

## To Backup Created Packages and Test Them for Stock Management Application.

### Description

Develop a stock management application that helps to search the product in the database and show inventory as well

### Background of the problem statement:

The E-commerce application shows the list of similar products available in stock. By checking the product details, it will search for that product in the database and displays the number of products available in the store.

### You must use the following:

- Eclipse IDE
- Jenkins server
- JFrog Artifactory
- Selenium WebDriver plugin

### Source Code:-

#### Pom.xml:-

```
<project xmlns="https://maven.apache.org/POM/4.0.0"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="https://maven.apache.org/POM/4.0.0
https://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>ATE-Phase2-Selenium-TestNG</groupId>
  <artifactId>ATE-Phase2-Selenium-TestNG</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <dependencies>
    <!--

    https://mvnrepository.com/artifact/org.seleniumhq.selenium/selenium-java -->
    <dependency>
      <groupId>org.seleniumhq.selenium</groupId>
      <artifactId>selenium-java</artifactId>
      <version>4.12.1</version>
    </dependency>
    <!--

    https://mvnrepository.com/artifact/io.github.bonigarcia/webdrivermanager -->
    <dependency>
      <groupId>io.github.bonigarcia</groupId>
      <artifactId>webdrivermanager</artifactId>
      <version>5.4.1</version>
    </dependency>
  </dependencies>
</project>
```

```

        </dependency>
        <!--https://mvnrepository.com/artifact/org.testng/testng
-->
        <dependency>
            <groupId>org.testng</groupId>
            <artifactId>testng</artifactId>
            <version>7.7.1</version>
            <scope>test</scope>
        </dependency>
        <!--
https://mvnrepository.com/artifact/com.aventstack/extentreports-->
        <dependency>
            <groupId>com.aventstack</groupId>
            <artifactId>extentreports</artifactId>
            <version>5.0.9</version>
        </dependency>
        <!--
https://mvnrepository.com/artifact/org.apache.poi/poi-->
        <dependency>
            <groupId>org.apache.poi</groupId>
            <artifactId>poi</artifactId>
            <version>4.1.1</version>
        </dependency>
        <!--
https://mvnrepository.com/artifact/org.apache.poi/poi-ooxml-->
        <dependency>
            <groupId>org.apache.poi</groupId>
            <artifactId>poi-ooxml</artifactId>
            <version>4.1.1</version>
        </dependency>
        <!--
https://mvnrepository.com/artifact/org.apache.poi/poi-ooxml-schemas
-->
        <dependency>
            <groupId>org.apache.poi</groupId>
            <artifactId>poi-ooxml-schemas</artifactId>
            <version>4.1.1</version>
        </dependency>
    </dependencies>
</project>

```

## AmezonTest:-

```

package Amazon;

import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
import org.testng.Assert;
import org.testng.annotations.AfterTest;
import org.testng.annotations.BeforeTest;
import org.testng.annotations.Test;

```

```

public class AmazonTest {
    private WebDriver driver;

    @BeforeTest
    public void setup() {
        driver = new ChromeDriver();

        driver.manage().window().maximize();

        driver.get("https://www.amazon.in/");

        WebElement sign = driver.findElement(By.id("nav-link-accountList"));
        sign.click();

        WebElement email = driver.findElement(By.id("ap_email"));
        email.sendKeys("6306006966");

        WebElement continuee =
        driver.findElement(By.id("continue"));
        continuee.click();

        WebElement password =
        driver.findElement(By.id("ap_password"));
        password.sendKeys("samrat@26");

        WebElement signIn =
        driver.findElement(By.id("signInSubmit"));
        signIn.click();

        WebElement accountListAfterLogin =
        driver.findElement(By.id("nav-link-accountList"));
        boolean isUserSignedIn =
        accountListAfterLogin.getText().contains("YourAccount");

        if (isUserSignedIn) {
            System.out.println("Successful Signin");
        } else {
            System.out.println("Invalid entry");
        }
    }

    @Test

    public void searchProductAndCheckInventory() {
        //Locate the search input field and enter the product name
        WebElement searchBox =
        driver.findElement(By.id("twotabsearchtextbox"));
        searchBox.sendKeys("Laptop");

        //Locate the search icon and click
    }
}

```

```

        WebElement searchIcon=driver.findElement(By.id("nav-
search-submit-button"));
        searchIcon.click();

        //Waitforthesearchresultspagetoload(Youmightneed to enhance
this wait strategy)
        try{
            Thread.sleep(2000);
        }catch(InterruptedException e){ e.printStackTrace();
        }

        //Assumingthefirstsearchresultistheproductpage WebElement
        firstSearchResult =
driver.findElement(By.cssSelector("div.s-main-slotdiv.s-asin:nth-
child(1)"));
        firstSearchResult.click();

        //Waitfortheproductdetailspagetoload(Youmightneed to enhance
this wait strategy)
        try{
            Thread.sleep(2000);
        }catch(InterruptedException e){ e.printStackTrace();
        }

        //Simulateextractingtheinventoryinformationfromthe web page
        WebElement inventoryElement =
driver.findElement(By.id("availability"));
        String inventoryText=inventoryElement.getText();
        System.out.println("Inventory:"+inventoryText);

        //Example:Assertthattheinventorytextcontainsthe expected
quantity information
        Assert.assertTrue(inventoryText.contains("InStock"));

        //Youcanaddmoreassertionsbasedonyourapplication's actual
behavior
    }

    @AfterTest
    public void tearDown() {
        //Close the browser driver.quit();
    }
}

```

### AmazonWebserviceTerminalcommand:-

```

sudo su-
#yum install git-y
#sudo amazon-linux-extras install java-openjdk11-y

```

```
sudo wget -O /etc/yum.repos.d/jenkins.repo
https://pkg.jenkins.io/redhat-stable/jenkins.repo
sudo rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io-
2023.key
yum install jenkins -y
systemctl start Jenkins

cat /var/lib/jenkins/secrets/initialAdminPassword
```

GitHub link:

<https://github.com/chandana161/phase2-lesson1-projects-.git>