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?

Ans: Django is widely used in web development due to its high-level, Python-based framework that emphasizes rapid development, scalability, and a clean design.

How to Create a Django Project

Step-1: Install virtualenv: pip install virtualenv

Step-2: Create a Virtual Environment: virtualenv myenv

Step-3: Activate the Virtual Environment: myenv/Scripts\activate

Step-4: Install Django: pip install Django

Step-5: : Create a New Django Project: django-admin startproject myproject

Step-6: Navigate to the Project Directory: cd myproject

Step-7: code .

Step-8: python manage.py startapp myapp

Step-9: Go to myproject/settings.py and add your app to the INSTALLED_APPS list:

INSTALLED_APPS = [

  ...

  'myapp',

]

Step-10: Create Database Migrations:

python manage.py makemigrations

python manage.py migrate

Step-11: Create a Superuser for Admin Access: python manage.py createsuperuser

Step-12: Run the Server and Access Admin Panel: python manage.py runserver

Q-2 How to check installed version of django?

Ans: Using django-admin

If Django is installed and the virtual environment is activated (if you're using one), you can check the version with this command:

django-admin –version

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

Ans: In Django, URLs are used to route web requests to the appropriate views in your application. The Django URL dispatcher lets you define patterns that map URLs to views, allowing you to create clean, readable URLs for your website.

How Django URLs Work

- **URL Configuration (URLconf):** Each Django project has a main urls.py file (typically located in the project folder) and can have additional urls.py files within each app.

- **Patterns:** Django URLs are defined using regular expressions or path converters, making it easy to create dynamic URLs that accept parameters.

- **Views:** Each URL pattern points to a specific view function or class, which handles the request and returns a response.

myapp/urls.py

from django.urls import path

from myapp import views

urlpatterns = [

path('', views.home, name='home'), # URL for the home page

path('about/', views.about, name='about'), # URL for the about page

]

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

Ans: In Django, a QuerySet is a collection of database queries that retrieve data from the database in a way that's optimized and easy to work with. QuerySets allow you to filter, sort, and retrieve records from the database, and they represent the collection of database rows that meet specific conditions.

Example Program to Create a New Post Object in the Database

Assuming we have a Django model Post defined in models.py, we can create a new object and save it to the database.

Define the Post Model

First, define a simple Post model with fields like title, content, and created_at.

myapp/models.py

from django.db import models

class Post(models.Model):

title = models.CharField(max_length=200)

content = models.TextField()

created_at = models.DateTimeField(auto_now_add=True)

Step 2: Run Migrations to Create the Post Table in the Database

python manage.py makemigrations

python manage.py migrate

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

Ans: python manage.py loaddata <fixture_file>

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

Ans: The django-admin.py makemessages command in Django is used to create translation message files for your Django project or application. It is part of Django's internationalization (i18n) framework, which allows you to create multilingual websites by translating strings into different languages.

How It Works

The command scans your project for all strings marked for translation using special syntax like:

- _("string to translate") in Python code

- {% trans "string to translate" %} or {% blocktrans %} in templates

These strings are collected and added to .po files, which are Portable Object files used for translations.

Common Syntax

django-admin makemessages -l <language_code>