```
from django.urls import path
from .models import views
urlpatterns = [
path(‘members/’, views.members, name=’members’),]
```

Q 5.) What is a QuerySet?Write program to create a new Post object in database:

- A QuerySet is a collection of data from a database.
- A QuerySet is built up as a list of objects.
- QuerySets makes it easier to get the data you actually need, by allowing you to filter and order the data at an early stage.

Q 6.) Mention what command line can be used to load data into Django?

- To load data into Django you have to use the command line Django-admin.py loaddata. The command line will searches the data and loads the contents of the named fixtures into the database.

Q 7.) Explain what does django-admin.py make messages command is used for?

- Manage.py in Django is a command-line utility that works similar to the django-admin command. The difference is that it points towards the project's settings.py file. This manage.py utility provides various commands that you must have while working with Django. Some of the most commonly used commands are –
  - python manage.py startapp
  - python manage.py makemigrations
  - python manage.py migrate
  - python manage.py runserver