



Django

Syllabus with

MYSQL INTEGRATION



1: Introduction to Django

- What is Django?
- Features of Django
- MVT (Model-View-Template) Architecture
- Installing Django
- Django Project Structure
- Creating a Django Project
- Running the development server

2: Setting Up Environment

- Install Python and Django
- Create virtual environment using venv
- Install MySQL
- Install MySQL client for Django:
pip install mysqlclient or pip install django-mysql
- Configure settings.py for MySQL

3: Django Apps & Admin Interface

- Creating Django Apps: python manage.py startapp appname
- Registering App in settings.py
- Exploring Django Admin Panel
- Creating Superuser
- Customizing Admin Panel

4: Models (ORM & MySQL)

- Defining Models in models.py
- Field types (CharField, IntegerField, DateField, etc.)
- Primary key and foreign key
- Relationships: One-to-One, One-to-Many, Many-to-Many
- Meta options (ordering, verbose_name, etc.)
- Model methods
- Run Migrations: makemigrations, migrate
- Raw SQL Queries in Django

5: Querying Database

- Using Django ORM: `all()`, `filter()`, `exclude()`, `get()`
- `order_by()`, `values()`, `annotate()`, `aggregate()`
- Q objects and complex lookups
- Join Queries (`ForeignKey`, `related_name`)
- `select_related` vs `prefetch_related`
- Pagination

6: Views & URL Routing

- Function-based Views
- Class-based Views (CBVs)
- `HttpResponse` and `JsonResponse`
- URL dispatcher and path converters
- `urls.py` structure for apps and project
- Named URLs and `reverse()`

7: Templates

- Template directory structure
- Django Template Language (DTL)
- Template variables, filters, tags
- Template inheritance
- Using static files (CSS, JS)
- Using dynamic data in templates

8: Forms in Django

- Creating HTML forms
- Django Form class
- `ModelForm`
- Validation (clean methods)
- CSRF protection
- Form rendering: `as_p`, `as_table`, `as_ul`
- Handling GET and POST requests
- File Upload using `FileField` or `ImageField`



9: User Authentication & Authorization

- Built-in User model
- Login, Logout, Registration
- @login_required decorator
- Permissions and Groups
- Custom User Model
- Password hashing and reset
- Session and Cookies

10: Middleware & Context Processors

- What is Middleware?
- Built-in vs custom middleware
- Common Middleware use cases (e.g., logging, authentication)
- Context Processors for global variables in templates

11: Email Functionality

- Sending email using SMTP (e.g. Gmail)
- Configuring EMAIL_BACKEND in settings.py
- Sending HTML or plain-text emails
- Sending with attachments

12: API Integration

- Consuming external APIs using requests
 - Using Django to build REST APIs: Django REST Framework (DRF)
 - Serializers, ViewSets, Routers, Authentication (Token, JWT)
 - CRUD APIs with DRF
- 



13: Testing

- Writing unit tests
- Testing views, models, forms
- Using Django's test client
- Coverage report



14: Deployment

- Static and media file management (collectstatic)
- Using gunicorn/uWSGI with Nginx
- Deploy on platforms: PythonAnywhere, Heroku, AWS EC2
- Use of .env file for secrets and settings



15: Django with MySQL Optimization

- Indexing tables in MySQL
- Using .only(), .defer(), and select_related() properly
- Profiling queries using Django Debug Toolbar
- Query optimization tips



16: Advanced Topics

- Signals (pre_save, post_save)
 - Caching (File-based, Memcached, Redis)
 - Cron Jobs using celery or django-cron
 - WebSockets with Django Channels
 - Multi-database setup
 - Writing reusable Django apps
- 



✉ alok@webmyne.com

☎ +91 94276 02525

📍 702, Ivory Terrace , Opp. Circuit House
R.C. Dutt Road, Vadodara-07
Gujarat - India.

