

Flask Application Installation Guide

This guide will walk you through setting up and running the Flask application project we completed in class. Follow these steps carefully to ensure your development environment is properly configured.

Prerequisites

Before starting, make sure you have:

- **Python 3:** Verify installation by running `python3 --version` in your terminal
- **Terminal/Command Prompt:** Basic familiarity with navigating directories
- **VS Code:** We'll be using VS Code as our code editor
- **Project Files:** The Flask project folder from class

Step-by-Step Installation

Step 1: Navigate to Your Project Directory

Open your terminal and navigate to the root directory of your Flask project:

```
bash  
cd /path/to/your/Flask_db/Flask_db
```

Replace `/path/to/your/` with the actual path where you saved the project files.

Step 2: Create a Virtual Environment

Create an isolated Python environment for your project dependencies:

```
bash  
python3 -m venv venv
```

This creates a new folder named `venv` in your project directory containing an isolated Python environment.

Step 3: Activate the Virtual Environment

Activate the virtual environment using the appropriate command for your operating system:

macOS/Linux:

```
bash
```

```
source venv/bin/activate
```

Windows (Command Prompt):

```
cmd
```

```
venv\Scripts\activate
```

Windows (PowerShell):

```
powershell
```

```
.\venv\Scripts\Activate.ps1
```

Confirmation: Your terminal prompt should now show `(venv)` at the beginning, indicating the virtual environment is active.

Step 4: Install Project Dependencies

With the virtual environment activated, install all required packages:

```
bash
```

```
pip install -r flask_requirements.txt
```

This installs Flask, Flask-SQLAlchemy, Werkzeug, and all necessary dependencies.

Step 5: Verify Installation (Optional)

Check that packages are correctly installed:

```
bash
```

```
pip freeze
```

You should see Flask, Flask-SQLAlchemy, Werkzeug, and their dependencies listed.

Step 6: Configure VS Code

1. Open your Flask project folder in VS Code
2. Press `(Ctrl+Shift+P)` (Windows/Linux) or `(Cmd+Shift+P)` (macOS)
3. Type "Python: Select Interpreter" and select it

4. Choose the interpreter pointing to your virtual environment:

```
/path/to/your/project/venv/bin/python
```

5. Restart VS Code completely to ensure all changes take effect

Step 7: Run Your Flask Application

1. **Ensure virtual environment is active** (you should see `(venv)` in your terminal)

2. **Set the Flask app variable:**

```
bash
```

```
export FLASK_APP=flask_app_with_db.py
```

3. **Start the development server:**

```
bash
```

```
flask run
```

4. **Access your application:**

- Open your web browser
- Navigate to `http://127.0.0.1:5000`
- You should see your Flask application running!

Development Tips

Enable Debug Mode

For development, enable debug mode for better error messages and auto-reload:

```
bash
```

```
export FLASK_DEBUG=1
```

```
flask run
```

Common Commands

Stop the server: Press `Ctrl + C` in the terminal

Deactivate virtual environment:

```
bash
```

```
deactivate
```

Reactivate virtual environment:

bash

`source venv/bin/activate` # macOS/Linux

or

`venv\Scripts\activate` # Windows

Troubleshooting

Common Issues

1. "Command not found" errors:

- Ensure Python 3 is installed and in your PATH
- Try using `python` instead of `python3` on some systems

2. Import errors in VS Code:

- Verify VS Code is using the correct Python interpreter
- Restart VS Code after changing the interpreter

3. Permission errors on Windows:

- Run terminal as administrator
- Or use `py -m venv venv` instead of `python3 -m venv venv`

4. Flask app not found:

- Ensure `FLASK_APP` is set correctly
- Verify the Python file name matches exactly

Getting Help

If you encounter issues:

1. Check that your virtual environment is activated
2. Verify all dependencies are installed with `pip freeze`
3. Ensure VS Code is configured with the correct Python interpreter
4. Check file paths and names for typos

Project Structure

Your project should look like this:

```
Flask_db/
└── Flask_db/
    ├── venv/          # Virtual environment (created)
    ├── flask_app_with_db.py  # Main Flask application
    ├── flask_requirements.txt # Dependencies list
    └── [other project files]
```

Next Steps

Once your Flask application is running successfully:

- Explore the application features
- Review the code structure
- Make modifications to understand the workflow
- Test different endpoints and functionality

Happy coding!