Page Object Model

- Creating Selenium test cases can result in an unmaintainable project. One of the reasons is that too many duplicated code is used. Duplicated code could be caused by duplicated functionality and this will result in duplicated usage of locators. The disadvantage of duplicated code is that the project is less maintainable. If some locator will change, you have to walk through the whole test code to adjust locators where necessary. By using the page object model we can make non-brittle test code and reduce or eliminate duplicate test code. Beside of that it improves the readability and allows us to create interactive documentation. Last but not least, we can create tests with less keystroke. An implementation of the page object model can be achieved by separating the abstraction of the test object and the test scripts.

What is POM?

- Page Object Model is a design pattern to create Object Repository for web UI elements. Under this model, for each web page in the application, there should be corresponding page class. This Page class will find the WebElements of that web page and also contains Page methods which perform operations on those WebElements. Name of these methods should be given as per the task they are performing, i.e., if a loader is waiting for the payment gateway to appear, POM method name can be waitForPaymentScreenDisplay().

Advantages of POM

- Page Object Pattern says operations and flows in the UI should be separated from verification. This concept makes our code cleaner and easy to understand. The Second benefit is the object repository is independent of test cases, so we can use the same object repository for a different purpose with different tools. For example, we can integrate POM with TestNG/JUnit for functional Testing and at the same time with JBehave/Cucumber for acceptance testing. Code becomes less and optimized because of the reusable page methods in the POM classes. Methods get more realistic names which can be easily mapped with the operation happening in UI. i.e. if after clicking on the button we land on the home page, the method name will be like 'gotoHomePage()'.
- On executing the program, we will staleElementReferenceException because, when the page gets refreshed, the address of the webElement gets changed.
- To avoid staleElementReferenceException,we will use POM in our framework.
- initElements() static method in pagefactory class performs lazy initialisation.
- In lazy initialization, the reference variables will be initialized at runtime when we try to perform some actions using those reference variables.