



# Web programming

## (Django)

Mohammad Nazari

Sharif University of Technology

2023

# Contents

| Django            | Display     | Admin        | Syntax    | QuerySets | Static Files  | PostgreSQL  | Deploy            |
|-------------------|-------------|--------------|-----------|-----------|---------------|-------------|-------------------|
| Introduction      | Prepare     | Admin        | Variables | QuerySet  | Add Static    | Intro       | Elastic Beanstalk |
| Getting Started   | Add Details | Create User  | Tags      | Get Data  | WhiteNoise    | AWS Account | Requirements      |
| Virtual Environme | Add Master  | Models       | If...Else | Filter    | Collect       | RDS         | django.config     |
| Install Django    | Add Main    | List Display | For Loop  | Order By  | Global Static | Connect     | Create .zip       |
| Create Project    | Add 404     | Update       | Comment   |           | Add to Poject | Add Members | Deploy            |
| Create App        | Add Test    | Add          | Include   |           |               |             | Update            |
| Views             |             | Delete       |           |           |               |             |                   |
| Urls              |             |              |           |           |               |             |                   |
| Templates         |             |              |           |           |               |             |                   |
| Models            |             |              |           |           |               |             |                   |
| Insert Data       |             |              |           |           |               |             |                   |
| Update Data       |             |              |           |           |               |             |                   |
| Delete Data       |             |              |           |           |               |             |                   |
| Update Model      |             |              |           |           |               |             |                   |

# What is Django?

- Django is a Python framework that makes it easier to create web sites using Python.
- Django takes care of the difficult stuff so that you can concentrate on building your web applications.
- Django emphasizes reusability of components, also referred to as DRY (Don't Repeat Yourself), and comes with ready-to-use features like login system, database connection and CRUD operations (Create Read Update Delete).

# How does Django Work?

Django follows the MVT design pattern (Model View Template):

- Model - The data you want to present, usually data from a database.
- View - A request handler that returns the relevant template and content - based on the request from the user.
- Template - A text file (like an HTML file) containing the layout of the web page, with logic on how to display the data.

# Model

The model provides data from the database.

- In Django, the data is delivered as an Object Relational Mapping (ORM), which is a technique designed to make it easier to work with databases.
- Django, with ORM, makes it easier to communicate with the database, without having to write complex SQL statements.
- The models are usually located in a file called [models.py](#).

# View

A view is a function or method that takes http requests as arguments, imports the relevant model(s), and finds out what data to send to the template, and returns the final result.

The views are usually located in a file called `views.py`.

# Template

A template is a file where you describe how the result should be represented.

Templates are often .html files, with HTML code describing the layout of a web page, but it can also be in other file formats to present other results, but we will concentrate on .html files.

The templates of an application is located in a folder named `templates`.

```
<h1>My Homepage</h1>  
  
<p>My name is {{ firstname }}.</p>
```

# URLs

Django also provides a way to navigate around the different pages in a website.

When a user requests a URL, Django decides which view it will send it to.

This is done in a file called `urls.py`.