

1. What is Django and what kind of framework is it?
2. What are the main benefits of using Django over other frameworks?
3. Compare Django and Flask. Name three key-differences.
4. What does "batteries-included" mean in Django?
5. How does Django follow the MVC/MVI architecture?
6. `pip install Django`
7. `django-admin startproject projectname`
8. `python manage.py runserver`
9. URL is used to access a running Django project by default (R..)
10. How do you verify if Django is installed and working correctly?

⑤ Day 4: Django Templates (Q10-15)

1. What is a view in Django?
2. What are function-based views (FBVs)?
3. How do you map a URL to a view in Django?
4. Which Django function is used to return plain text responses?
5. What is the purpose of the `render()` function?
6. What is the proper folder structure for templates in an app?
7. How do you include an app's `urls.py` inside the project's main `urls.py`?
8. How do you pass data from a view to a template?
9. Explain the concept of a context dictionary

⑤ Day 4: Django Templates & Jinja2

10. What is Django's default template engine?

in a Template

9. How do you include an app's `urls.py` inside the project's main `urls.py`?
 10. How do you verify a new model data using the Django shell?
- ⑤ Day 3: Django, URLs, Templates (Q19-29)

⑤ Day 3: Views, URLs, Templates (Q19-29)

1. What is a view in Django?
2. What are function-based views (FBVs)?
3. How do you map a URL to a view in Django?
4. Which Django function is used to return plain text responses?
5. What is the purpose of the `render()` function?
6. What is the proper folder structure for templates in an app?
7. How do you include an app's `urls.py` inside the project's main `urls.py`?
8. How do you pass data from a view to a template?
9. Explain the concept of a context dictionary
10. What's the difference between returning `HttpResponse()` and `render()`?

⑤ Day 4: Django & Database (Q40-45)

1. ORM (Object Relational Mapping) - Supports database interaction using Python
2. Define models as Python classes inheriting from `django.db.models.Model`
3. `CharField`, `TextField`, `BooleanField` and `DecimalField` for string, large text, true and for decimal number storage
4. Migrations created via `makemigrations` and `migrate` consider `makemigrations`, `migrate` (decimal numbers migrate)
5. Migrations created and default `__init__.py` in `migrations` folder?