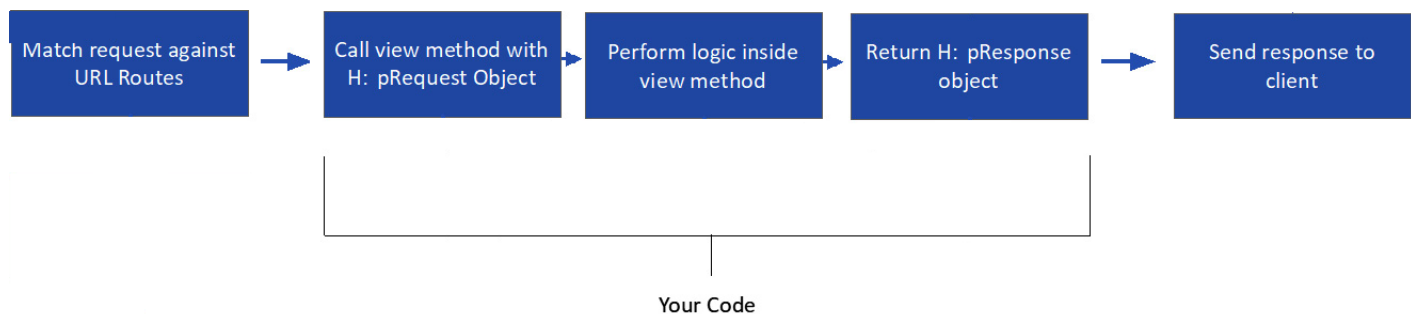


This lab session covers the basics of the Django URL and HTML template. By the end of this session, students should have a very good understanding of how to define URL patterns and create an HTML template and how both work together to Process an HTTP request. They should also have already practiced with 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.

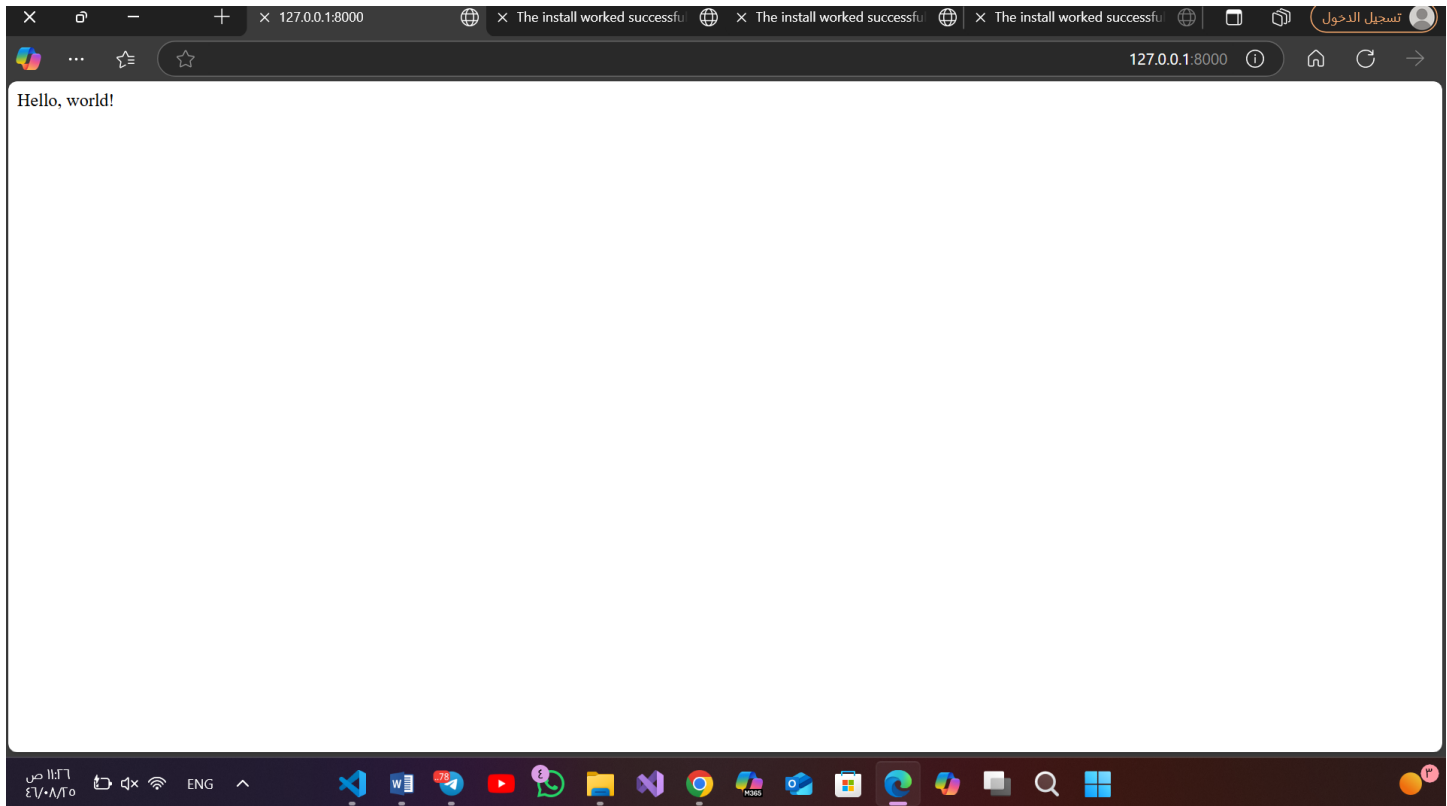


Pre-lab Preparation:

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

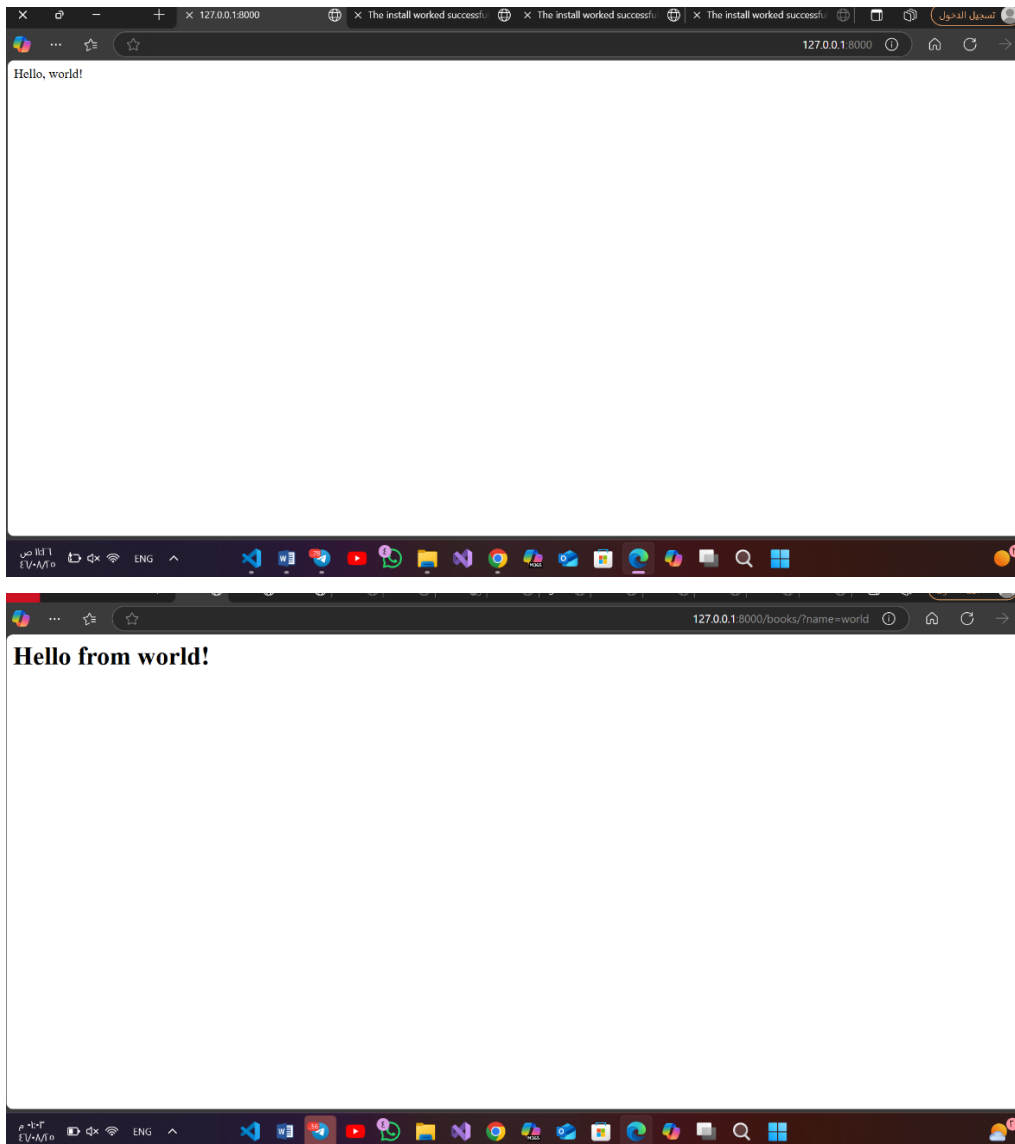
Lab Activities: Build and configure a simple link between a URL and a view (simple pattern), along with a simple HTML template.

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



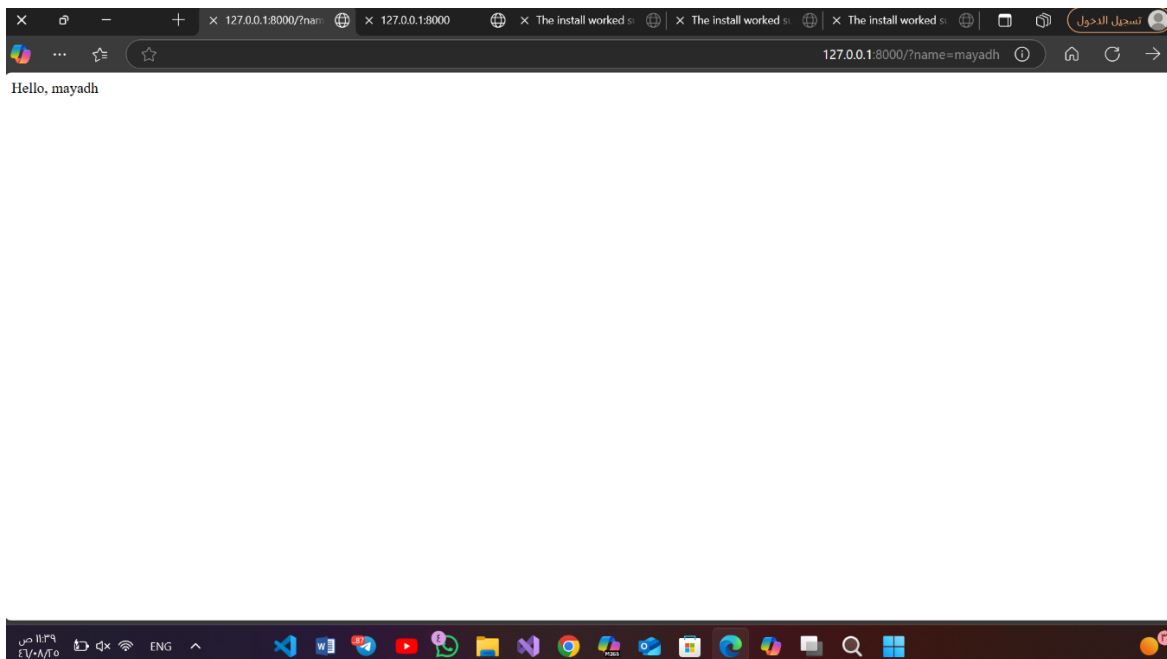
Task 2: Add parameters with HTTP requests¹

Step 2: Now from the browser navigate to the following urls:

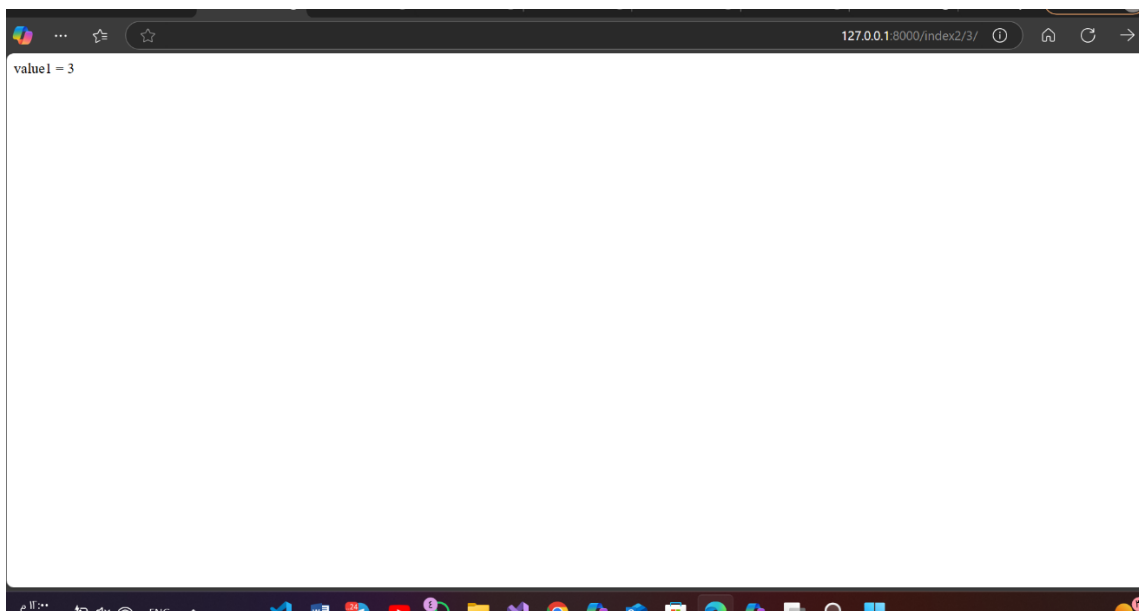


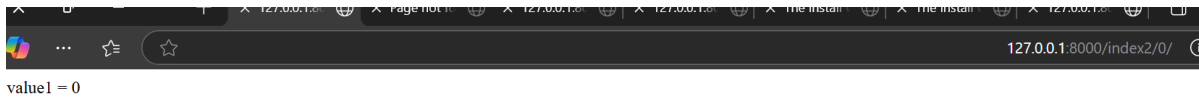
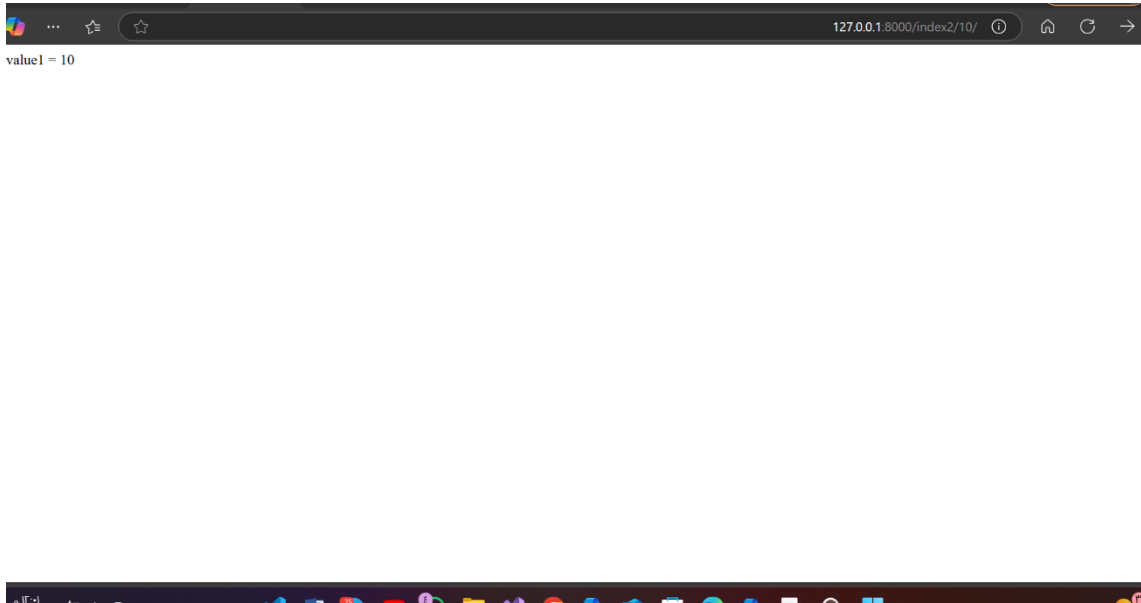
¹ It is possible to have more than one parameter, like <http://127.0.0.1:8000?id=10&name=world!&age=18>

Note: this method is GET HTTP request, which is not recommended for security reasons, and use POST HTTP request instead.

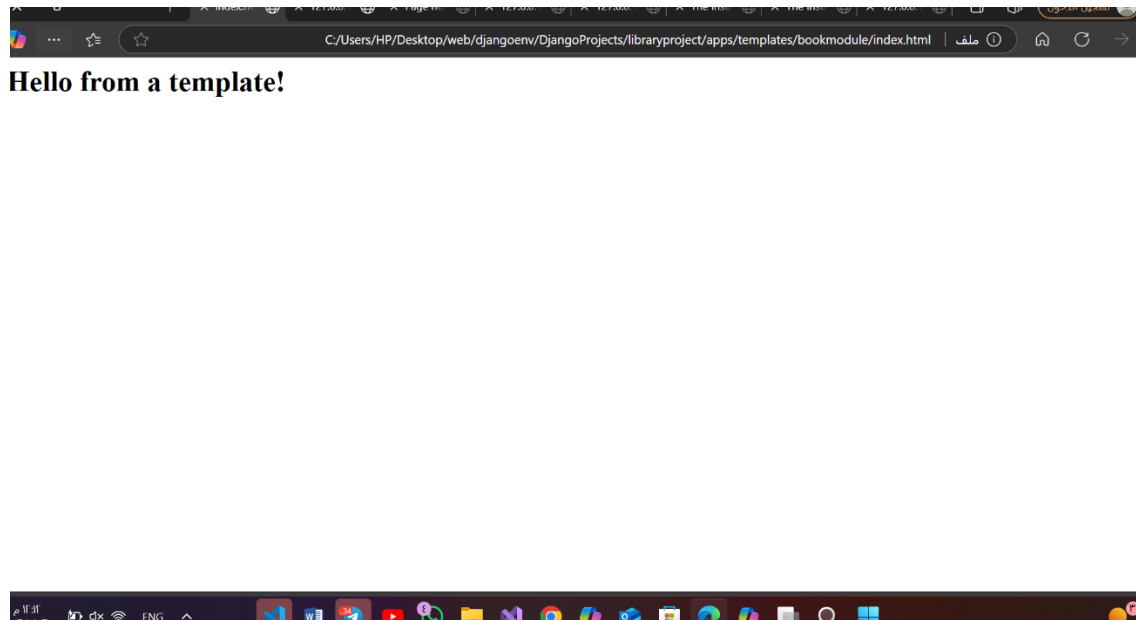


Task 3: Build your second view function and corresponding URL mapping with parameters within URL path

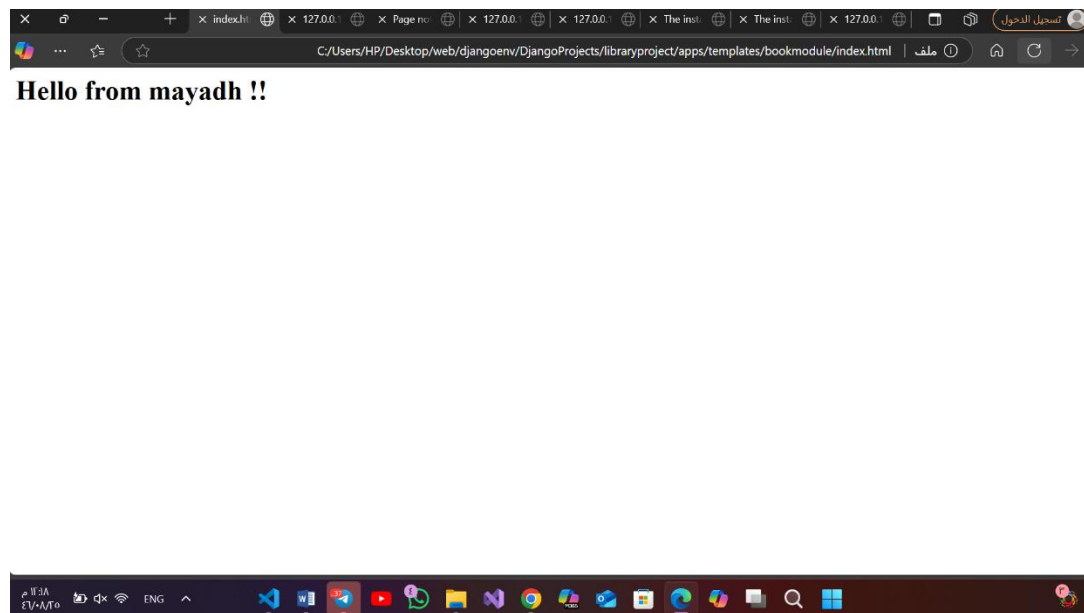




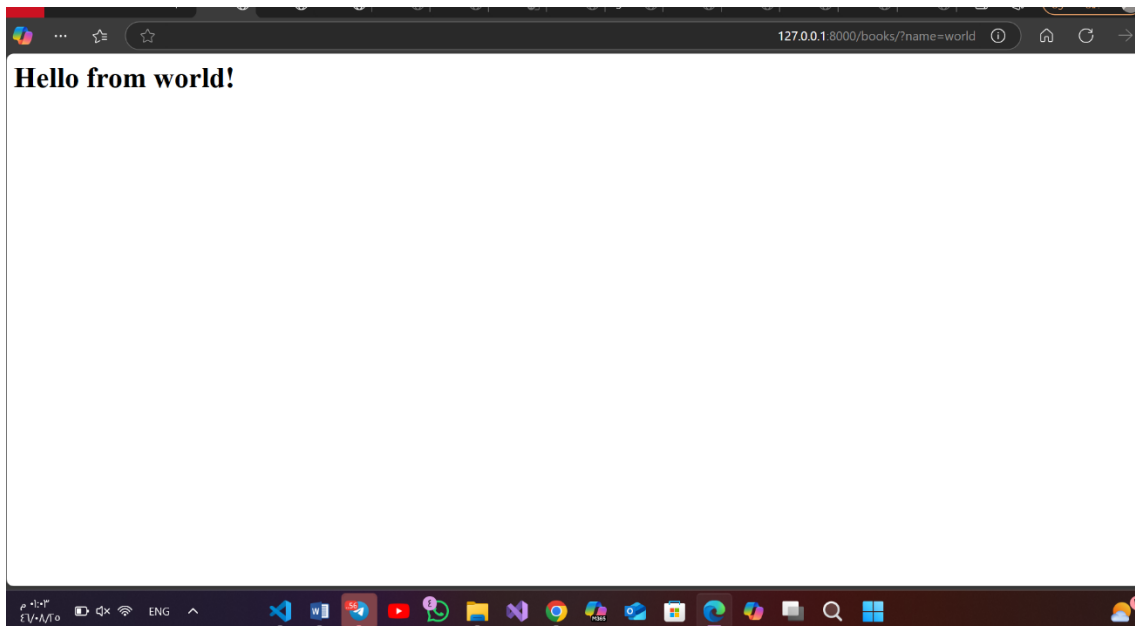
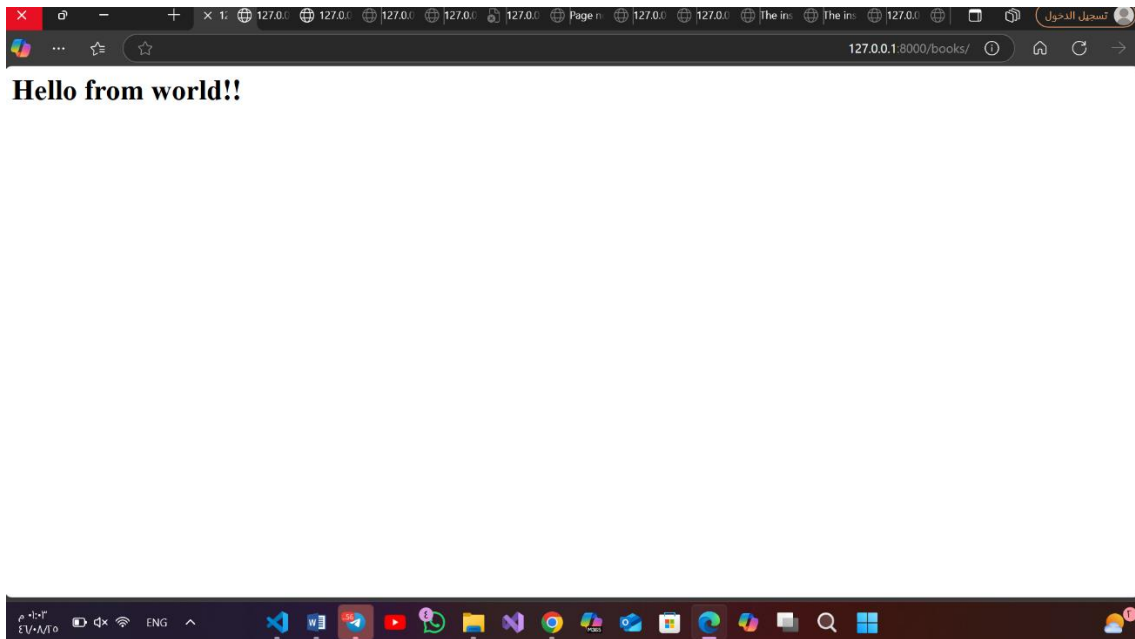
Task 4: Create a simple HTML template

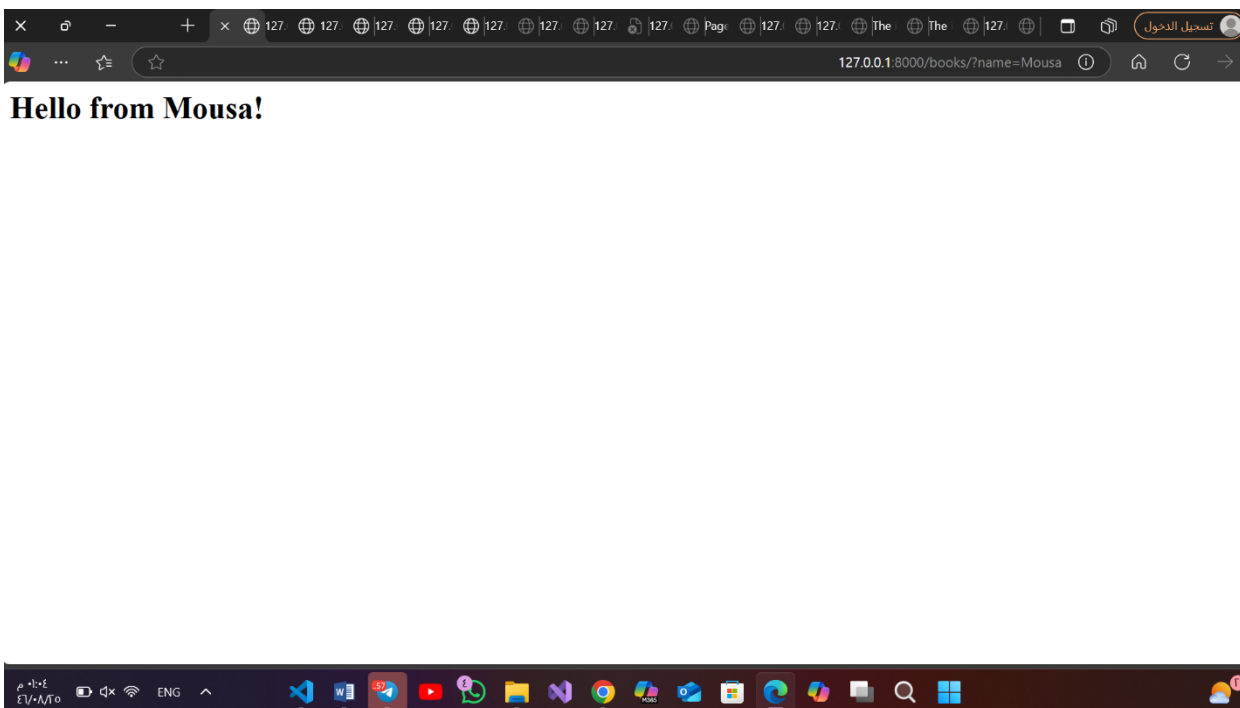
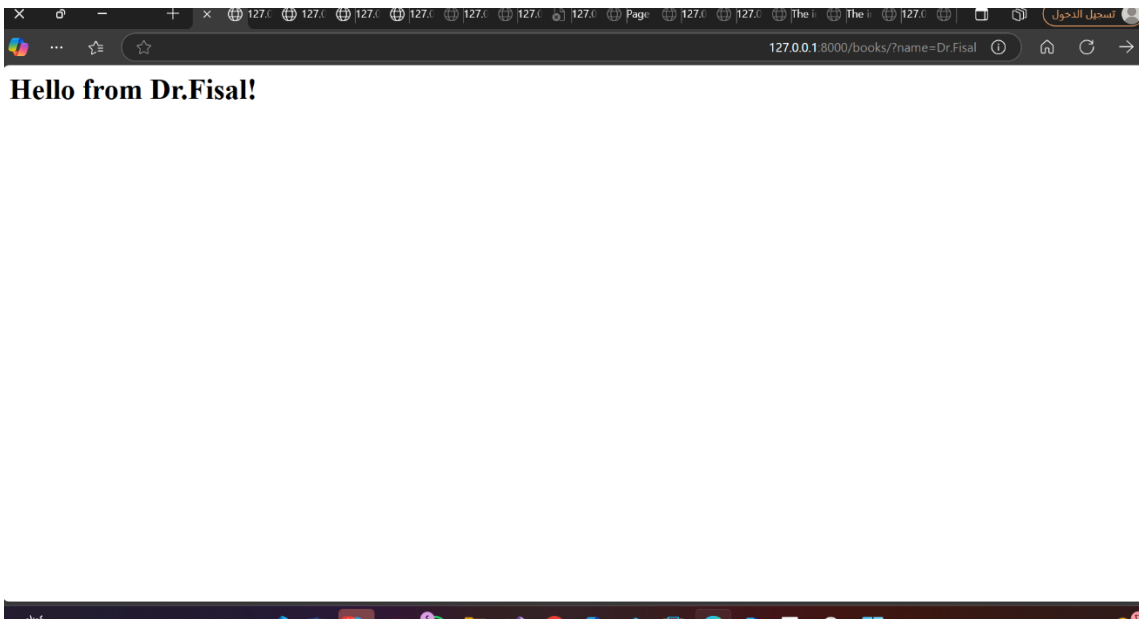


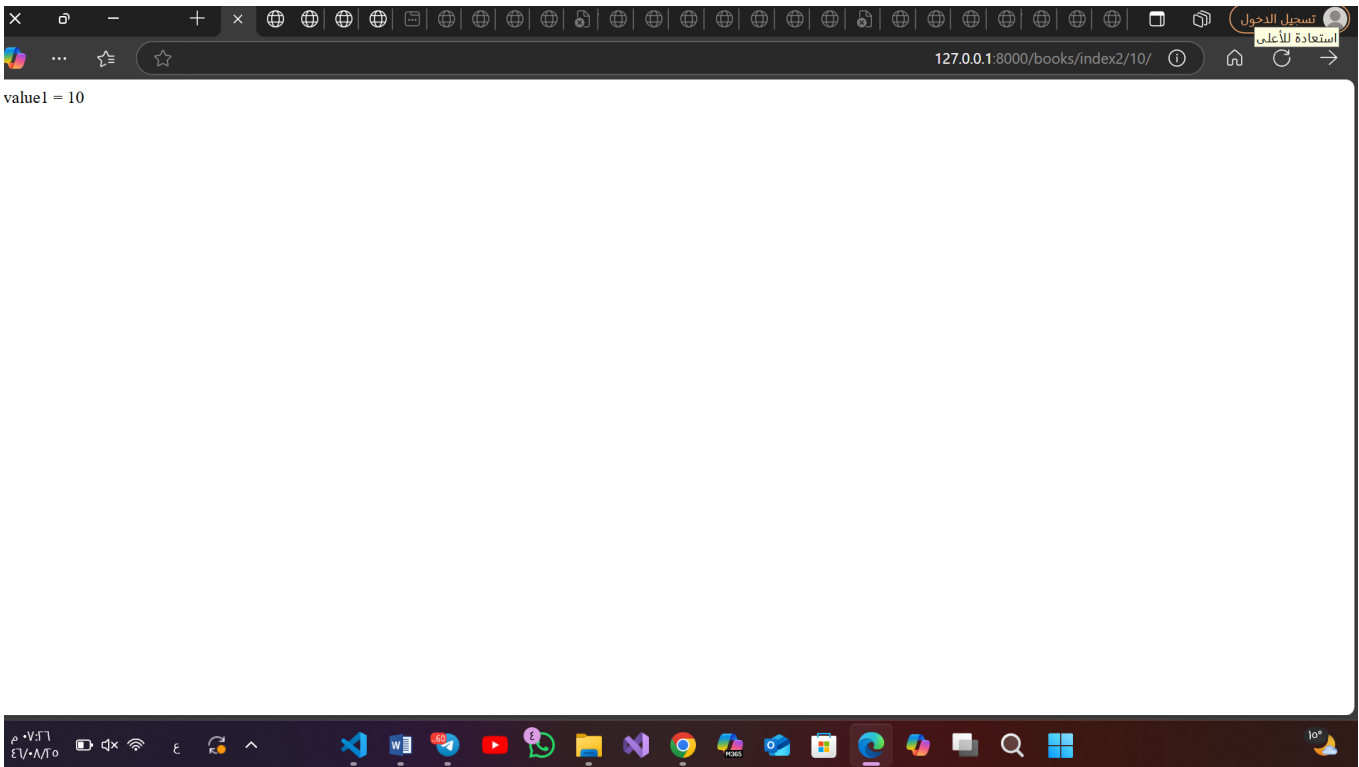
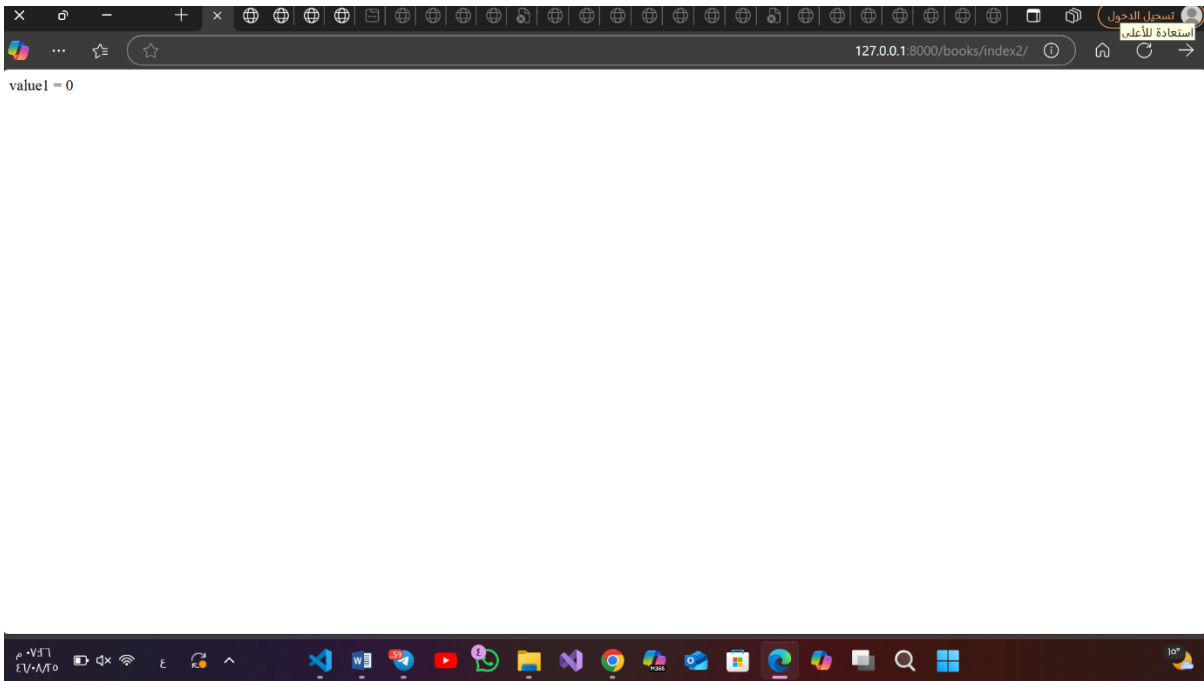
Task 5: Rendering variables in the HTML template that processes a context

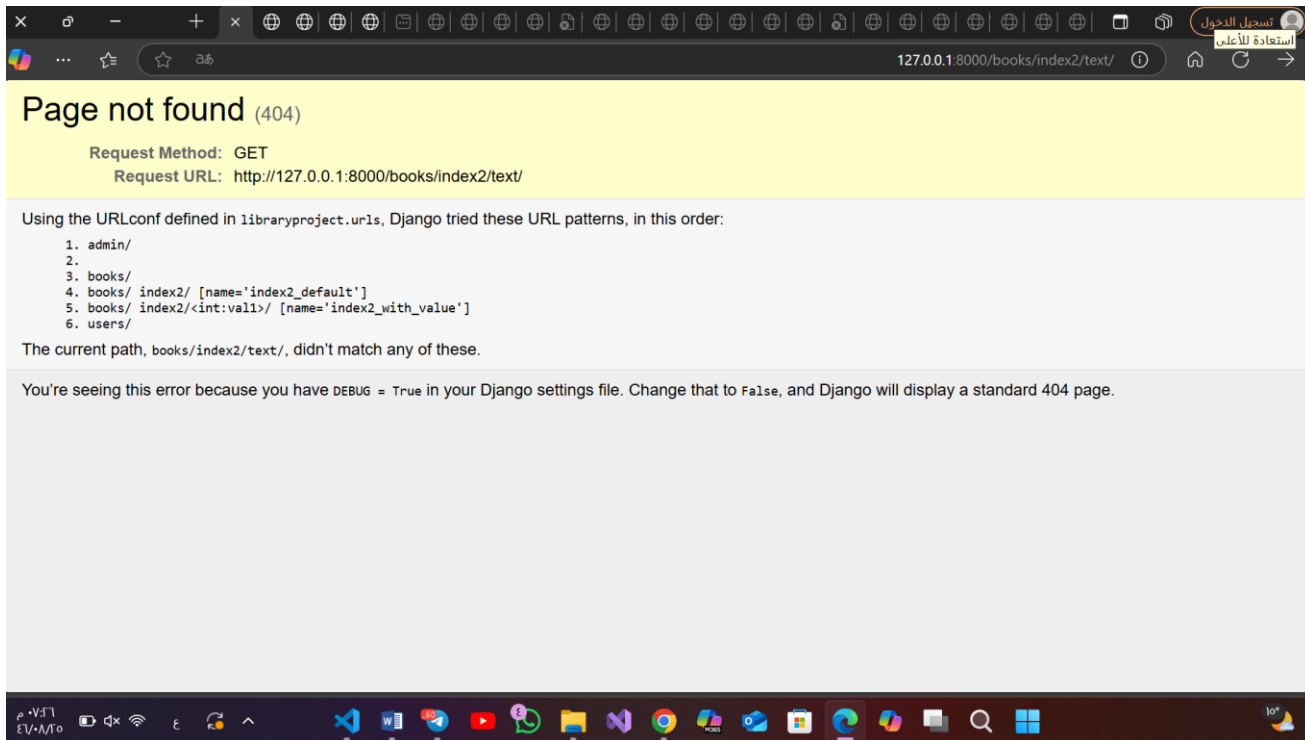


Task 6: Define patterns globally (DjangoProjects/urls.py) with specific urls.py file for each app/module









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

Step 4: Now from the browser navigate to the following urls:

<http://127.0.0.1:8000/books/123>

<http://127.0.0.1:8000/books/456>

