

GHAR SE CODE {

Django Syllabus:-

Course Title: Django

Course By: GSC (GHAR SE CODE)

Duration: 35 Hrs

Code Spirit: “Convention Over Configuration”

Module	Topic	Detailed Overview
1	Introduction to Django	- What is Django?- MVT architecture vs MVC- Installing Django- Creating first project (django-admin startproject)- Running development server
2	Django Project Structure	- Overview of default files- settings.py, urls.py, manage.py- Creating apps (startapp)- Registering apps
3	URL Routing & Views	- URL patterns and routing- Writing basic views- Returning HTML/JSON responses- Dynamic URL patterns
4	Templates & Static Files	- Template engine basics- Creating and rendering templates- Template inheritance (extends, block)- Loading static files and CSS
5	Models and ORM	- Defining models and fields- Database migrations (makemigrations, migrate)- Admin interface (admin.site.register)- Querying with ORM (filter, get, exclude)
6	Forms & User Input	- Creating HTML forms- Handling POST/GET data- Django.forms module- Validating user input
7	Admin Customization	- Modifying admin interface- Customizing list display and fields- Admin filters and search

Module	Topic	Detailed Overview
8	User Authentication	- Django's built-in auth system- Login/logout system- Registration and password reset- Access control with decorators (@login_required)
9	Class-Based Views (CBVs)	- Introduction to CBVs- Using ListView, DetailView, CreateView, UpdateView, DeleteView- CBV vs FBV (Function-Based Views)
10	Middleware and Request Handling	- Understanding Django request/response cycle- Writing custom middleware- Common built-in middleware
11	REST API with Django REST Framework (DRF)	- Installing DRF- Serializers, ViewSets, Routers- Creating simple API endpoints
12	Deployment Essentials	- Preparing project for production- Debug vs Production settings- Hosting options: Heroku, Railway, PythonAnywhere- Collecting static files, security settings
13	Testing in Django	- Writing unit tests- Using Django's TestCase- Client testing with the test client
14	Project Work / Capstone	- Build a blog, task manager, or e-commerce prototype- Use templates, models, auth, forms, and deployment- Code review and improvement

}