ADAJAR, MILVI M.
BSCpE 2A2
Software Design

Laboratory Activity No. 1:

Topic: **Introduction to Software Design, History, and Overview**
Title: *Setting Up the Development Environment for Django Project*

Introduction: This activity will guide you through the process of setting up your development environment to start building the Library Management System (LMS) in Django. The process involves installing necessary software, setting up Python and Django, and verifying the installation.

Objectives:

- Install Python and Django on your system.
- Create a virtual environment to manage dependencies.
- Verify the installation by running a simple Django project.

Theory and Detailed Discussion: To develop the Library Management System, we will use the Django framework. Django is a high-level Python web framework that allows developers to create robust web applications quickly and efficiently. Before we can start developing, we need to set up the development environment.

Materials, Software, and Libraries:

- **Python** (version 3.8 or above)
- **Django** (version 4.0 or above)
- **pip** (Python package manager)
- **Text Editor** (Visual Studio Code or PyCharm)
- **Database** (SQLite – comes with Django by default)

Time Frame: 1 Hour

Procedure:

1. **Install Python**:
    - Go to python.org and download the latest version of Python.
    - Install Python by following the installation instructions for your operating system.

2. **Install pip** (Python package installer):
    - Open a terminal and type the following command:

```
python -m ensurepip –upgrade
```

```
C:\Users\USER>python -m ensurepip --upgrade
Looking in links: c:\Users\USER\AppData\Local\Temp\tmpfi2gntfu
Requirement already satisfied: pip in c:\users\user\appdata\local\programs\python\python313\lib\site-packages (25.0)
```

3. **Install Virtual Environment**:
    - Create a virtual environment for our project to avoid conflicts with global packages.

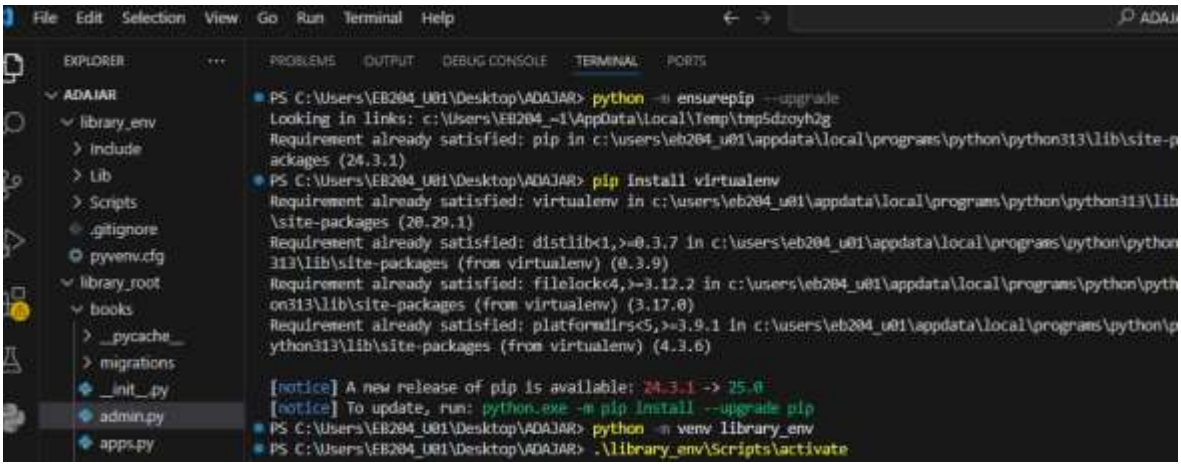
```
pip install virtualenv
```

    - Create a new virtual environment:

```
python -m venv library_env
```

    - Activate the virtual environment:
    - On Windows:

```
.\library_env\Scripts\activate
```

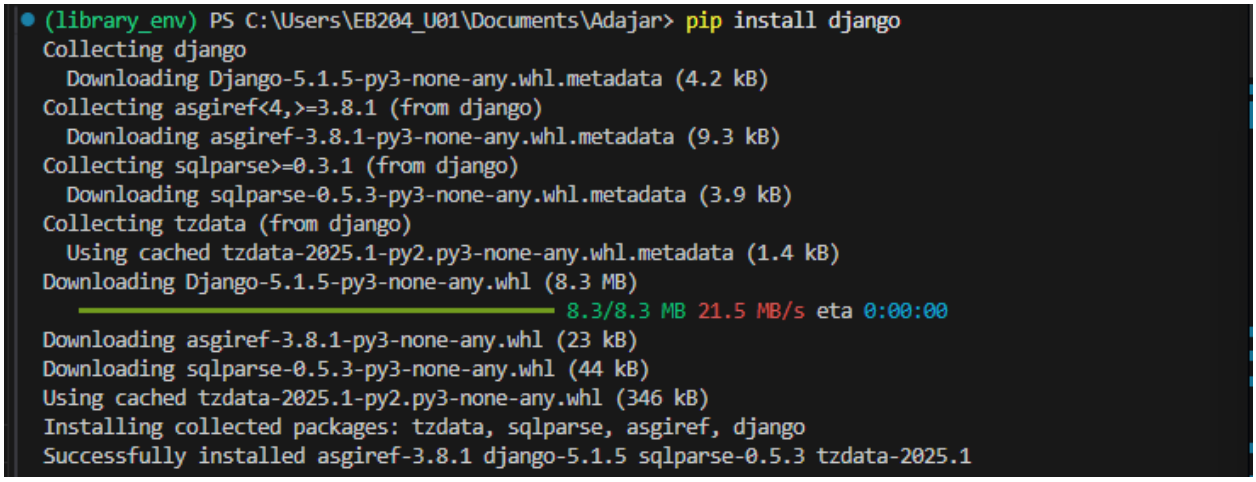


- On Mac/Linux:

```
source library_env/bin/activate
```

1. **Install Django**:
    - After activating the virtual environment, install Django by running:
2.

```
pip install django
```



3. **Verify the Django Installation**:
    - Run the following command to verify if Django is installed:

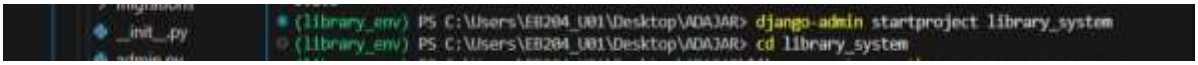
```
django-admin --version
```

4. **Create a New Django Project**:
    - Create a new Django project called "library_system":

```
django-admin startproject library_system
```

    - Navigate into the project directory:

```
cd library_system
```

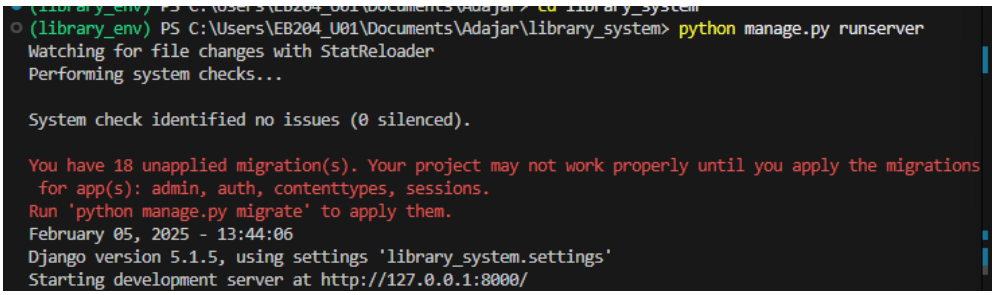


5. **Run the Django Development Server**:

- Start the development server to verify everything is working:

```
python manage.py runserver
```

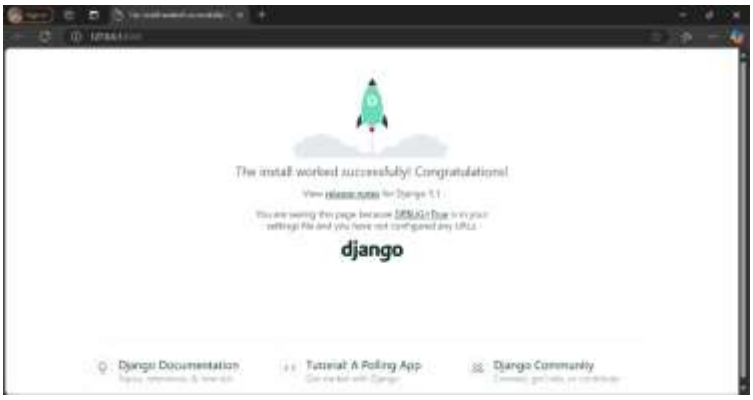
- Open a browser and go to `http://127.0.0.1:8000/`. You should see the Django welcome page.



Program/Code: The code here is focused on setting up the environment. The following commands should be run in the terminal:

```
python -m venv library_env
source library_env/bin/activate  # or .\library_env\Scripts\activate on
Windows
pip install django
django-admin startproject library_system
cd library_system
python manage.py runserver
```

Results: (print screen the result and provide the github link of your work)



Github Link:

Follow-Up Questions:

1. What is the role of a virtual environment in Django development?

- A virtual environment isolates the dependencies of different projects, ensuring that each project has its own version of libraries without conflicting with global installations. This makes development more manageable and prevents version-related issues.

2. What are the advantages of using Django for web development over other frameworks?

- Django enables rapid development with built-in features like authentication and ORM, ensures security against common threats, supports scalability for large applications, and benefits from strong community support and documentation.

Findings:

During the setup process, it was observed that installing and configuring Django was straightforward, provided that dependencies were correctly installed. Some common issues included virtual environment activation errors in Windows, which were resolved by adjusting the execution policy.

Summary:

The key steps in this activity included installing Python, setting up a virtual environment, installing Django, creating a project, and running the development server. The activity provided an understanding of the importance of an isolated environment and the basic setup required for Django projects.

Conclusion:

Setting up the Django development environment is a crucial first step in web application development. This activity demonstrated the importance of using a virtual environment to manage dependencies, ensuring a smooth and organized development workflow. By following the outlined steps, developers can create and manage Django projects efficiently.