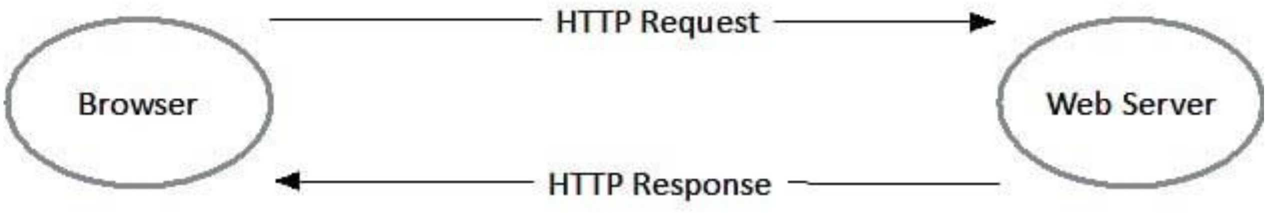
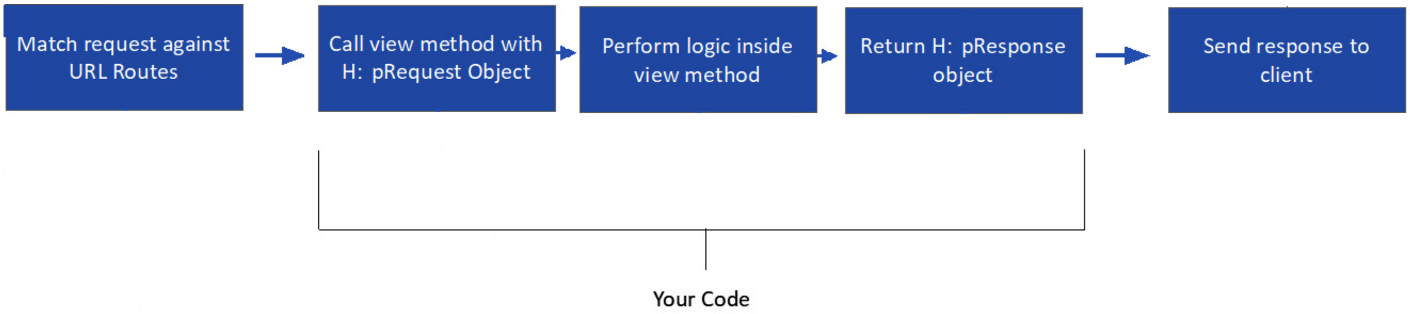
CS471 – Web Technologies (Laboratory)	Qassim University College of Computer	Lab 3 Django and MVT Architecture

This lab session covers the basics of the Django URL and HTML template. By the end of this session, students should have a very good understanding of how to define URL patterns and create aan HTML template and how both work together to Processing an HTTP request. They should also have already practiced with the Git system and how to use basic commands to push and manage their repository.

The following diagram shows the direction of the transmission of HTTP requests and HTTP responses, between a browser and a web server



Inside the framework, the flow of HTTP request and response is illustrated in the following figure. The sections indicated as **Your Code** are for the code that you write, and the first and last steps are taken care of by Django. Django does the URL matching for you, calls your view code, and then handles passing the response back to the client. Actually, you are going to build and configure a link between a URL and a view where the URL matching is taking place by Django.

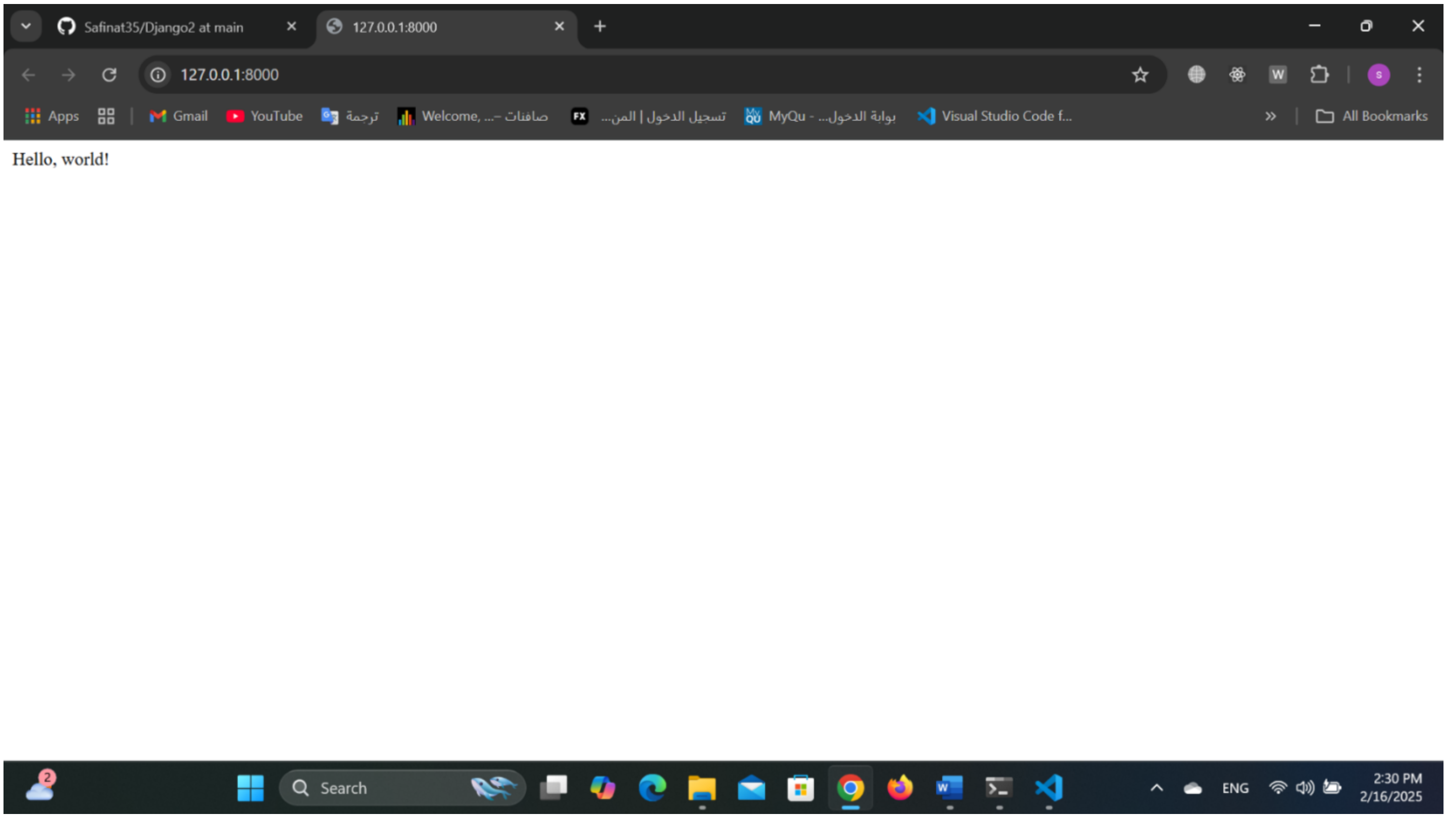

Match request against URL Routes
Call view method with H: pRequest Object
Perform logic inside view method
Return H: pResponse object
Send response to client
Your Code

Pre-lab Preparation:

1. Complete the previous lab (Lab3): must have the initial Django project with apps (bookmodule & usermodule) and the necessaries configurations.
2. Good understanding of HTTP protocol, mainly the request, response, and HTTP status codes.
3. Basic knowledge about HTTP tags.

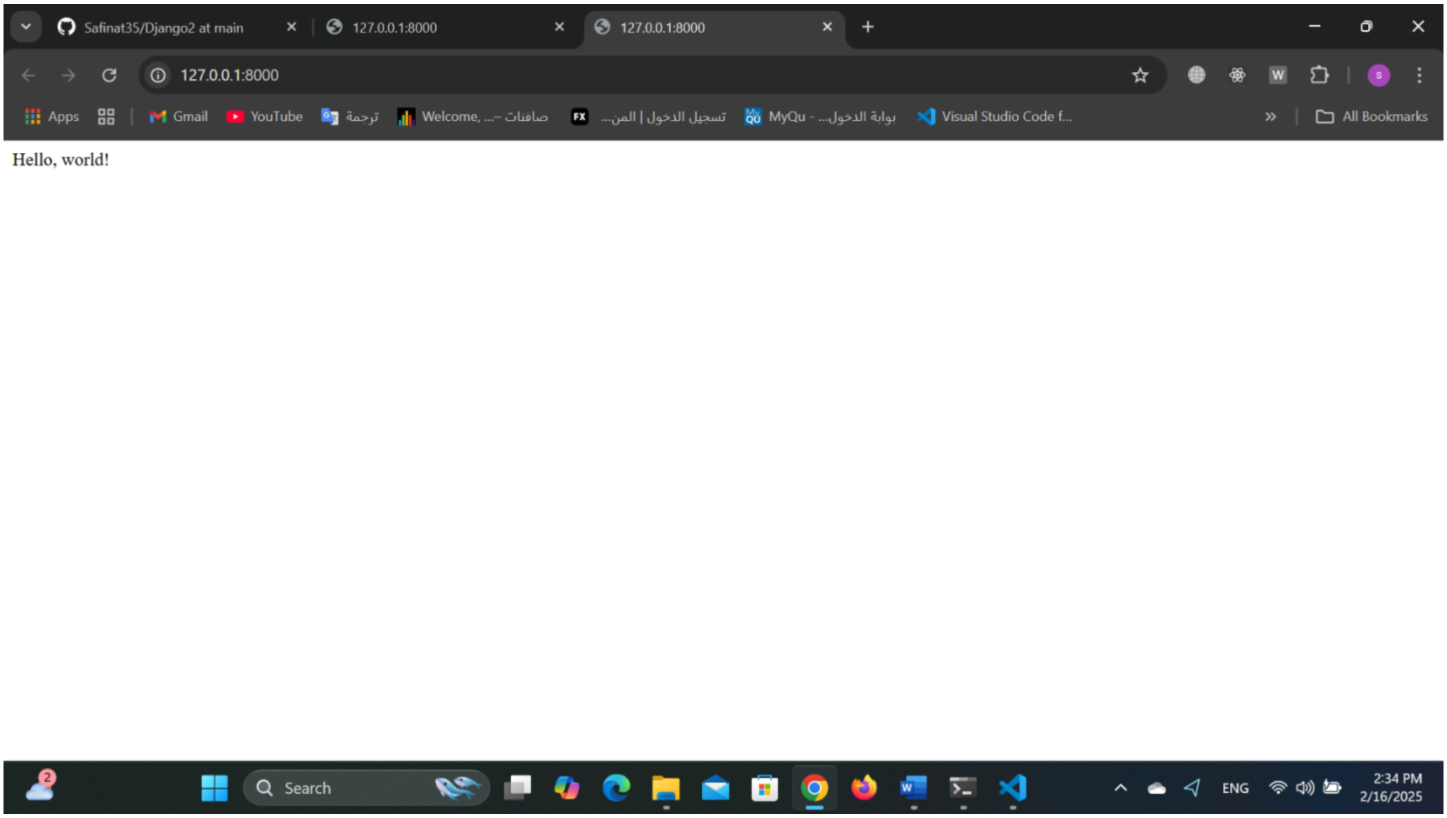
Lab Activities: Build and configure a simple link between a URL and a view (simple pattern), along with a simple HTML template.

Task 1: Build your first view function and corresponding URL mapping in the core/urls.py

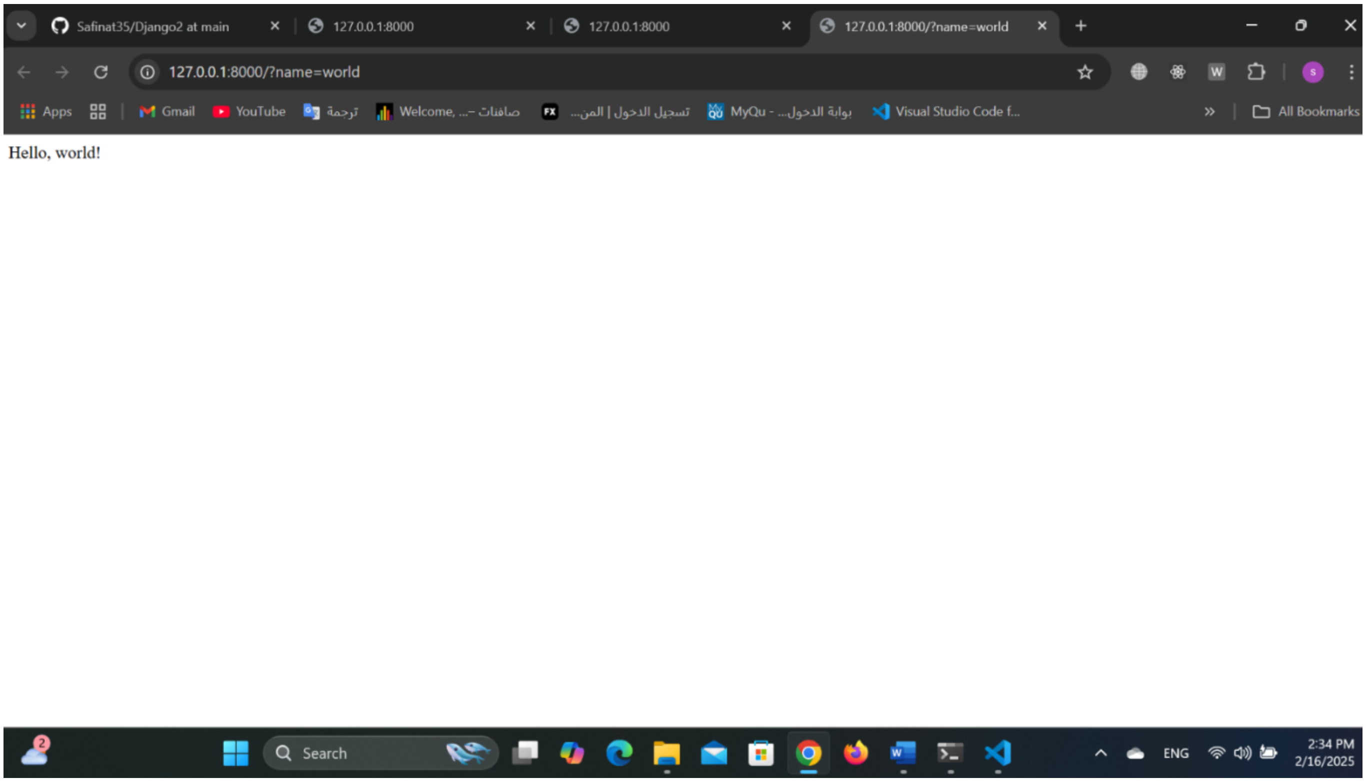


Hello, world!

Task 2: Add parameters with HTTP requests[1]

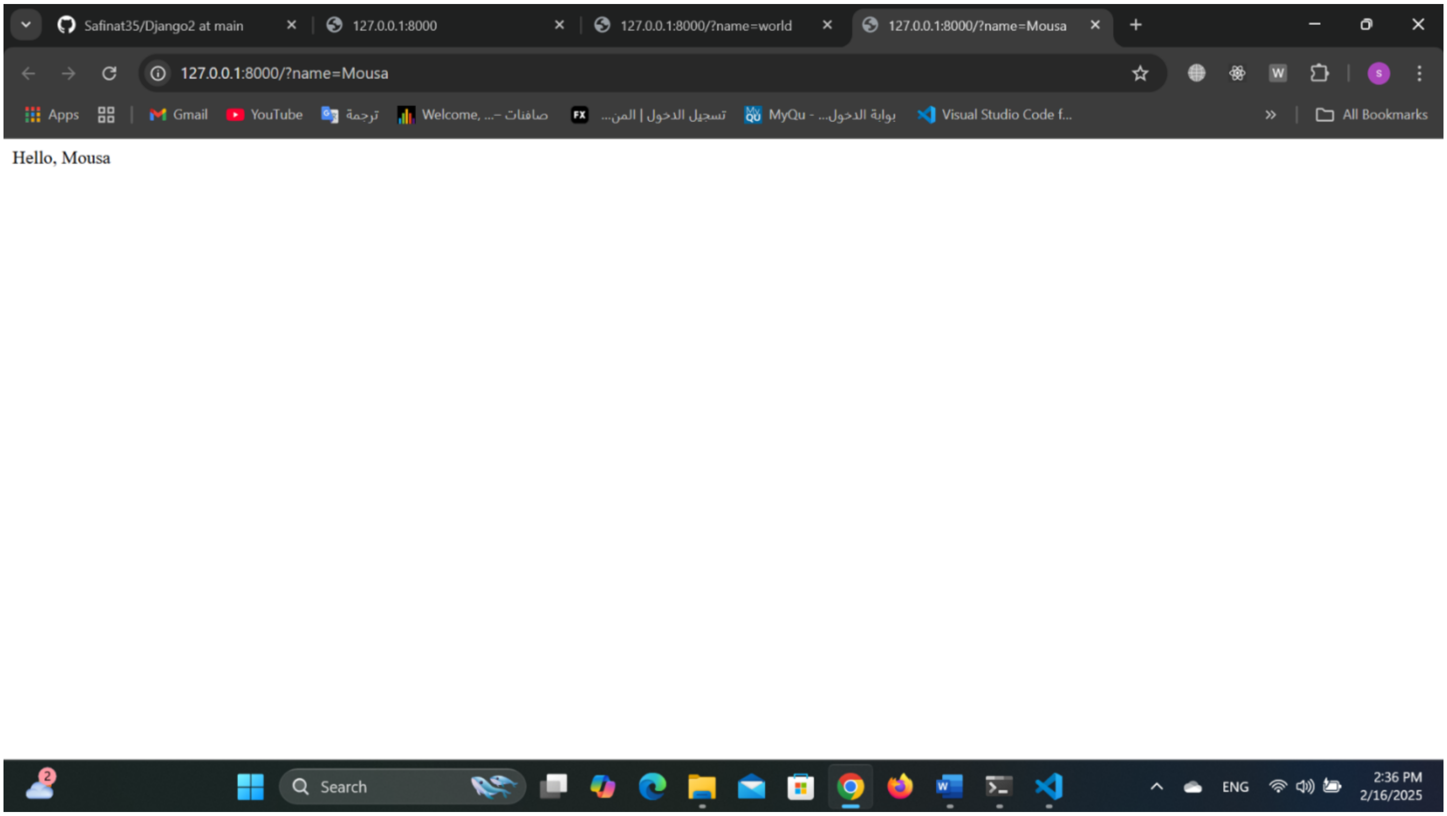


Hello, world!



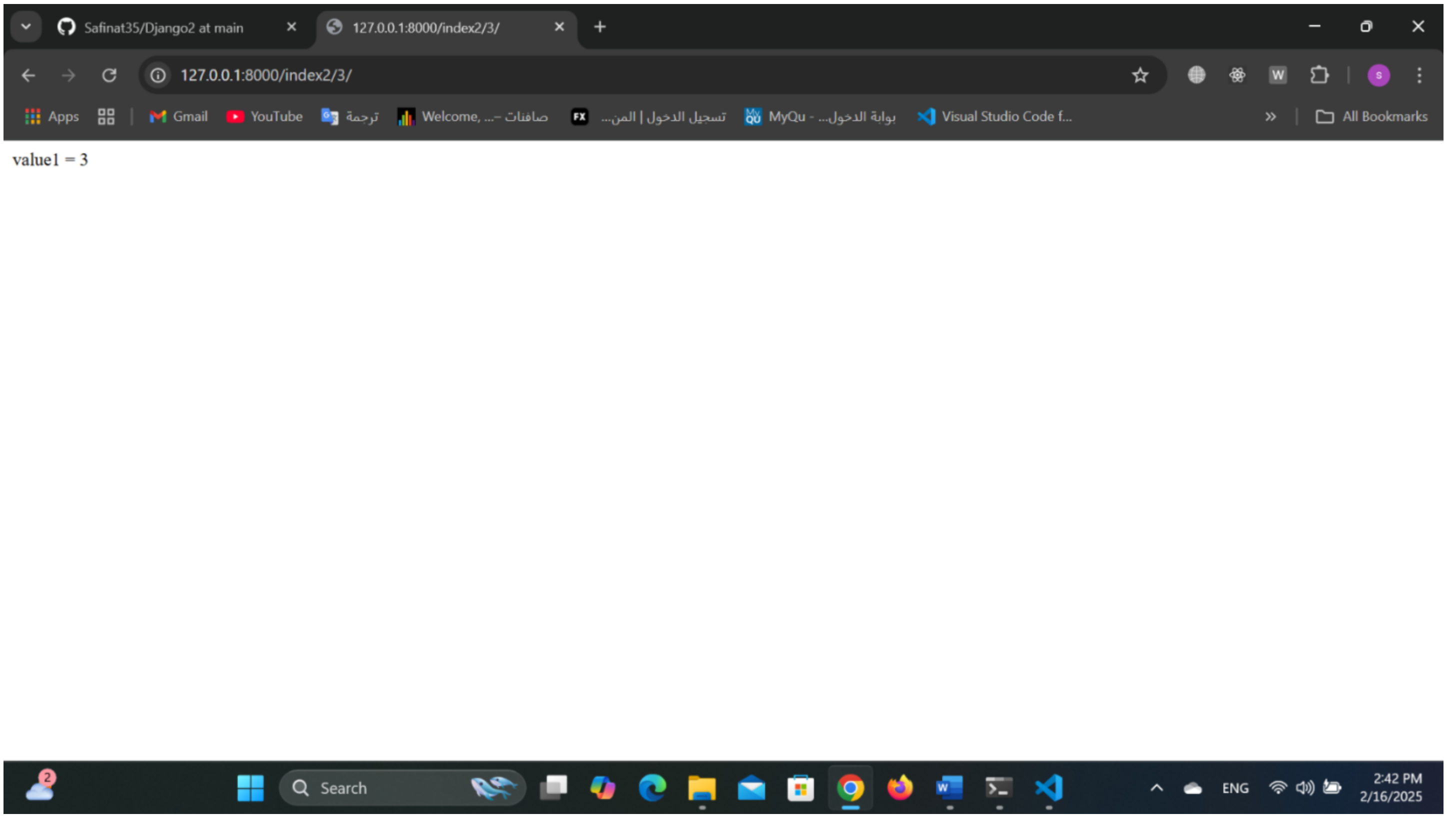
Hello, world!

[1] It is possible to have more than one parameter, like http://127.0.0.1:8000?id=10&name=world!&age=18
Note: this method is GET HTTP request, which is not recommended for security reasons, and use POST HTTP request instead.

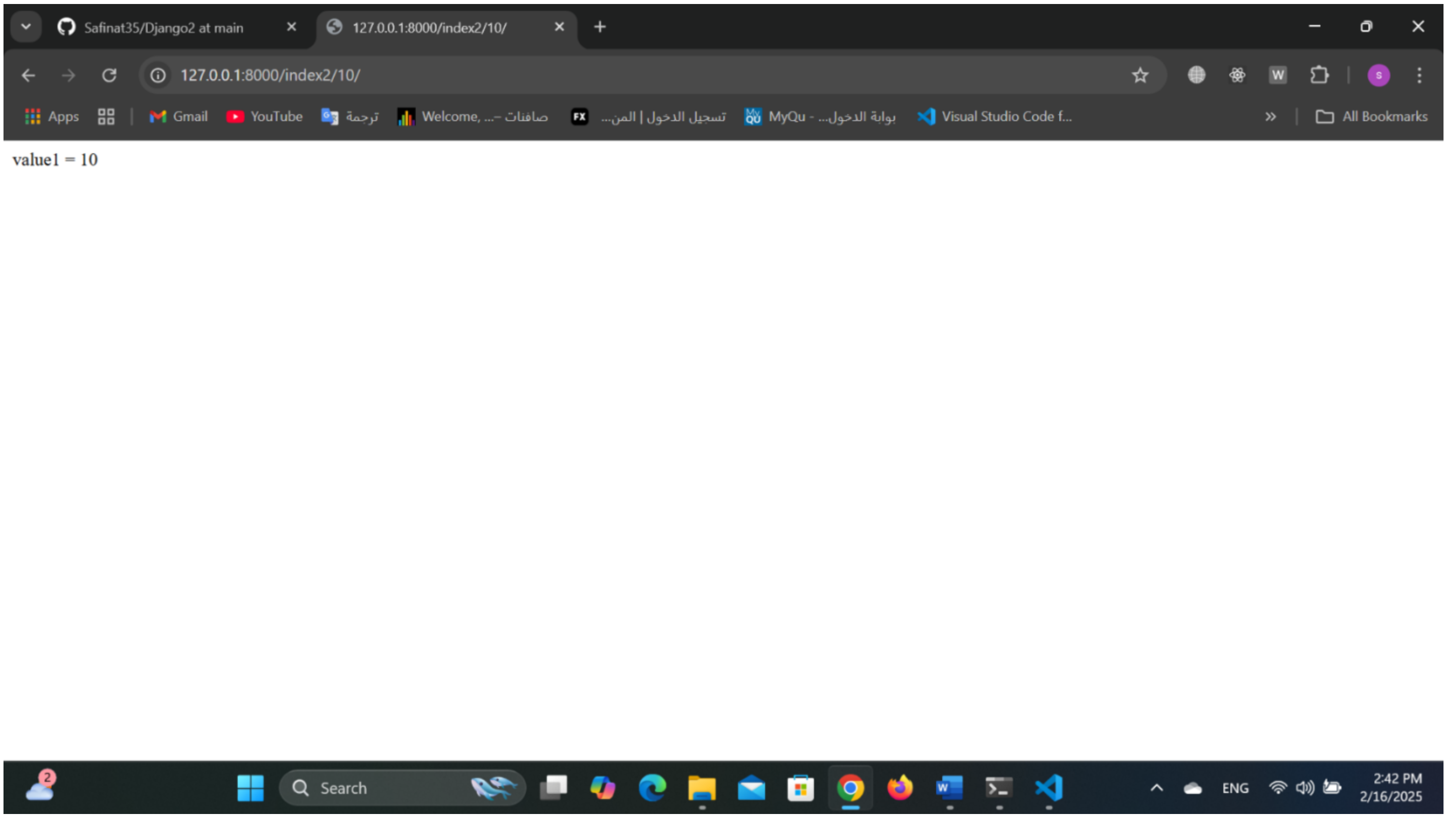

Safinat35/Django2 at main
127.0.0.1:8000
127.0.0.1:8000/?name=world
127.0.0.1:8000/?name=Mousa
127.0.0.1:8000/?name=Mousa
Apps
Gmail
YouTube
Welcome, ...
Visual Studio Code f...
All Bookmarks
Search
ENG
2:36 PM
2/16/2025

```
Hello, Mousa
```

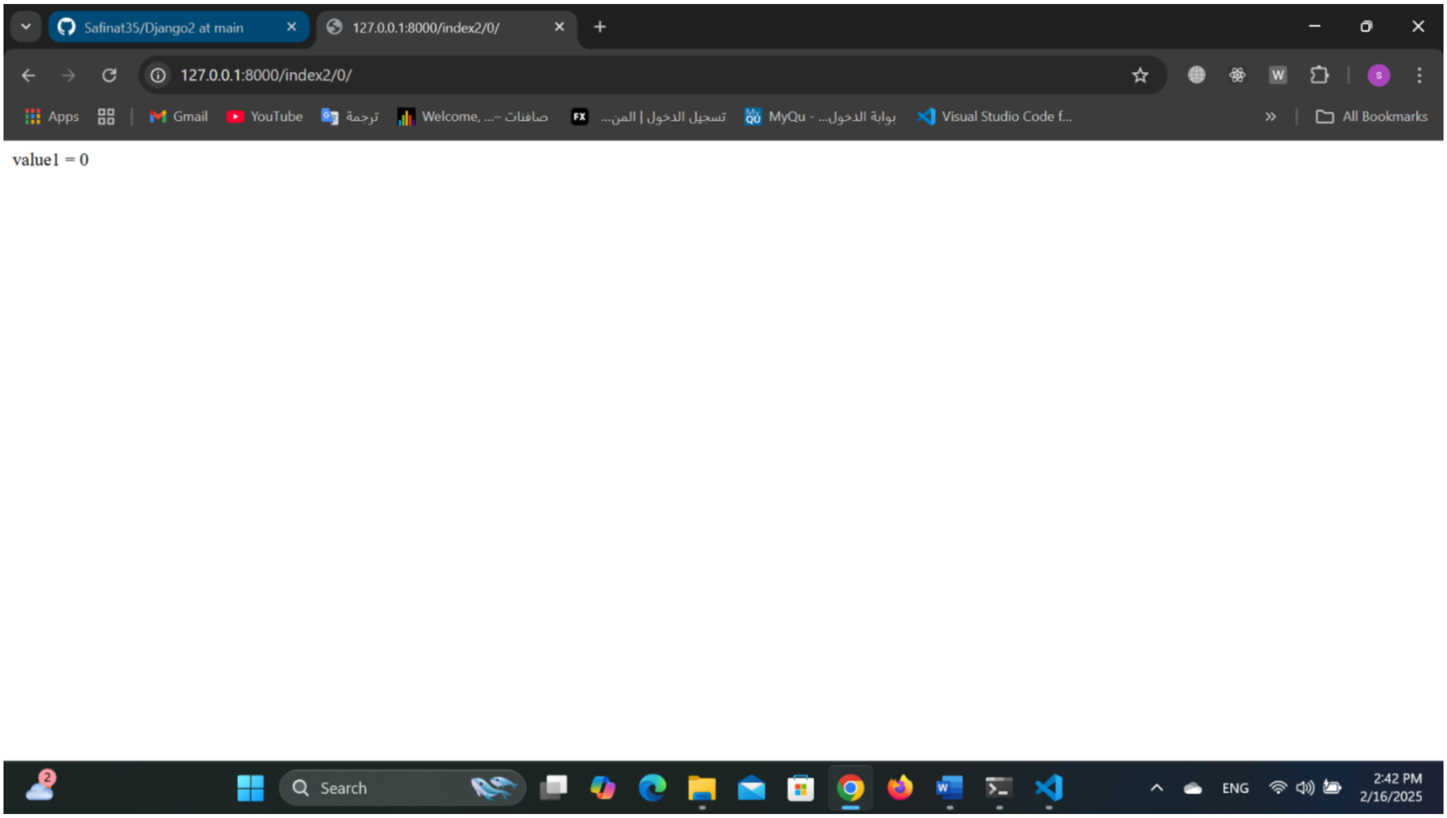
Task 3: Build your second view function and corresponding URL mapping with parameters within URL path



value1 = 3

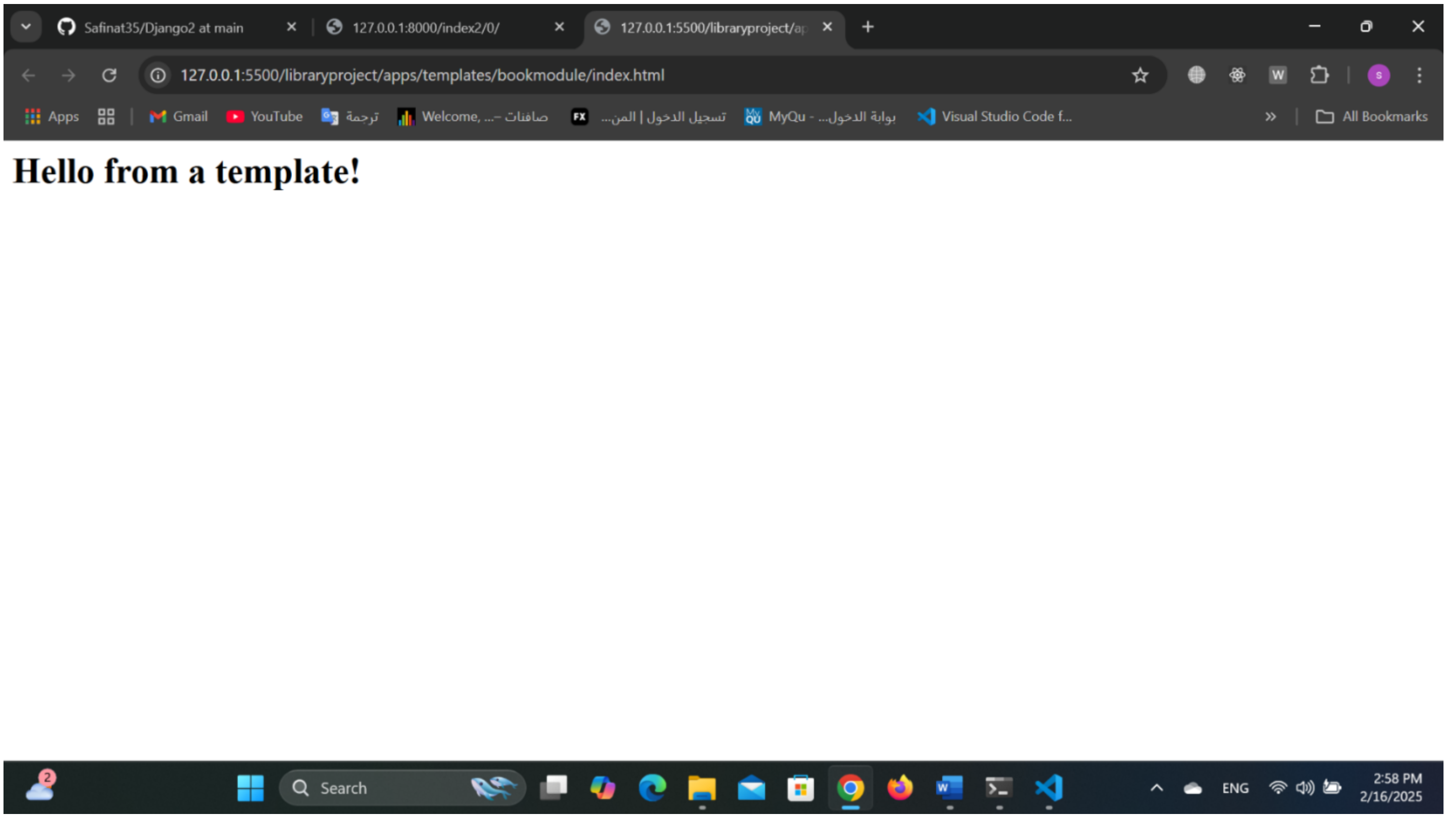


value1 = 10



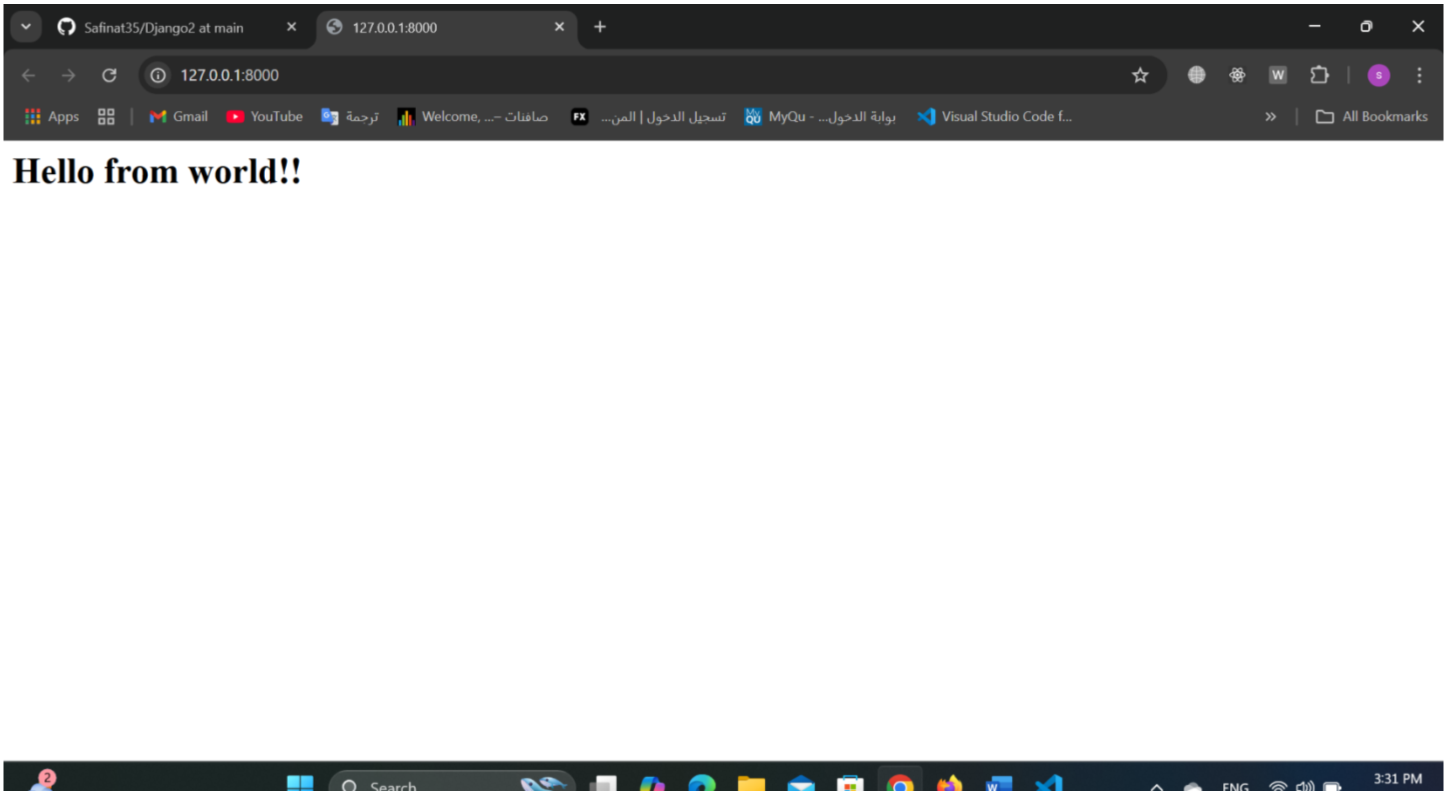
```
value1 = 0
```

Task 4: Create a simple HTML template



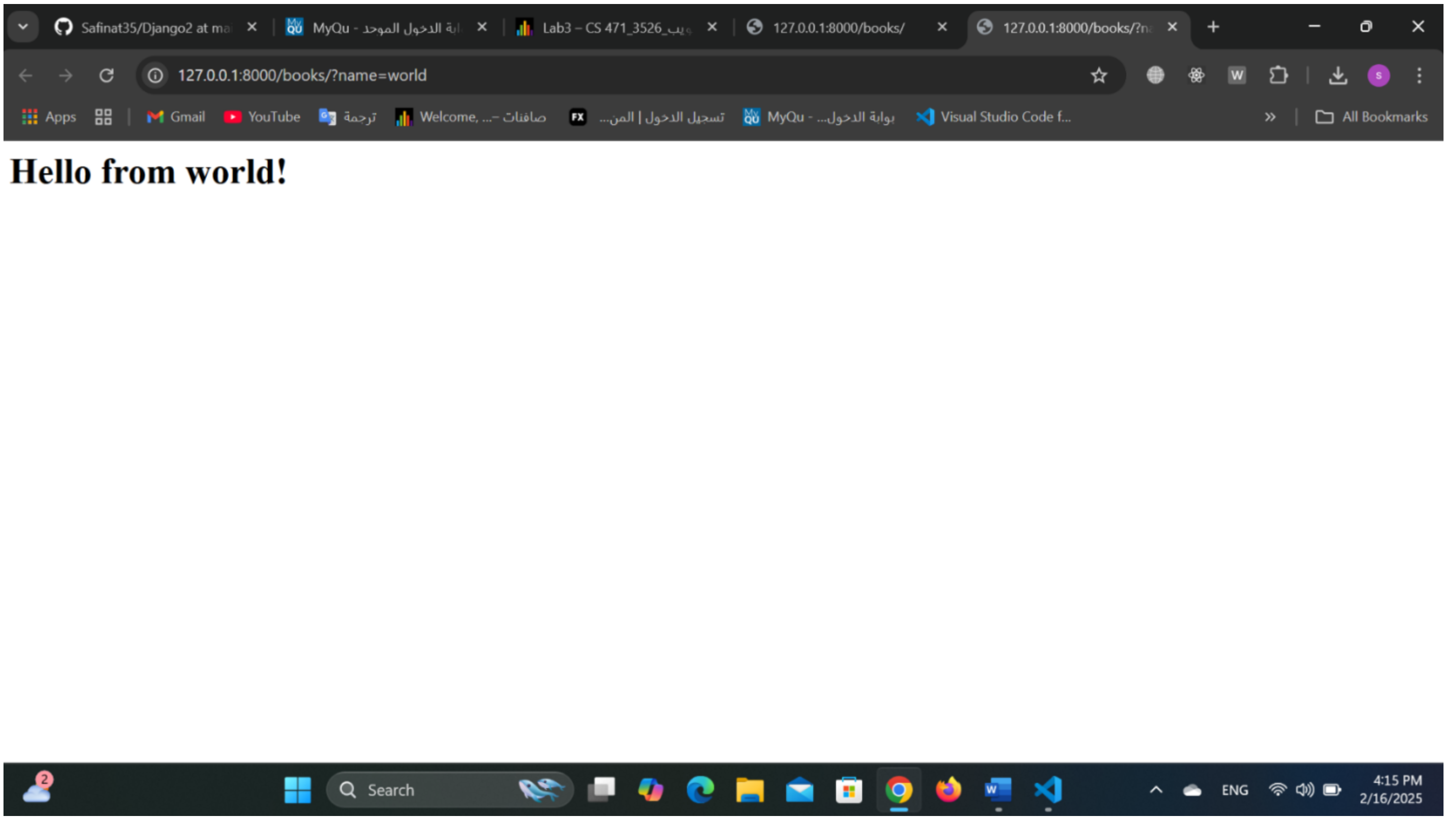
```
Hello from a template!
```

Task 5: Rendering variables in the HTML template that processes a context

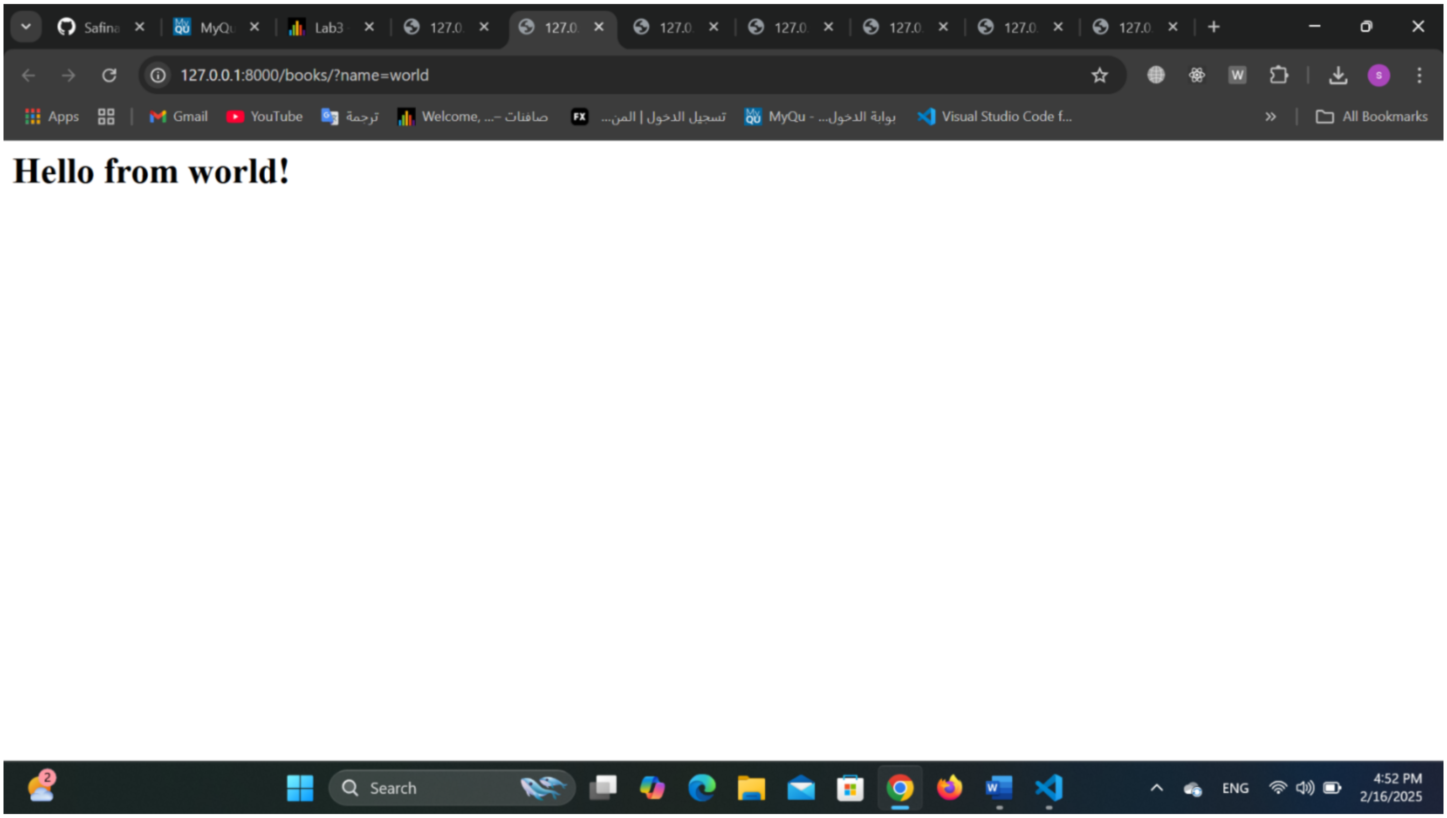


Hello from world!!

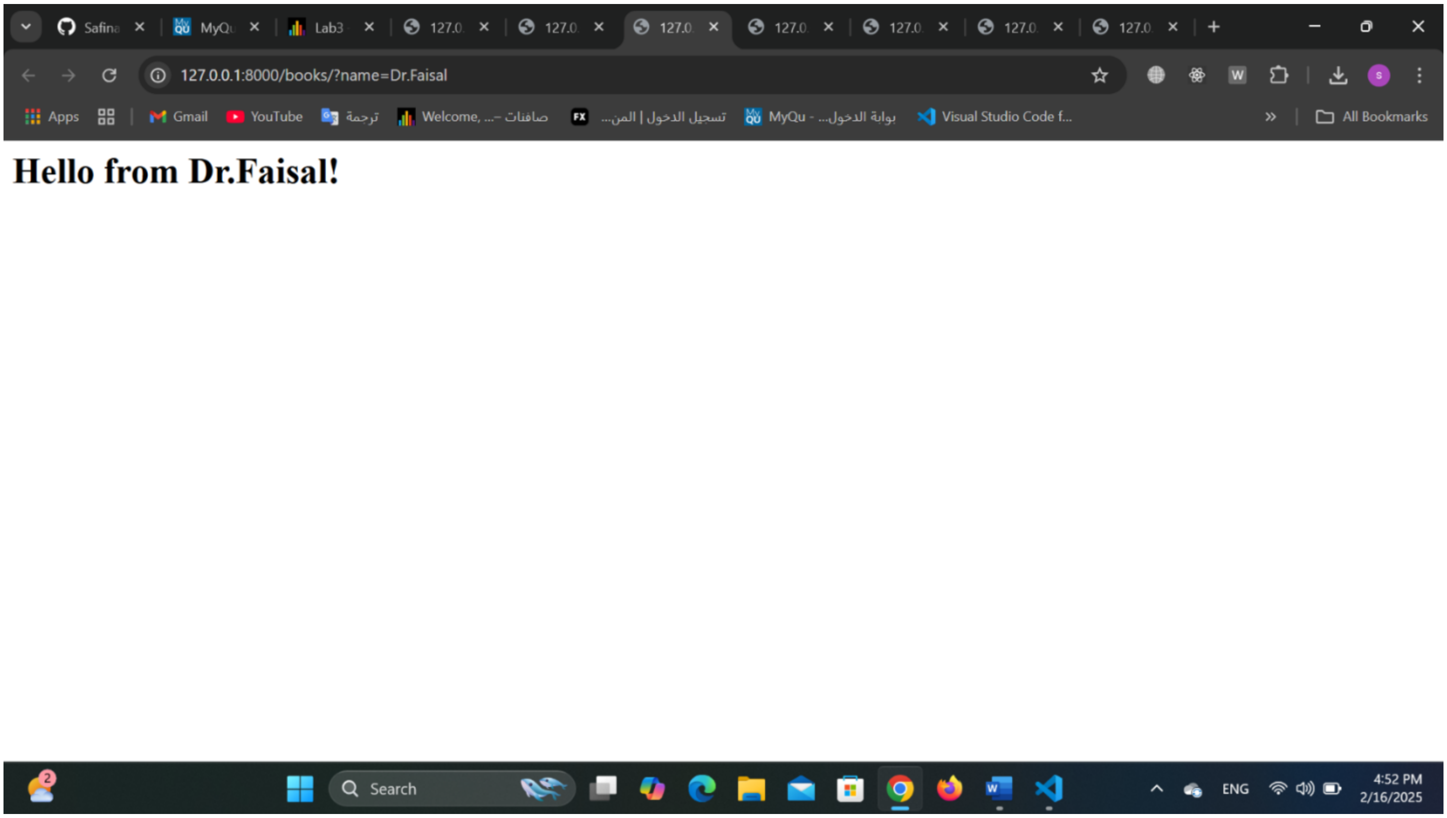
Task 6: Define patterns globally (DjangoProjects/urls.py) with specific urls.py file for each app/module



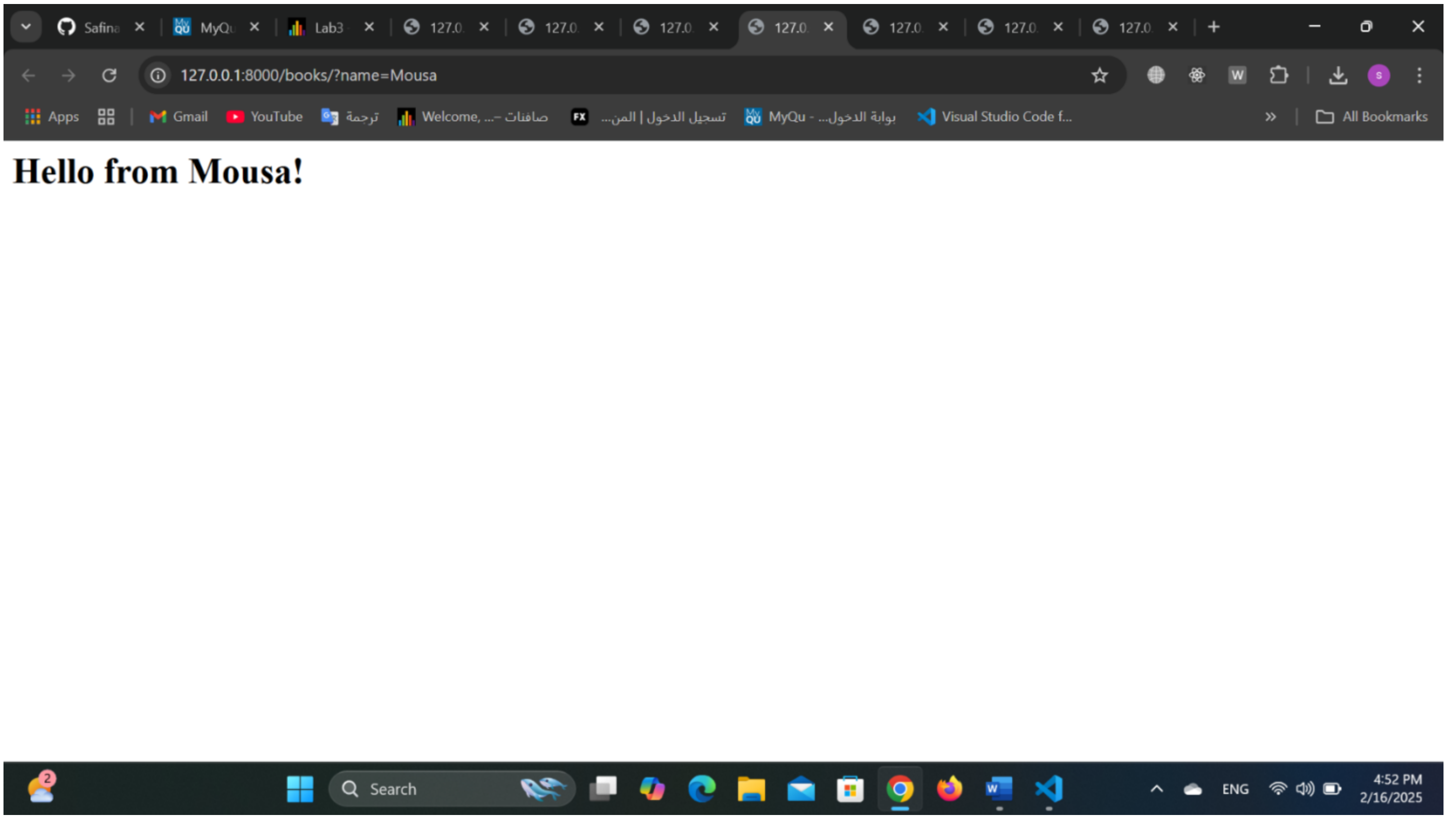
Hello from world!



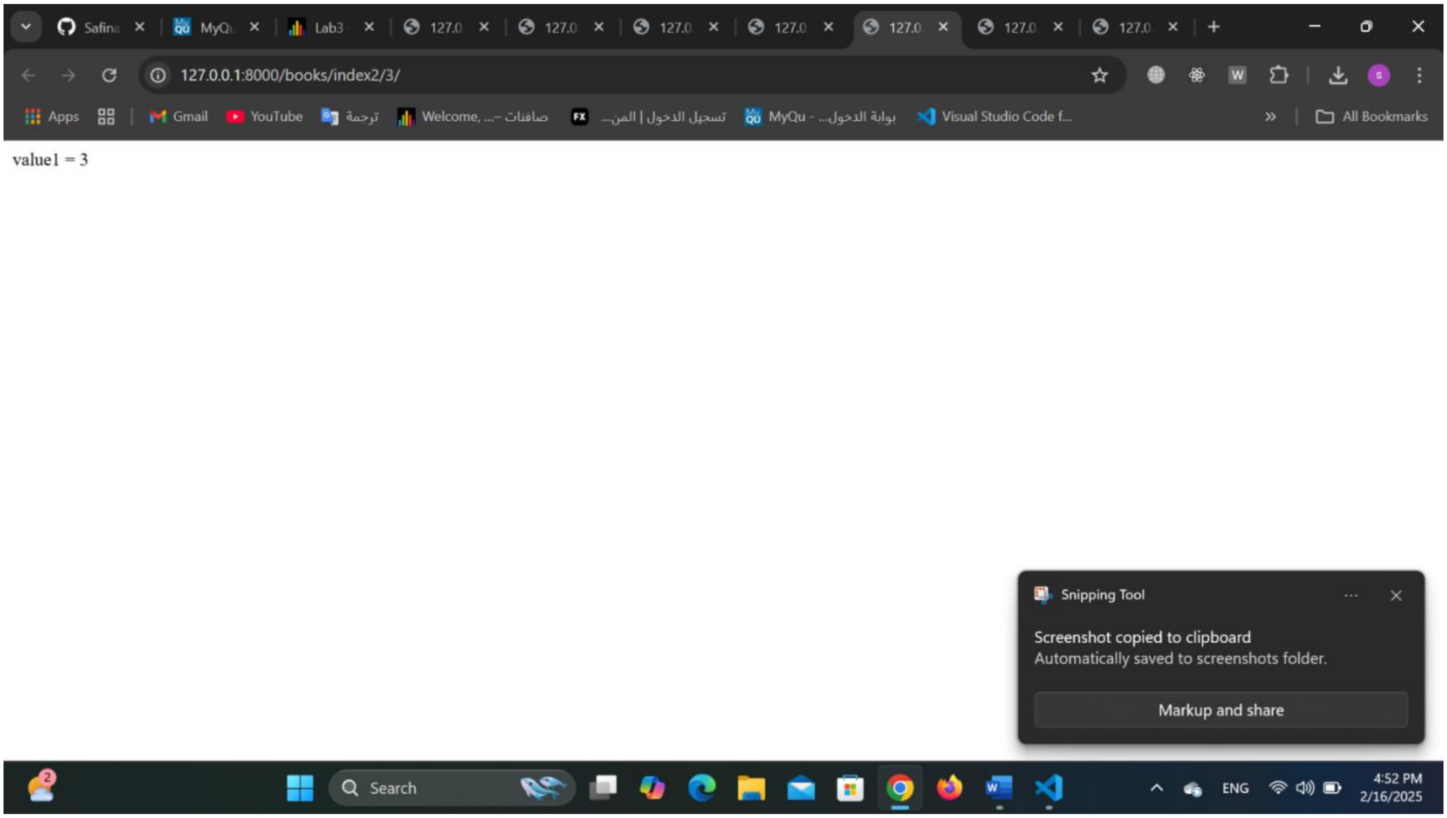
Hello from world!



Hello from Dr.Faisal!

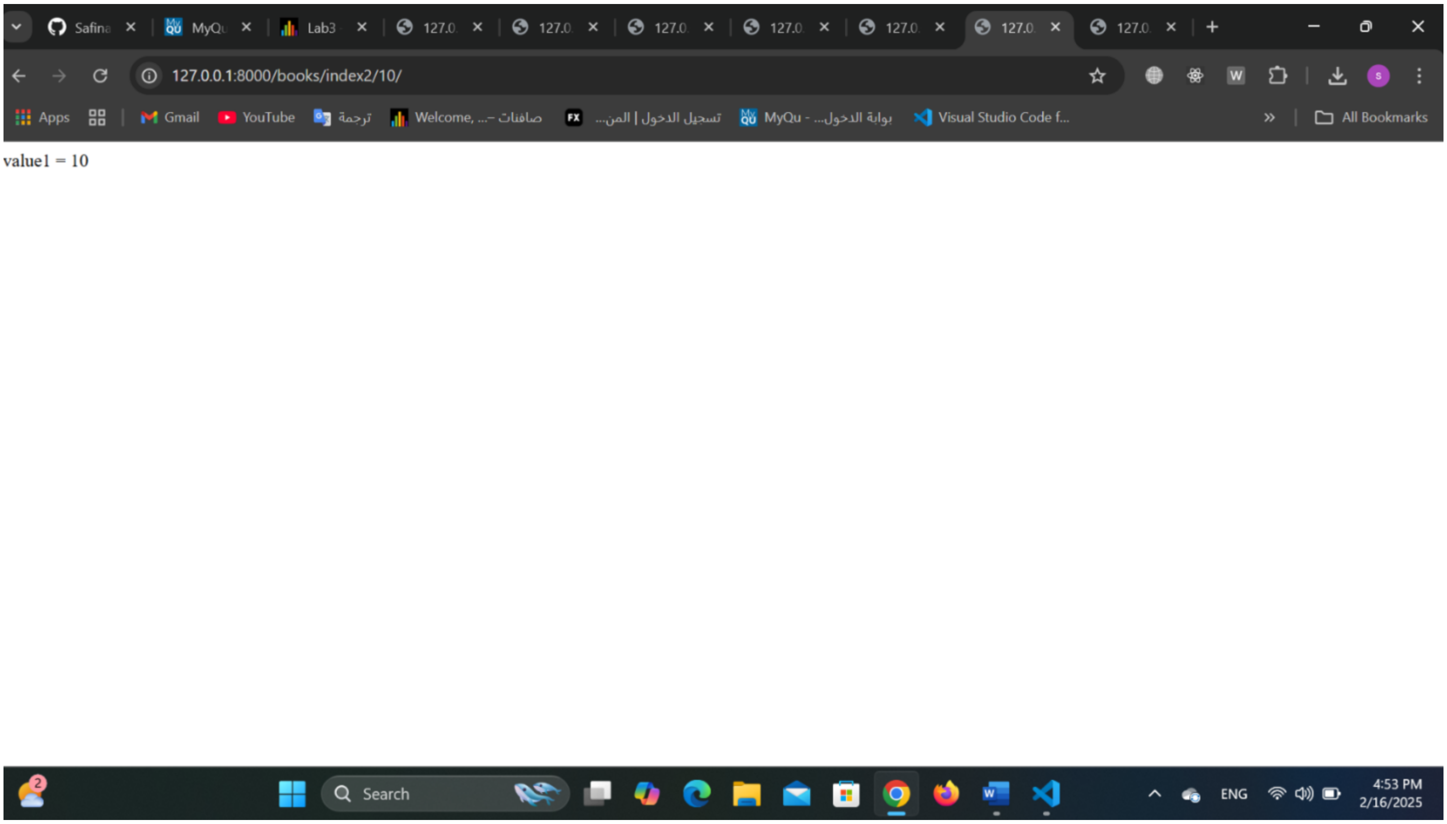


Hello from Mousa!

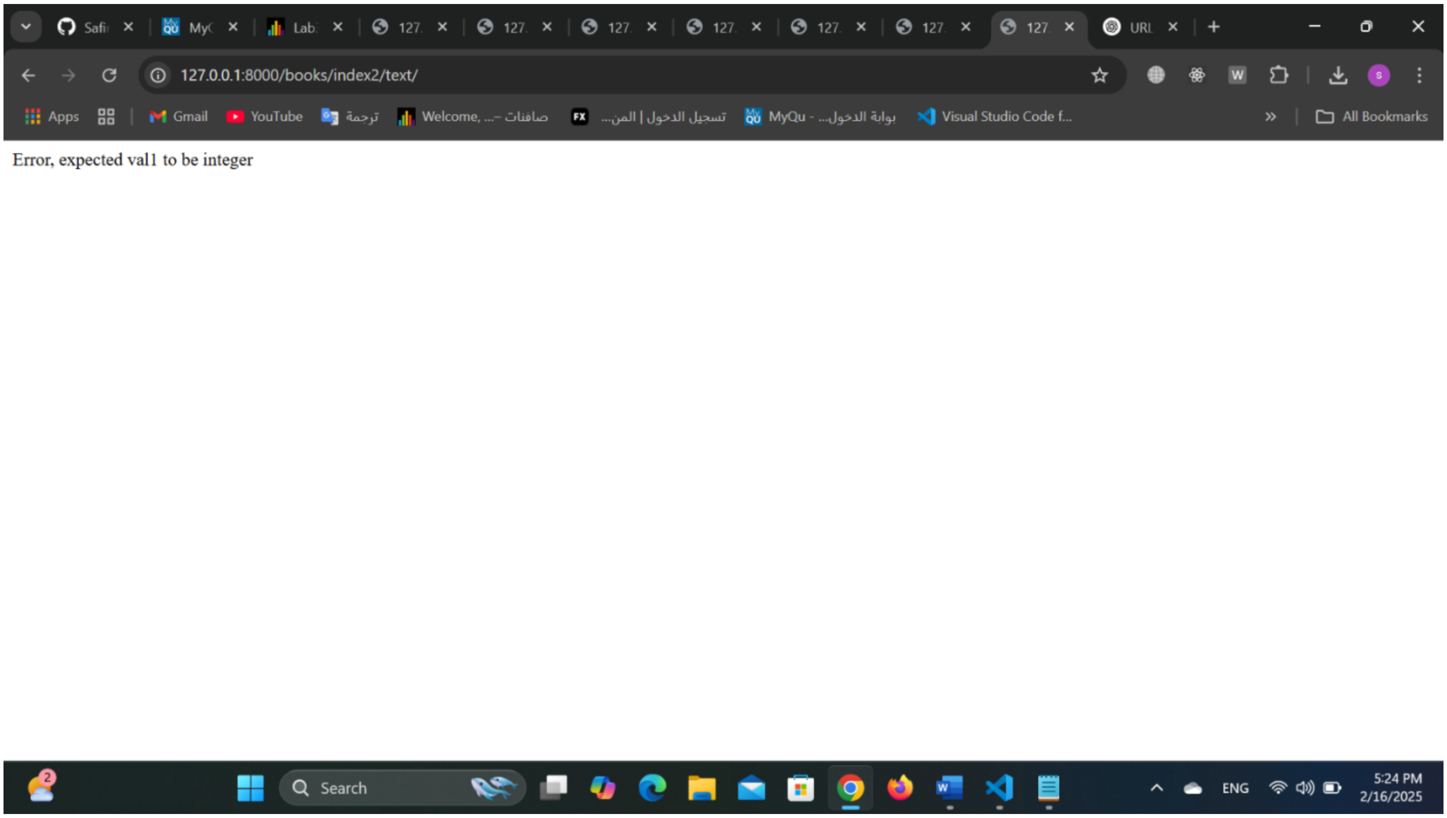


value1 = 3

Snipping Tool
Screenshot copied to clipboard
Automatically saved to screenshots folder.
Markup and share

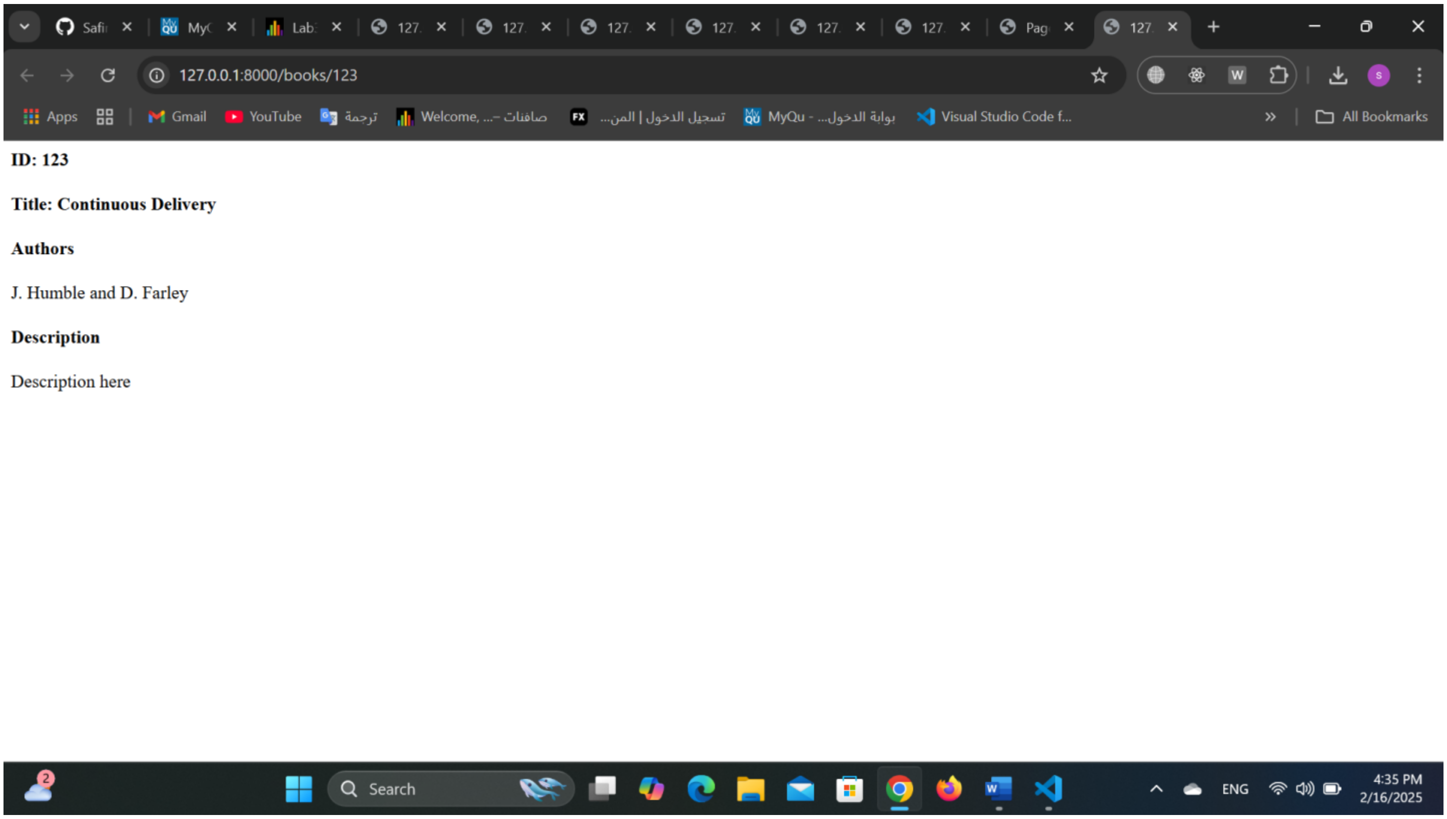


value1 = 10



Error, expected val1 to be integer

Task7: Create a URL, view, and HTML to display one book details



ID: 123

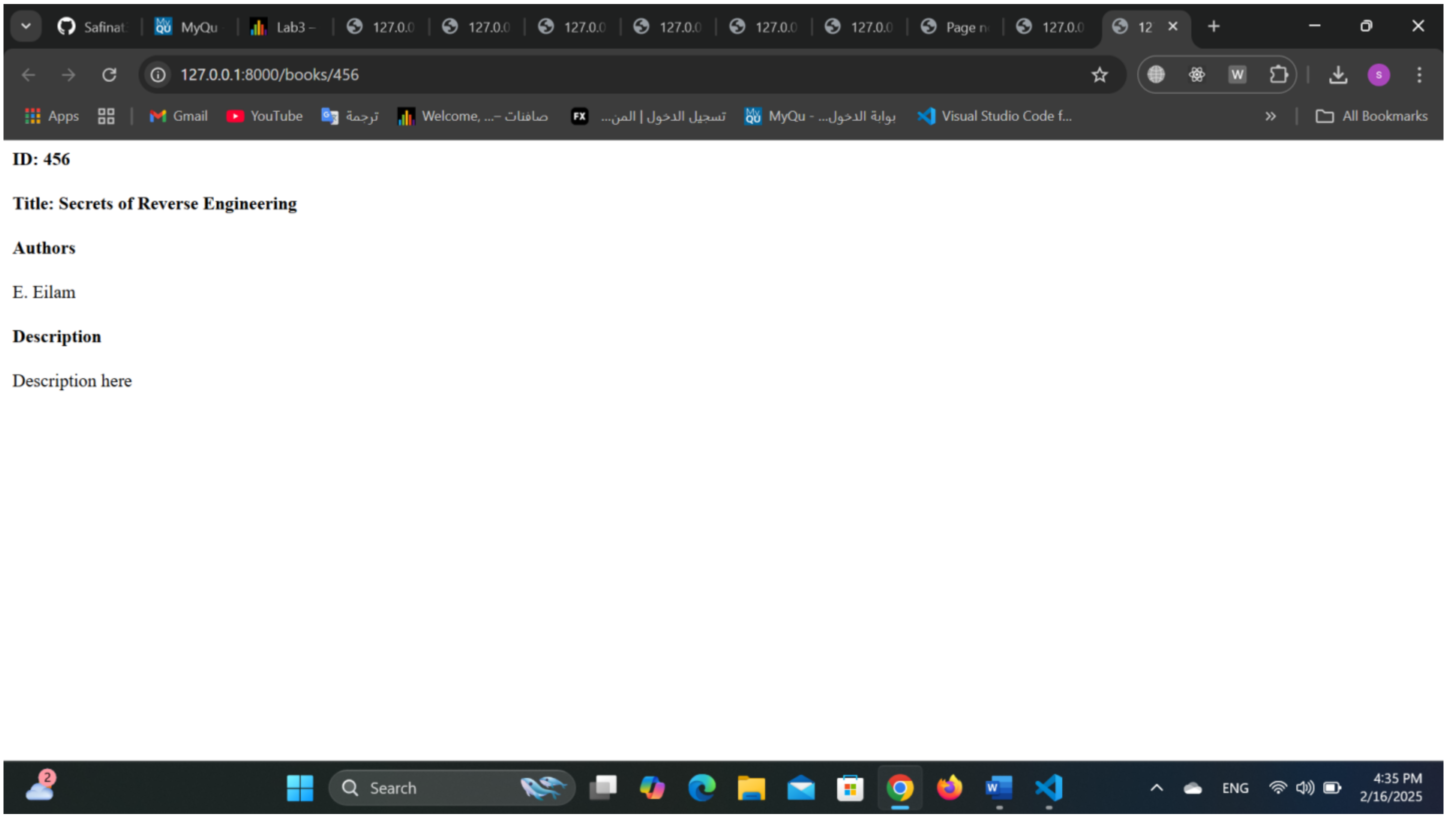
Title: Continuous Delivery

Authors

J. Humble and D. Farley

Description

Description here



ID: 456

Title: Secrets of Reverse Engineering

Authors

E. Eilam

Description

Description here

Additional content: To redirect to another view or URL, use the function redirect, and for details and examples visit the following link: https://docs.djangoproject.com/en/5.1/topics/http/shortcuts/#redirect