Source code:-

FlipkartTestChrome.java

```
package com.demo;

import org.testng.annotations.AfterClass;
import org.testng.annotations.Test;
import org.testng.annotations.BeforeClass;
import org.testng.annotations.AfterClass;
import org.testng.annotations.Test;
import org.testng.annotations.BeforeClass;
import org.testng.annotations.Test;

import java.io.File;
import java.io.IOException;
import java.util.concurrent.TimeUnit;
import java.util.function.Function;

import org.apache.commons.io.FileUtils;
import org.openqa.selenium.By;
import org.openqa.selenium.Dimension;
import org.openqa.selenium.JavascriptExecutor;
import org.openqa.selenium.NoSuchElementException;
import org.openqa.selenium.OutputType;
import org.openqa.selenium.TakesScreenshot;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.support.ui.FluentWait;
import org.openqa.selenium.support.ui.Wait;
import org.testng.annotations.AfterClass;
import org.testng.annotations.BeforeClass;
import org.openqa.selenium.TakesScreenshot;

public class FlipkartTestChrome {
	WebDriver driver;

	@BeforeClass
	public void beforeClass() {
```

```
System.setProperty("webdriver.chrome.driver","F:\\chromedriver-win64\\chromedriver.exe");
		driver = new ChromeDriver();
		driver.get("https://www.flipkart.com/");
		driver.manage().window().maximize();
	}

	@AfterClass
	public void afterClass() {
		driver = null;
	}

	@Test(priority = 1)
	public void closeLogin() throws InterruptedException {

		try {
			System.out.println("\nChrome Browser Result:\n");
			System.out.println(driver.getTitle());
			driver.findElement(By.cssSelector("body > div._2Sn47c > div > div >
button")).click();
			Thread.sleep(1000);
		} catch (NoSuchElementException e) {
			e.printStackTrace();
		}
		screenshot(driver,"closelogin");
	}

	@Test(priority = 2)
	public void scroll() throws InterruptedException {

		Thread.sleep(2000);
		JavascriptExecutor js = (JavascriptExecutor) driver;
		js.executeScript("window.scrollBy(0,document.body.scrollHeight)");
		System.out.println("\nNavigated to bottom of the page");
		Thread.sleep(2000);
		js.executeScript("window.scrollBy(0,-document.body.scrollHeight)", "");
		System.out.println("\nScroll Feature available");
		Thread.sleep(2000);
		screenshot(driver,"scroll");
	}

	@Test(priority = 3)
	public void searchProduct() throws InterruptedException {
		Thread.sleep(1000);
```

```
		driver.findElement(By.name("q")).sendKeys("iPhone 13");
		Thread.sleep(1000);
		By search = By.cssSelector(
				"#container > div > div._1kfTjk > div._1rH5Jn > div._2Xfa2_ >
div._1cmsER > form > div > button > svg");
		driver.findElement(search).click();

		Thread.sleep(3000);
		By load = By.cssSelector(
				"#container > div > div._36fx1h._6t1WkM._3HqJxg >
div._1YokD2._2GoDe3 > div:nth-child(2) > div:nth-child(9) > div > div");
		long start = System.currentTimeMillis();
		driver.findElement(load).click();
		long finish = System.currentTimeMillis();
		long totalTime = finish - start;
		System.out.println("\nTime to load page in millisecs - " + totalTime);
		screenshot(driver,"searchproduct");
	}

	@Test(priority = 4)
	public void loadImage() throws InterruptedException {

		String url =
"https://www.flipkart.com/apple-iphone-13-blue-256-gb/p/itmd68a015aa1e39?pid=MOBG6VF56
6ZTUVFR&lid=LSTMOBG6VF566ZTUVFR2RQLVU&marketplace=FLIPKART&q=iPhone+13&st
ore=tyy%2F4io&srno=s_1_8&otracker=search&otracker1=search&fm=organic&iid=1c0c7402-fe
4f-4f45-9aa8-cc59dffe8503.MOBG6VF566ZTUVFR.SEARCH&ppt=hp&ppn=homepage&ssid=i4
t60bsv4g0000001665375424769&qH=c3d519be0191fbf8";
		driver.get(url);
		Thread.sleep(3000);
		//driver.navigate().refresh();

		Wait<WebDriver> wait = new FluentWait<WebDriver>(driver).withTimeout(10,
TimeUnit.SECONDS)
				.pollingEvery(2,
TimeUnit.SECONDS).ignoring(NoSuchElementException.class);
		wait.until(new Function<WebDriver, WebElement>() {

			@Test
			public WebElement apply(WebDriver driver) {
				WebElement img = driver.findElement(By.xpath(

"//*[@id=\"container\"]/div/div[3]/div[1]/div[2]/div[9]/div[4]/div[3]/div/div/div/div[1]/img"));
```

```
                        if (img.isDisplayed()) {
                                System.out.println("\nImage Loaded");
                                return img;
                        } else {
                                System.out.println("\nFluent Wait Fail!, Element Not
Loaded Yet");
                                return null;
                        }
                }
        });
        screenshot(driver,"pageLoad");
    }

    @Test(priority = 5)
    public void scrollFrequency() throws InterruptedException {
        Thread.sleep(2000);
        long start = System.currentTimeMillis();
        WebElement element = driver.findElement(By.cssSelector(
                "#container > div > div._2c7YLP.UtUXW0._6t1WkM._3HqJxg >
div._1YokD2._2GoDe3 > div._1YokD2._3Mn1Gg.col-8-12 > div._1YokD2._3Mn1Gg >
div:nth-child(7) > div > div:nth-child(3) > div > div > div:nth-child(1)"));
        ((JavascriptExecutor) driver).executeScript("arguments[0].scrollIntoView(true);",
element);
        long stop = System.currentTimeMillis();
        long totalTime = stop - start;
        System.out.println("\nScroll Frequency in millisecs - " + totalTime);
        screenshot(driver,"scrollfrequency");

    }

    @Test(priority = 6)
    public void downloadImages() throws InterruptedException {
        WebElement img = driver.findElement(By
                .xpath("//*[@id=\"container\"]/div/div[3]/div[1]/div[1]"));
        Boolean p = (Boolean) ((JavascriptExecutor) driver).executeScript("return
arguments[0].complete "
                + "&& typeof arguments[0].naturalWidth != \"undefined\" " + "&&
arguments[0].naturalWidth > 0", img);

        if (p) {
            System.out.println("\nImage present");
        } else {
            System.out.println("\nImage not present");
        }
```

```
            screenshot(driver,"downloadImages");
        }

        @Test(priority = 7)
        public void screenResolution() throws InterruptedException {
            Thread.sleep(1000);
            Dimension dimension = new Dimension(720, 1080);
            driver.manage().window().setSize(dimension);
            Thread.sleep(3000);

            Dimension dimension1 = new Dimension(1280, 800);
            driver.manage().window().setSize(dimension1);
            Thread.sleep(3000);

            Dimension dimension2 = new Dimension(2256, 1504);
            driver.manage().window().setSize(dimension2);

            JavascriptExecutor js = (JavascriptExecutor) driver;
            int contentHeight = ((Number) js.executeScript("return
window.innerHeight")).intValue();
            int contentWidth = ((Number) js.executeScript("return
window.innerWidth")).intValue();
            System.out.println("\nHeight: " + contentHeight + " Width: " + contentWidth +
"\n");
            screenshot(driver,"screenshotResolution");

        }

        public static void screenshot(WebDriver driver,String screenshotName){
             TakesScreenshot ts = (TakesScreenshot)driver;
             File scr = ts.getScreenshotAs(OutputType.FILE);
             try {
                        FileUtils.copyFile(scr, new File(screenshotName+".png"));
                        System.out.println("Screenshot taken");
                } catch (IOException e) {
                        e.printStackTrace();
                }
        }


          @AfterClass
            public void tearDown() {
               // Uncomment the following line to close the browser after all tests
                driver.quit();
```

```
        }
}
```

AppTest.java

```
package com.demo;

/**
 * Hello world!
 *
 */
public class App
{
    public static void main( String[] args )
    {
        System.out.println( "Hello World!" );
    }
}
```

Pom.xm

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
https://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>flipkart</groupId>
  <artifactId>flipkart</artifactId>
  <version>0.0.1-SNAPSHOT</version>
   <dependencies>
<dependency>
    <groupId>org.slf4j</groupId>
    <artifactId>slf4j-api</artifactId>
    <version>1.7.32</version> <!-- Use the latest version available -->
```

```
</dependency>

<dependency>
    <groupId>ch.qos.logback</groupId>
    <artifactId>logback-classic</artifactId>
    <version>1.2.6</version> <!-- Use the latest version available -->
</dependency>



</dependencies>


</project>
```