

# Django

# Syllabus

By: Er. Sujan Thadarai

Email: [sujanthadarai710@gmail.com](mailto:sujanthadarai710@gmail.com)

Phone: 9816107823

## Course Description

This course offers an in-depth introduction to Django, a powerful and popular Python-based web framework. Designed for aspiring web developers, it covers the fundamentals of building dynamic, database-driven web applications. Students will learn how to set up Django projects, create models, views, templates, and manage databases. By the end of the course, participants will have the skills to develop and deploy robust web applications with Django, focusing on best practices for security, scalability, and maintainability.

## Prerequisites

Before joining this course, students need to be familiar with

- HTML
- CSS
- Bootstrap
- Python
- SQL



## Course Contents

• Introduction to Django .....	3
• Django Architecture .....	3
• Routing .....	3
• Django Views .....	3
• Django Templates System .....	4
• Interacting With Database .....	4
• Django Admin Dashboard .....	4
• Built-in Django Apps and Web Development Concerns .....	5
• Django Sessions .....	5
• Django Class Based Views .....	5
• Rest Web Service .....	6



## Introduction to Django

- Overview of web development
- Django framework introduction
- Installation of Django
- Creating a virtual environment for better dependency management
- Create your first sample Django project
- Understanding the project structure of a newly created django project

## Django Architecture

- Django Project MVT Architecture and MVC
- Django Models and Databases
- Django views
- Templates system

## Routing

- Understanding urls.py and the purpose of it
- Creating URLs with names and namespaces
- Mapping of the URLs with the view functions
- Linking the URLs of an app in a project with the root app of the project using 'include'

## Django Views

- Django views : Function base view & class base view
- Django CRUD operation using function
- What is Render and relative



## Django Templates System

- Basic Using Template System
- Need for templates
- How to Configure Template
- Template Loading
- Using conditionals and looping in
- django templates
- Django convention of storing the static assets like images, css and javascript and referencing them from the template

## Interacting With Database

- Django ORM and benefit of ORM
- Django model Field and field types and option fields
- Relationship between django models - One to One,
- Many to One, Many to Many
- Django migrations
- Saving, updating, deleting, and querying django model

## Django Admin Dashboard

- Activating the admin interface
- Creating superuser for admin panel
- Customizing django admin templates look and feel (perform CRUD Operations)



## Built-in Django Apps and Web Development Concerns

- Authentication system
- Messages framework
- Emailing system

## Django Sessions

- HTTP as a stateless protocol
- Django session management built in app
- Concept of session management and how it internally
- uses cookies and session id
- Storing attributes in a session and retrieving from
- the same session
- Destroying a session using 'flush'

## Django Class Based Views

- Introduction Class Views and benefits of class views
- Types of class View : class Based view and generic view
- Built in class-based generic view
- Class based CRUD operations: listview, createview, updateview, deleteview, templateview



## Rest Web Service

- Web Services and REST
- Django REST framework
- What is serializers?
- Creating Views
- Creating URLs
- Update and delete records using
- serializer/Django Rest Framework (DRF)
- POSTMAN details and how to check
- APIs in POSTMAN

