

## Report on FastAPI Deployment and Selenium Testing Sessions

### Session 1: FastAPI Deployment on Render

In the first session, we focused on deploying a FastAPI-based portfolio website on Render. The following steps were covered:

Creating a Virtual Environment:

```
python -m venv ~/profile
```

```
source ~/profile/scripts/activate
```

```
python -m pip install "fastapi[standard]"
```

Setting Up the FastAPI Application:

Navigated to the project directory and created app.py

Used touch app.py to create the file

Opened the file using code app.py

Ran the FastAPI application with `uvicorn app:app --reload`

Created a templates directory and added index.html

Deploying on Render:

Build Command: `pip install fastapi "uvicorn[standard]"`

Start Command: `uvicorn app:app --host 0.0.0.0 --port $PORT`

Git Commands for Version Control:

`git diff` to check changes

Introduction to Selenium:

Installed Selenium: `python -m pip install selenium`

Launched a Chrome browser using Selenium

Interacted with a webpage by locating elements using `find_element` and sending keys

Virtual Environment Management:

Created and deleted 10 virtual environments

Used `deactivate` to exit the virtual environment

Session 2: Testing and Continuous Integration with GitHub Actions

Installing FinTest Pro Extension:

Added `test_main.py` in the portfolio website folder

Used TestClient from fastapi.testclient to test the homepage

Setting Up GitHub Actions for CI/CD:

Created main.yml to automate testing

Configured GitHub Actions to trigger on push to main or any branch

Installed dependencies in the workflow

Ran tests using pytest test\_main.py

Adding Dependencies to requirements.txt:

Included FastAPI and pytest for testing

This session provided hands-on experience with deploying a web application on Render and implementing CI/CD using GitHub Actions. It also introduced web automation testing using Selenium.

