



## Django Framework

### Section 1

---

- **HTTP protocol**
- **MVC model**
- **Virtual Environment**
- **Django Structure**
- **Generic Views**
- **HTML templates**
- **URL dispatcher**
- **ORM**
- **Custom Views**
- **Get and Post Methods**
- **URL shortener**
- **User model**
- **Logic in templates**
- **Querying models**
- **Serving Static files**
- **Deployment of Django**
- **Automating tasks with Django**
- **Generic Views in Django**
- **Rest Serializers**

1. Create a basic Django project that displays “Hello, World!” on the homepage..
2. Create a blog application with Django where users can create, read, update, and delete blog posts.
3. Implement user authentication and registration functionality in a Django project.
4. Develop a URL shortner application using Django that generates short URLs for long URLs.
5. Create a user management system in Django that allows administrators to manage user accounts.
6. Build an e-commerce website using Django that allows users to browse and purchase products.
7. Implement a task tracking application in Django that allows users to create and manage tasks.
8. Develop a social media platform using Django that allows users to post, follow others, and like posts.
9. Create a photo gallery application in Django that allows users to upload and display photos.
10. Build a job board website using Django that allows employers to post job listings and job seekers to apply.

- 
1. Implement a news aggregator application in Django that fetches and displays news articles from different sources.
  2. Develop a weather forecast application using Django that retrieves weather data from an API and displays forecasts.
  3. Create a polling app in Django that allows users to create and participate in polls.

4. Build a recipe sharing platform using Django that allows users to share and discover recipes.
5. Implement a file storage application in Django that allows users to upload, download, and manage files.
6. Develop a forum website using Django where users can create threads, post replies, and interact with each other.
7. Create a book marketing application in Django that allows users to save and categorize bookmarks.
8. Build a chat application using Django and WebSocket that enables real – time communication between users.
9. Implement a video streaming platform in Django that allows users to upload, view, and share videos.
10. Develop a blog API using Django Rest Framework that provides CRUD functionality for blog posts and comments.

## **Section 2 – Database Connectivity**

- Introduction to MySQL, Database, Tables, Rows, columns
- Introduction to SQL commands , DDL and DML

\*1 create a program to access the data of an employee from database table.

\*2 create a program to insert , update and delete the record of an employee.

\*3 create a program to manage all the CRUD operations through a python class.

\*4 create a program to increase and access the salary of a particular employee.

\*5 Create a program to read the data from a csv file and enter into the database table.

### **Lab Practice :**

#1 create a program to fetch the data of all the students whose course is MCA from a database table.

#2 Create a python class to manage all the CRUD operations of Student table.

#3 Create a program to fetch the data of all the students and put into a csv file.