Setup Instructions

Python Setup

Download the latest Python version from Python.org.

Download a Python editor/IDE of your choice. Good choices include PyCharm and Visual Studio Code.

Git Setup

Create an account on Github.com
Download GitHub for desktop

New Project Setup

Create a new repository in GitHub :
- Click New
- Complete Repository name
- Add a description (Optional)
- Check Add a README file (complete this with details about project setup and how to run the tests)
- Add python .gitignore file
- Choose MIT Licence

Open in Github for desktop and clone on your machine:
- Click Code -> Open with Github for desktop
- Click clone (The new repository appear on your machine)
- Keep you project up to date on github : commit and push

Open the new repository(project) with the python editor (Pycharm)
Install dependencies:
- Run `pip install pipenv` to install the dependencies.
- Run `pipenv install`
- Run `pipenv install pytest` to install pytest (testing framework)
- Run `pipenv install selenium`

- Run `pipenv install webdriver_manager`
- Run `pipenv install pytest-html` - for the report
- Run `pipenv install pytest-xdist` - for run in parallel

How to run tests:

- Run `pipenv run python -m pytest` to run all tests in the project
- Run `pipenv run python -m pytest tests/test_login_page.py` to run all tests from a file
- Run `pipenv run python -m pytest tests/test_login_page.py::test_login_with_success` to run one test from a specific file
- Run `pipenv run python -m pytest --html=report.html` to create a report (name of the report can be changed)
- Run `pipenv run python -m pytest -n 3` to run tests in parallel (3 can be changed)

Details about project:

- Pytest :
    - Package pages - create a class page for each page under test
    - Package tests - each file and test should start with test_
        - Create a new file for each functionality
        - Conftest.py - setup for tests (what is doing before and after each test)
- Pytest-bdd:
    - Package tests
        - Conftest.py - setup for tests (what is doing before and after each test)
        - Package features
            - Create a new file for each functionality loginPage.feature
        - Package step_defs
            - create a page for each page under test
            - each file should start with test_