**WriteUP**

Step 1 : Create a simple Maven project and add below POM dependencies:

Step 2 : Create <dependencies>

Step3: Create all the packages in the Project.

Create a TestNG test class to set up your testing environment.

Step 4: Implement Page Object Model (POM) Design Pattern

Create classes representing the web pages you'll interact with Testpage

Define methods within these classes to interact with the elements on the respective pages.

Step 5: Write Test Cases Using TDD

Write a test case for a feature you want to implement (e.g., displaying the course list).

Run the test case (it should fail initially).

Implement the corresponding functionality in your application code to make the test pass.

Refactor your code if needed.

Repeat these steps for practices and training features.

Step 6: Set Up GitHub Repository

Create a new repository on GitHub.

Initialize a Git repository in your Eclipse project.

Add a .gitignore file to specify files/folders to ignore.

Commit your initial code to the repository.

Step 7: Track Changes and Push to GitHub

Commit changes regularly as you implement new features.

Push the commits to your GitHub repository.

Document any files or directories that are ignored during the final push..

Step 9: Test on Simplilearn's Website

Use the TestNG test suite to run all your test cases.

Ensure that your application interacts correctly with the Simplilearn website.

Handle any potential issues that may arise during the testing phase.

Remember to adapt these steps based on your specific requirements and the structure of your web application. This approach ensures that your application is thoroughly tested, and the codebase is well-organized and maintainable.