

ALGORITHM FOR PHASE END PROJECT AUTOMATE AN ECOMMERCE WEB APPLICATION

Algorithm: Capture Screenshot

1. Start the WebDriver and create a constructor for the BasePage class to initialize the WebDriver using the DriverManager class.
2. Implement the captureScreenshot method in the BasePage class, taking the name of the screenshot as input.
3. Typecast the WebDriver instance to TakesScreenshot to enable capturing screenshots.
4. Capture the screenshot using the getScreenshotAs(OutputType.FILE) method, obtaining a source file containing the screenshot.
5. Create a destination path with a unique name for the screenshot, using the screenshotCounter to generate unique names.
6. Attempt to create the directories specified in the destination path using Files.createDirectories(destination.getParent()).
7. Copy the source file to the destination path using Files.copy(source.toPath(), destination).
8. If successful, print a message indicating that the screenshot is captured and saved.
9. If any error occurs during the process, catch the IOException and print an error message.

Algorithm: Search for a Product

1. In the FlipkartHomePage class, create a constructor to initialize the WebDriver using the DriverManager class.
2. Implement the getHomepageLoadTime method to get the homepage loading time by navigating to the Flipkart homepage using driver.get(baseUrl).
3. Execute JavaScript to calculate the load time using window.performance.timing.
4. Implement the searchForProduct method to search for a product:

- ❖ Close any login popup if displayed using
`driver.findElement(searchInput).sendKeys(Keys.chord(Keys.ESCAPE)).`
- ❖ Enter the search text using
`driver.findElement(searchInput).sendKeys(productName).`
- ❖ Construct the XPath for the specified product using the provided
`searchOptions` string.
- ❖ Select the product from the search options using
`driver.findElement(By.xpath(searchOptionsProductXpathString)).click().`
- ❖ Click the search button using
`driver.findElement(searchButton).click().`

Algorithm: Search Results Page

1. In the `SearchResultsPage` class, create a constructor to initialize the `WebDriver` using the `DriverManager` class.
2. Implement the `isSearchResultsDisplayed` method to check if search results are displayed by checking if the product results list is empty using `driver.findElements(productResults).isEmpty()`.
3. Implement the `getSearchResults` method to get the list of search results using `driver.findElements(productResults)`.
4. Implement the `scrollToElement` method to scroll to a specific element on the page using JavaScript executor.
5. Implement the `isImageDisplayed` method to check if the product image is displayed in the search results using JavaScript executor.
6. Implement the `hasScrollFeature` method to check if the page has a scroll feature using JavaScript executor.
7. Implement the `scrollToBottom` method to scroll to the bottom of the page using JavaScript executor.
8. Implement methods to get the number of visible products, visible images, and the total number of images using `driver.findElements(productResults)` and `findElements(imageSelector)`.
9. Implement the `isAtBottomOfPage` method to check if the cursor is at the bottom of the page using JavaScript executor.

Algorithm: Create Excel File

1. Create a class CreateExcelFile with the main method.
2. Initialize the workbook using new XSSFWorkbook().
3. Create a new sheet named "Sheet1" using
workbook.createSheet("Sheet1").
4. Add headers "Name", "Age", and "Occupation" to the header row.
5. Create the data array with the sample data.
6. Iterate over the data array and create rows for each entry. For each row, create cells for "Name", "Age", and "Occupation" and populate them with the corresponding data.
7. Save the workbook to a file using workbook.write(fileOut).
8. Catch any IOException that might occur and print the stack trace.

Algorithm: DriverManager

1. Create a class DriverManager with static methods.
2. Implement the initializeDriver method to set up the Chrome driver with desired capabilities using
WebDriverManager.chromedriver().capabilities(options).create().
3. Implement the getDriver method to return the initialized driver instance or initialize it if null.
4. Implement the quitDriver method to quit the driver if it is not null.

Algorithm: FileUtils

1. Create a class FileUtils with a static method readTestData.
2. The method should take the file path and sheet name as inputs.
3. Use Apache POI to read the data from the Excel file and store it in a 2D array.
4. Return the 2D array as the test data.

SOURCE CODE FOR PHASE END PROJECT AUTOMATE AN ECOMMERCE WEB APPLICATION

BasePage.java: (src/main/java/pages)

```
package pages;

import org.openqa.selenium.OutputType;
import org.openqa.selenium.TakesScreenshot;
import org.openqa.selenium.WebDriver;
import utils.DriverManager;
import java.io.File;
import java.io.IOException;
import java.nio.file.Files;
import java.nio.file.Path;
import java.nio.file.Paths;
import java.util.concurrent.atomic.AtomicInteger;

public class BasePage {
    protected WebDriver driver;
    private static AtomicInteger screenshotCounter = new
AtomicInteger(0);

    /**
     * This is the constructor
     */
    public BasePage() {
        this.driver = DriverManager.getDriver();
    }

    /**
     * This method captures a screenshot and saves it with a
unique name
     *
     * @param screenshotName The name of the screenshot
     */
    public void captureScreenshot(String screenshotName) {
        TakesScreenshot ts = (TakesScreenshot) driver;
        File source = ts.getScreenshotAs(OutputType.FILE);

        Path destination = Paths.get("screenshots",
```

```

        "screenshot_" +
screenshotCounter.getAndIncrement() + "_" + screenshotName
+ ".png");

    try {
        Files.createDirectories(destination.getParent());
        Files.copy(source.toPath(), destination);
        System.out.println("Screenshot captured and saved: "
+ destination);
    } catch (IOException e) {
        System.out.println("Failed to capture screenshot: "
+ e.getMessage());
    }
}
}

```

FlipkartHomePage.java: (src/main/java/pages)

```

package pages;

import org.openqa.selenium.By;
import org.openqa.selenium.JavascriptExecutor;
import org.openqa.selenium.Keys;
import org.openqa.selenium.WebDriver;
import utils.DriverManager;

public class FlipkartHomePage {
    private WebDriver driver;

    private final String baseUrl =
"https://www.flipkart.com/";
    private final By searchInput =
By.cssSelector("input[name='q']");
    private final By searchButton =
By.cssSelector("button[type='submit']");
    String searchOptions = "(//div[contains(text(),'"
"PRODUCT_SEARCH_OPTION" + "')]/parent::a/parent::div)[1]";

    /**
     * This is the constructor of the current page
     */
    public FlipkartHomePage() {

```

```
        this.driver = DriverManager.getDriver();
    }

    /**
     * This method gets the home page loading time
     *
     * @return load time in long
     */
    public long getHomepageLoadTime() {
        driver.get(baseUrl);
        System.out.println("Application is launched. ");
        System.out.println("Application URL is: " + baseUrl);

        return (Long) ((JavascriptExecutor)
driver).executeScript(
            "return (window.performance.timing.loadEventEnd -
window.performance.timing.navigationStart);");
    }

    /**
     * This method is used to search for a product
     *
     * @param productName
     * @throws InterruptedException
     */
    public void searchForProduct(String productName) throws
InterruptedException {
        driver.findElement(searchInput).sendKeys(Keys.chord(Ke
ys.ESCAPE));
        System.out.println("Pressed escape key to close the
login popup if it is displayed");

        driver.findElement(searchInput).sendKeys(productName);
        System.out.println("Search text is entered as : " +
productName);

        String searchOptionsProductXpathString =
searchOptions.replaceAll("PRODUCT_SEARCH_OPTION",
        productName.toLowerCase());
        Thread.sleep(5000);
    }
}
```

```

        String searchProduct =
driver.findElement(By.xpath(searchOptionsProductXpathString)).getText();
    driver.findElement(By.xpath(searchOptionsProductXpathString)).click();
    System.out.println("Searched product is selected from
list option as: " + searchProduct);

    Thread.sleep(4000);
    driver.findElement(searchButton).click();
    System.out.println("Search button is clicked on to
find the product results");
}

}

```

SearchResultsPage.java : (src/main/java/pages)

```

package pages;

import org.openqa.selenium.By;
import org.openqa.selenium.JavascriptExecutor;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import utils.DriverManager;

import java.util.List;

public class SearchResultsPage {
    private WebDriver driver;
    private final By productResults =
By.cssSelector("div[data-id]");
    private final By imageSelector = By.xpath(
        "//span[contains(text(), 'results
for')]/ancestor::div[4]/div[starts-
with(@class, '_1AtVbE')]/descendant::img[@loading='eager']")
);

    /**
     * This is the constructor of the current page
     */
    public SearchResultsPage() {

```

```
        this.driver = DriverManager.getDriver();
    }

    /**
     * This method is used to verify if search results are
displayed or not
     *
     * @return !isEmpty - boolean value
     */
    public boolean isSearchResultsDisplayed() {
        boolean isEmpty =
driver.findElements(productResults).isEmpty();
        return !isEmpty;
    }

    /**
     * This method is used to get the search results
     *
     * @return productResultsList
     * @throws InterruptedException
     */
    public List<WebElement> getSearchResults() throws
InterruptedException {
        List<WebElement> productResultsList =
driver.findElements(productResults);
        Thread.sleep(5000);
        return productResultsList;
    }

    /**
     * This method is used to scroll to the element
     *
     * @param element
     * @throws InterruptedException
     */
    public void scrollToElement(WebElement element) throws
InterruptedException {
        Thread.sleep(500);
        ((JavascriptExecutor)
driver).executeScript("arguments[0].scrollIntoView(true);"
, element);
```

```
}

/**
 * This method is used to check if the product image is
displayed in the search
 * results
 *
 * @param product
 * @return
 * @throws InterruptedException
 */
public boolean isImageDisplayed(WebElement product)
throws InterruptedException {
    Thread.sleep(1000);
    return (Boolean) ((JavascriptExecutor)
driver).executeScript(
        "return arguments[0].complete && typeof
arguments[0].naturalWidth != 'undefined' &&
arguments[0].naturalWidth > 0;",
        product.findElement(imageSelector));
}

/**
 * This method is used to check if the page has scroll
feature once the results
 * are displayed upon search
 *
 * @return
 */
public boolean hasScrollFeature() {
    return (boolean) ((JavascriptExecutor) driver)
        .executeScript("return
document.documentElement.scrollHeight >
document.documentElement.clientHeight;");
}

/**
 * This method is used to scroll to the bottom of the
page
 */
public void scrollToBottom() {
```

```
((JavascriptExecutor)
driver).executeScript("window.scrollTo(0,
document.body.scrollHeight);");
}

/**
 * This method is used to get the number of visible
products
*
* @return product size
*/
public int getNumberOfVisibleProducts() {
    return driver.findElements(productResults).size();
}

/**
 * This method is used to get the number of visible
images
*
* @return images count
*/
public int getNumberOfVisibleImages() {
    List<WebElement> visibleImages =
driver.findElements(imageSelector);
    return (int)
visibleImages.stream().filter(WebElement::isDisplayed).cou
nt();
}

/**
 * This method is used to get the total number of images
*
* @return images size
*/
public int getTotalNumberOfImages() {
    return driver.findElements(imageSelector).size();
}

/**
 * This method is used to verify if the cursor at the
bottom page
*
```

```

    * @return
    */
public boolean isAtBottomOfPage() {
    double scrollY = (Double) ((JavascriptExecutor)
driver).executeScript("return window.scrollY;");
    long innerHeight = (Long) ((JavascriptExecutor)
driver).executeScript("return window.innerHeight;");
    long scrollHeight = (Long) ((JavascriptExecutor)
driver)
        .executeScript("return
document.documentElement.scrollHeight;");
    return scrollY + innerHeight >= scrollHeight;
}

}

```

CreateExcelFile.java : (src/main/java/utils)

```

package utils;

import org.apache.poi.ss.usermodel.*;
import org.apache.poi.xssf.usermodel.XSSFWorkbook;

import java.io.FileOutputStream;
import java.io.IOException;

public class CreateExcelFile {
    public static void main(String[] args) {
        try (Workbook workbook = new XSSFWorkbook()) {
            Sheet sheet = workbook.createSheet("Sheet1");
            Row headerRow = sheet.createRow(0);

            String[] headers = {"Name", "Age",
"Occupation"};

            for (int i = 0; i < headers.length; i++) {
                Cell cell = headerRow.createCell(i);
                cell.setCellValue(headers[i]);
            }

            String[][] data = {
                {"John Doe", "30", "Engineer"},


```

```

        {"Jane Smith", "25", "Teacher"},  

        {"Michael Brown", "40", "Doctor"}  

    };  
  

    for (int i = 0; i < data.length; i++) {  

        Row dataRow = sheet.createRow(i + 1);  
  

        for (int j = 0; j < data[i].length; j++) {  

            Cell cell = dataRow.createCell(j);  

            cell.setCellValue(data[i][j]);  

        }  

    }  
  

    try (FileOutputStream fileOut = new  

FileOutputStream("testdata.xlsx")) {  

    workbook.write(fileOut);  

}  
  

    System.out.println("Excel file created  

successfully!");  

} catch (IOException e) {  

    e.printStackTrace();  

}  

}  

}
}

```

DriverManager.java : (src/main/java/utils)

```

package utils;  
  

import io.github.bonigarcia.wdm.WebDriverManager;  

import org.openqa.selenium.WebDriver;  

import org.openqa.selenium.chrome.ChromeOptions;  

import org.openqa.selenium.remote.CapabilityType;  

import org.openqa.selenium.remote.DesiredCapabilities;  
  

public class DriverManager {  
  

    private static WebDriver driver;  
  

    /**

```

```
* This method is used to initialize the driver for
chrome driver
*/
public static void initializeDriver() {
    ChromeOptions options = new ChromeOptions();
    options.addArguments("--incognito");
    options.addArguments("--disable-popup-blocking");
    options.setAcceptInsecureCerts(true);
    options.addArguments("--remote-allow-origins=*");
    options.addArguments("--start-maximized");

    DesiredCapabilities capabilities = new
DesiredCapabilities();
    capabilities.setJavascriptEnabled(true);
    capabilities.setCapability(CapabilityType.SUPPORTS_ALE
RTS, true);
    capabilities.setCapability(CapabilityType.SUPPORTS_LOC
ATION_CONTEXT, true);
    capabilities.setCapability(CapabilityType.SUPPORTS_APP
PLICATION_CACHE, true);
    capabilities.setCapability(ChromeOptions.CAPABILITY,
options);

    driver =
WebDriverManager.chromedriver().capabilities(options).crea
te();
    System.out.println("Chrome Browser is launched");

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

}

public static WebDriver getDriver() {
    if (driver == null) {
        initializeDriver();
    }
    return driver;
}

/**
 * This method quits/close the driver if it is not null
 */
```

```
    public static void quitDriver() {
        if (driver != null) {
            driver.quit();
            driver = null;
        }
    }
}
```

FileUtils.java : (src/main/java/utils)

```
package utils;

import java.io.FileInputStream;
import java.io.IOException;
import org.apache.poi.ss.usermodel.Sheet;
import org.apache.poi.ss.usermodel.Workbook;
import org.apache.poi.xssf.usermodel.XSSFWorkbook;

public class FileUtils {
    public static Object[][] readTestData(String filePath,
String sheetName) {
        Object[][] testData = null;
        try (FileInputStream fis = new
FileInputStream(filePath); Workbook workbook = new
XSSFWorkbook(fis)) {

            Sheet sheet = workbook.getSheet(sheetName);
            int rowCount = sheet.getLastRowNum();
            int colCount = sheet.getRow(0).getLastCellNum();
            testData = new Object[rowCount][colCount];

            for (int i = 0; i < rowCount; i++) {
                for (int j = 0; j < colCount; j++) {
                    testData[i][j] = sheet.getRow(i +
1).getCell(j).toString();
                }
            }
        } catch (IOException e) {
            e.printStackTrace();
        }
        return testData;
    }
}
```

```
}
```

FlipkartLazyLoadingTest.java : (src/test/java/tests)

```
package tests;

import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeOptions;
import org.openqa.selenium.firefox.FirefoxOptions;
import org.openqa.selenium.remote.RemoteWebDriver;
import org.testng.Assert;
import org.testng.annotations.AfterClass;
import org.testng.annotations.BeforeClass;
import org.testng.annotations.Test;
import pages.*;
import utils.DriverManager;

import java.net.URL;
import java.util.List;

public class FlipkartLazyLoadingTest {
    private FlipkartHomePage HomePage;
    private SearchResultsPage searchResultsPage;

    @BeforeClass
    public void setup() {
        DriverManager.initializeDriver();
        HomePage = new FlipkartHomePage();
        searchResultsPage = new SearchResultsPage();
    }

    private void captureScreenshot(String screenshotName) {
        BasePage basePage = new BasePage();
        basePage.captureScreenshot(screenshotName);
    }

    @Test(priority = 1, enabled = true, description =
    "Verify the flipkart homepage load time", groups = {
        "regression" })
    public void testFlipkartHomepageLoadTime() {
```

```
long loadTime = homepage.getHomepageLoadTime();
Assert.assertTrue(loadTime < 8000, "Homepage load time
is within acceptable range.");
}

@Test(priority = 2, enabled = true, description =
"Verify the functionality of search product in mobile
category", groups = {
    "regression"})
public void testSearchProductInMobileCategory() throws
InterruptedException {
    String product = "iPhone 13";
//    homepage.searchForProduct(product, "Mobile");
    homepage.searchForProduct(product);
    Assert.assertTrue(searchResultsPage.isSearchResultsDis
played(), "Search results are displayed.");
}

@Test(priority = 3, enabled = true, description =
"Verify the functionality of image loading and
visibility", groups = {
    "regression"})
public void testImageLoadingAndVisibility() throws
InterruptedException {
    List<WebElement> products =
searchResultsPage.getSearchResults();
    for (WebElement product : products) {
        try {
            searchResultsPage.scrollToElement(product);
            Thread.sleep(3000);
            boolean isImageDisplayed =
searchResultsPage.isImageDisplayed(product);
            Assert.assertTrue(isImageDisplayed, "Image is
loaded and visible.");

            // Capture screenshot after verifying the image
            visibility
            captureScreenshot("image_visible_" +
products.indexOf(product));
        } catch (Exception e) {
            // Handle any exceptions that might occur during
            scrolling or image verification
        }
    }
}
```

```
        e.printStackTrace();
    }
}
}

@Test(priority = 4, enabled = true, description =
"Verify the flipkart homepage load time", groups = {
    "regression"})
public void testScrollFeatureAndContentRefresh() {
    boolean hasScrollFeature =
searchResultsPage.hasScrollFeature();
    Assert.assertTrue(hasScrollFeature, "The page has a
scroll feature.");

    int initialProductCount =
searchResultsPage.getNumberOfVisibleProducts();
    searchResultsPage.scrollToBottom();
    int finalProductCount =
searchResultsPage.getNumberOfVisibleProducts();

    Assert.assertTrue(finalProductCount ==
initialProductCount, "Content is refreshed while
scrolling.");
}

@Test(priority = 5, enabled = true, description =
"Verify the lazy loading functionality", groups = {
    "regression"})
public void testLazyLoading() {
    int visibleImageCount =
searchResultsPage.getNumberOfVisibleImages();
    System.out.println("visibleImageCount is: " +
visibleImageCount);

    int totalImageCount =
searchResultsPage.getTotalNumberOfImages();
    System.out.println("totalImageCount is: " +
totalImageCount);

    Assert.assertTrue((visibleImageCount > 0) &&
(visibleImageCount == totalImageCount),
```

```
        "Images are lazily loaded as the user scrolls
down.");
    }

    @Test(priority = 6, enabled = true, description =
"Verify the functionality of navigating to bottom", groups
= {
        "regression" })
    public void testNavigationToBottom() {
        searchResultsPage.scrollToBottom();
        boolean isAtBottom =
searchResultsPage.isAtBottomOfPage();

        Assert.assertTrue(isAtBottom, "Successfully navigated
to the bottom of the page.");
    }

    @Test(priority = 7, enabled = true, description =
"Verify the functionality of cross browser and screen
resolution", groups = {
        "regression" })
    public void testCrossBrowserAndScreenResolution() {
        String[] browsers = { "chrome", "firefox" };
        String[] screenResolutions = { "1920x1080", "1366x768"
};

        for (String browser : browsers) {
            for (String resolution : screenResolutions) {
                try {
                    // Set the URL of the Selenium Grid hub
                    URL gridUrl = new URL("http://your-grid-hub-
url:4444/wd/hub");

                    // Create the WebDriver instance with the
desired capabilities
                    WebDriver driver =
createRemoteWebDriver(browser, resolution, gridUrl);

                    // Execute your test steps here with the
'driver' instance
                    // You may reuse the existing methods from the
SearchResultsPage class
                }
            }
        }
    }
}
```

```

        // Close the browser after the test is done
        driver.quit();
    } catch (Exception e) {
        e.printStackTrace();
    }
}
}

private WebDriver createRemoteWebDriver(String browser,
String resolution, URL gridUrl) {
    WebDriver driver = null;

    if ("chrome".equalsIgnoreCase(browser)) {
        ChromeOptions options = new ChromeOptions();
        options.addArguments("--window-size=" + resolution);
        driver = new RemoteWebDriver(gridUrl, options);
    } else if ("firefox".equalsIgnoreCase(browser)) {
        FirefoxOptions options = new FirefoxOptions();
        options.addArguments("--window-size=" + resolution);
        driver = new RemoteWebDriver(gridUrl, options);
    }

    return driver;
}

@AfterClass
public void tearDown() {
    DriverManager.quitDriver();
}
}

```

log4j.properties : (src/test/resources)

```

# Root logger option
log4j.rootLogger=INFO, file

# Define the appender
log4j.appender.file=org.apache.log4j.FileAppender
log4j.appender.file.File=logs/app.log
log4j.appender.file.layout=org.apache.log4j.PatternLayout

```

```
log4j.appender.file.layout.ConversionPattern=%d{yyyy-MM-dd  
HH:mm:ss} %-5p %c{1}:%L - %m%n
```

pom.xml :

```
<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>com.example</groupId>  
    <artifactId>FlipkartLazyLoadingProject</artifactI  
d>  
    <version>1.0-SNAPSHOT</version>  
    <name>FlipkartLazyLoadingProject</name>  
    <description>To automate a real-world web  
application</description>  
    <properties>  
        <maven.compiler.source>1.8</maven.compiler.source  
>  
        <maven.compiler.target>1.8</maven.compiler.target  
>  
        <project.build.sourceEncoding>UTF-  
8</project.build.sourceEncoding>  
    </properties>  
    <dependencies>  
        <!--  
            https://mvnrepository.com/artifact/io.github.boni  
garcia/webdrivermanager -->  
        <dependency>  
            <groupId>io.github.bonigarcia</groupId>  
            <artifactId>webdrivermanager</artifactId>  
            <version>5.3.2</version>  
        </dependency>  
        <!-- Selenium -->  
        <!--
```

```
https://mvnrepository.com/artifact/org.seleniumhq
.selenium/selenium-java -->
<dependency>

<groupId>org.seleniumhq.selenium</groupId>
<artifactId>selenium-java</artifactId>
<!--<version>4.10.0</version>-->
<version>4.4.0</version>

</dependency>

<!-- TestNG -->
<!--
https://mvnrepository.com/artifact/org.testng/testng
-->
<dependency>
<groupId>org.testng</groupId>
<artifactId>testng</artifactId>
<!--<version>7.7.1</version>-->
<version>7.5</version>
<!--<scope>test</scope>-->
</dependency>

<!-- Apache POI -->
<dependency>
<groupId>org.apache.poi</groupId>
<artifactId>poi</artifactId>
<version>5.0.0</version>
</dependency>
<dependency>
<groupId>org.apache.poi</groupId>
<artifactId>poi-ooxml</artifactId>
<version>5.0.0</version>
</dependency>
<dependency>
<groupId>ch.qos.logback</groupId>
<artifactId>logback-classic</artifactId>
<version>1.2.6</version>
</dependency>

<dependency>
<groupId>com.automation</groupId>
<artifactId>saucelabs</artifactId>
```

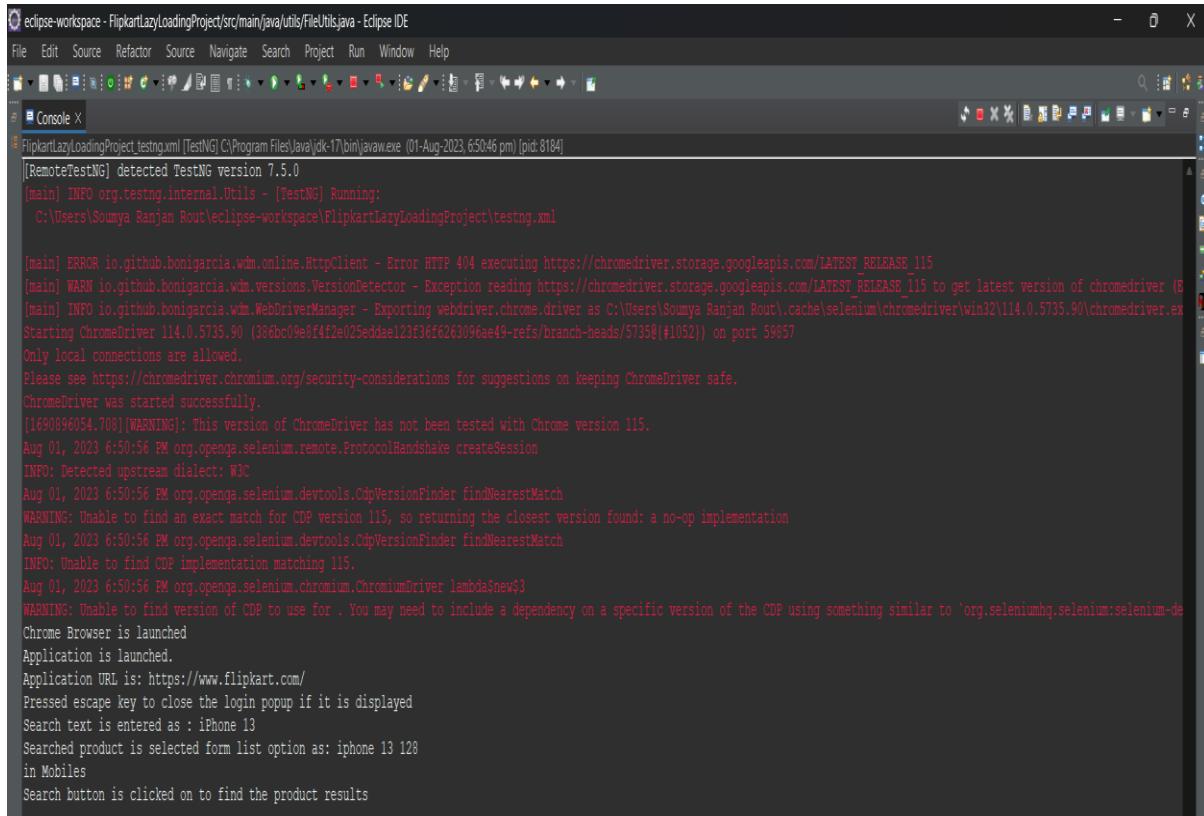
```
        <version>0.0.1-SNAPSHOT</version>
    </dependency>
</dependencies>
</project>
```

testing.xml :

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE suite SYSTEM "https://testng.org/testng-1.0.dtd">
<suite name="Suite">
    <test thread-count="5" name="Test">
        <classes>
            <class name="tests.FlipkartLazyLoadingTest" />
        </classes>
    </test> <!-- Test -->
</suite> <!-- Suite -->
```

OUTPUT SCREENSHOTS FOR PHASE END PROJECT AUTOMATE AN ECOMMERCE WEB APPLICATION

Application is on running state :

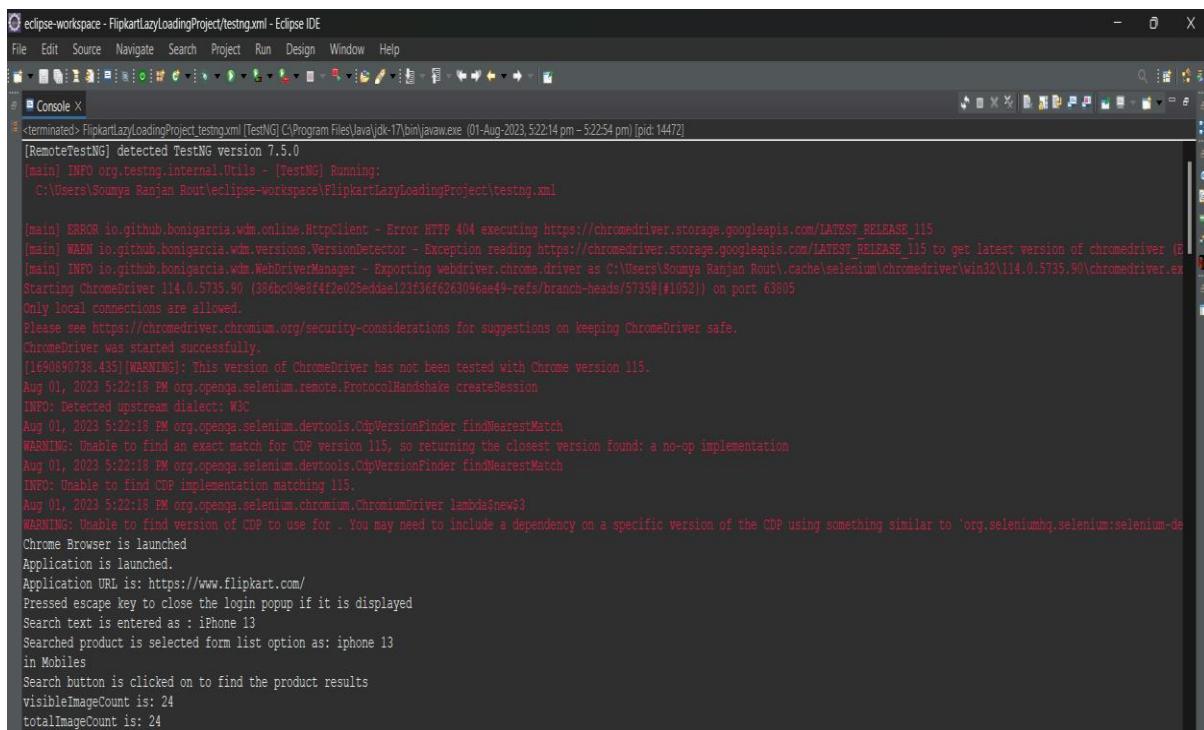


The screenshot shows the Eclipse IDE interface with the title bar "eclipse-workspace - FlipkartLazyLoadingProject/src/main/java/utils/FileUtils.java - Eclipse IDE". The menu bar includes File, Edit, Source, Refactor, Source, Navigate, Search, Project, Run, Window, Help. The toolbar has various icons for file operations. The main window shows the "Console" tab with the following log output:

```
[RemoteTestNG] detected TestNG version 7.5.0
[main] INFO org.testng.internal.Utils - [TestNG] Running:
C:\Users\Soumya Ranjan Rout\workspace\FlipkartLazyLoadingProject\testng.xml

[main] ERROR io.github.bonigarcia.wdm.online.HttpClient - Error HTTP 404 executing https://chromedriver.storage.googleapis.com/LATEST_RELEASE_115
[main] WARN io.github.bonigarcia.wdm.versions.VersionDetector - Exception reading https://chromedriver.storage.googleapis.com/LATEST_RELEASE_115 to get latest version of chromedriver (E)
[main] INFO io.github.bonigarcia.wdm.WebDriverManager - Exporting webdriver.chrome.driver as C:\Users\Soumya Ranjan Rout\.cache\selenium\chromedriver\win32\114.0.5735.90\chromedriver.exe
Starting ChromeDriver 114.0.5735.90 (386bc09e8f4f2e025eddae123f36f6263096ae49-refs/branch-heads/5735#(1052)) on port 59857
Only local connections are allowed.
Please see https://chromedriver.chromium.org/security-considerations for suggestions on keeping ChromeDriver safe.
ChromeDriver was started successfully.
[1690896054.708] [WARNING]: This version of ChromeDriver has not been tested with Chrome version 115.
Aug 01, 2023 6:50:56 PM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected upstream dialect: W3C
Aug 01, 2023 6:50:56 PM org.openqa.selenium.devtools.CdpVersionFinder findNearestMatch
WARNING: Unable to find an exact match for CDP version 115, so returning the closest version found: a no-op implementation
Aug 01, 2023 6:50:56 PM org.openqa.selenium.devtools.CdpVersionFinder findNearestMatch
INFO: Unable to find CDP implementation matching 115.
Aug 01, 2023 6:50:56 PM org.openqa.selenium.chromium.ChromiumDriver lambda$new$3
WARNING: Unable to find version of CDP to use for . You may need to include a dependency on a specific version of the CDP using something similar to 'org.seleniumhq.selenium:selenium-de
Chrome Browser is launched
Application is launched.
Application URL is: https://www.flipkart.com/
Pressed escape key to close the login popup if it is displayed
Search text is entered as : iPhone 13
Searched product is selected from list option as: iphone 13 128
in Mobiles
Search button is clicked on to find the product results
```

Application automation using Chrome Driver :



The screenshot shows the Eclipse IDE interface with the title bar "eclipse-workspace - FlipkartLazyLoadingProject/testng.xml - Eclipse IDE". The menu bar includes File, Edit, Source, Navigate, Search, Project, Run, Design, Window, Help. The toolbar has various icons for file operations. The main window shows the "Console" tab with the following log output:

```
[terminated] - FlipkartLazyLoadingProject\testng.xml [TestNG] C:\Program Files\Java\jdk-17\bin\javaw.exe (01-Aug-2023, 5:22:14 pm - 5:22:54 pm) [pid: 14472]
[RemoteTestNG] detected TestNG version 7.5.0
[main] INFO org.testng.internal.Utils - [TestNG] Running:
C:\Users\Soumya Ranjan Rout\workspace\FlipkartLazyLoadingProject\testng.xml

[main] ERROR io.github.bonigarcia.wdm.online.HttpClient - Error HTTP 404 executing https://chromedriver.storage.googleapis.com/LATEST_RELEASE_115
[main] WARN io.github.bonigarcia.wdm.versions.VersionDetector - Exception reading https://chromedriver.storage.googleapis.com/LATEST_RELEASE_115 to get latest version of chromedriver (E)
[main] INFO io.github.bonigarcia.wdm.WebDriverManager - Exporting webdriver.chrome.driver as C:\Users\Soumya Ranjan Rout\.cache\selenium\chromedriver\win32\114.0.5735.90\chromedriver.exe
Starting ChromeDriver 114.0.5735.90 (386bc09e8f4f2e025eddae123f36f6263096ae49-refs/branch-heads/5735#(1052)) on port 63805
Only local connections are allowed.
Please see https://chromedriver.chromium.org/security-considerations for suggestions on keeping ChromeDriver safe.
ChromeDriver was started successfully.
[1690890738.435] [WARNING]: This version of ChromeDriver has not been tested with Chrome version 115.
Aug 01, 2023 5:22:18 PM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected upstream dialect: W3C
Aug 01, 2023 5:22:18 PM org.openqa.selenium.devtools.CdpVersionFinder findNearestMatch
WARNING: Unable to find an exact match for CDP version 115, so returning the closest version found: a no-op implementation
Aug 01, 2023 5:22:18 PM org.openqa.selenium.devtools.CdpVersionFinder findNearestMatch
INFO: Unable to find CDP implementation matching 115.
Aug 01, 2023 5:22:18 PM org.openqa.selenium.chromium.ChromiumDriver lambda$new$3
WARNING: Unable to find version of CDP to use for . You may need to include a dependency on a specific version of the CDP using something similar to 'org.seleniumhq.selenium:selenium-de
Chrome Browser is launched
Application is launched.
Application URL is: https://www.flipkart.com/
Pressed escape key to close the login popup if it is displayed
Search text is entered as : iPhone 13
Searched product is selected from list option as: iphone 13 128
in Mobiles
Search button is clicked on to find the product results
visibleImageCount is: 24
totalImageCount is: 24
```

Console output :

The screenshot shows the Eclipse IDE interface with the 'Console' tab selected. The output window displays the stack trace of a failed test, followed by a summary of the test run.

```
terminated> FlipkartLazyLoadingProject_testing.xml [TestNG] C:\Program Files\Java\jdk-17\bin\javaw.exe (01-Aug-2023, 522:14 pm - 522:54 pm) [pid: 14472]
at org.openqa.selenium.remote.http.Filter.lambda$andFinally$1 [Filter.java:56]
at org.openqa.selenium.remote.http.netty.NettyHttpHandler.execute [NettyHttpHandler.java:49]
at org.openqa.selenium.remote.http.AddSeleniumUserAgent.lambda$apply$0 [AddSeleniumUserAgent.java:42]
at org.openqa.selenium.remote.http.Filter.lambda$andFinally$1 [Filter.java:56]
at org.openqa.selenium.remote.http.netty.NettyClient.execute [NettyClient.java:98]
at org.openqa.selenium.remote.tracing.TracedHttpClient.execute [TracedHttpClient.java:55]
at org.openqa.selenium.remote.ProtocolHandshake.createSession [ProtocolHandshake.java:120]
at org.openqa.selenium.remote.ProtocolHandshake.createSession [ProtocolHandshake.java:102]
at org.openqa.selenium.remote.ProtocolHandshake.createSession [ProtocolHandshake.java:67]
at org.openqa.selenium.remote.HttpCommandExecutor.execute [HttpCommandExecutor.java:156]
at org.openqa.selenium.remote.TracedCommandExecutor.execute [TracedCommandExecutor.java:51]
at org.openqa.selenium.remote.RemoteWebDriver.execute [RemoteWebDriver.java:547]
... 33 more
Caused by: java.net.UnknownHostException: your-grid-hub-url
at java.base/java.net.InetAddressCacheAddresses.get [InetAddress.java:801]
at java.base/java.net.InetAddress.getAllByName0 [InetAddress.java:1532]
at java.base/java.net.InetAddress.getAllByName [InetAddress.java:1384]
at java.base/java.net.InetAddress.getAllByName [InetAddress.java:1305]
at io.netty.util.internal.SocketUtils$9.run [SocketUtils.java:169]
at io.netty.util.internal.SocketUtils$9.run [SocketUtils.java:166]
at java.base/java.security.AccessController.doPrivileged [AccessController.java:569]
at io.netty.util.internal.SocketUtils.allAddressesByName [SocketUtils.java:166]
at io.netty.resolver.DefaultNameResolver.doResolveAll [DefaultNameResolver.java:50]
at io.netty.resolver.SimpleNameResolver.resolveAll [SimpleNameResolver.java:79]
at io.netty.resolver.SimpleNameResolver.resolveAll [SimpleNameResolver.java:71]
at org.asynchttpclient.resolver.RequestHostnameResolver.resolve [RequestHostnameResolver.java:50]
at org.asynchttpclient.netty.request.NettyRequestSender.resolveAddresses [NettyRequestSender.java:364]
at org.asynchttpclient.netty.request.NettyRequestSender.sendRequestWithNewChannel [NettyRequestSender.java:307]
at org.asynchttpclient.netty.request.NettyRequestSender.sendRequestWithCertainForceConnect [NettyRequestSender.java:143]
at org.asynchttpclient.netty.request.NettyRequestSender.sendRequest [NettyRequestSender.java:114]
at org.asynchttpclient.DefaultAsyncHttpClient.execute [DefaultAsyncHttpClient.java:259]
at org.asynchttpclient.DefaultAsyncHttpClient.executeRequest [DefaultAsyncHttpClient.java:228]
at org.asynchttpclient.DefaultAsyncHttpClient.executeRequest [DefaultAsyncHttpClient.java:249]
at org.openqa.selenium.remote.http.netty.NettyHttpHandler.makeCall [NettyHttpHandler.java:55]
... 46 more
```

=====

Suite
Total tests run: 7, Passes: 7, Failures: 0, Skips: 0

=====

TestNG results :

The screenshot shows the Eclipse IDE interface with the 'Results of running suite' tab selected. The results summary indicates 7 passed tests, 0 failed tests, and 0 skipped tests. Below the summary, a tree view shows the test structure and execution times.

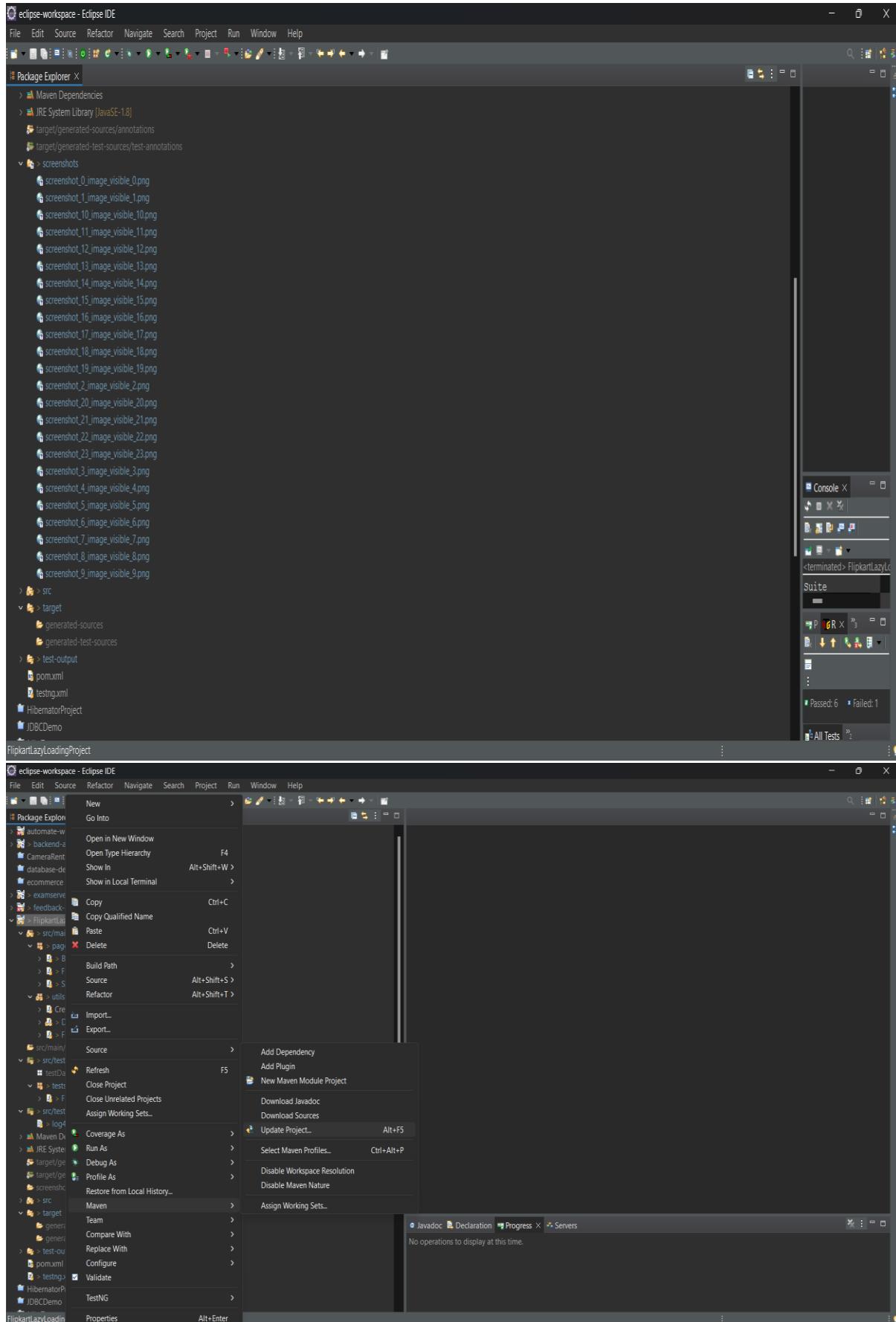
Passed: 7 Failed: 0 Skipped: 0 Tests: 1/1 Methods: 7 (38288 ms)

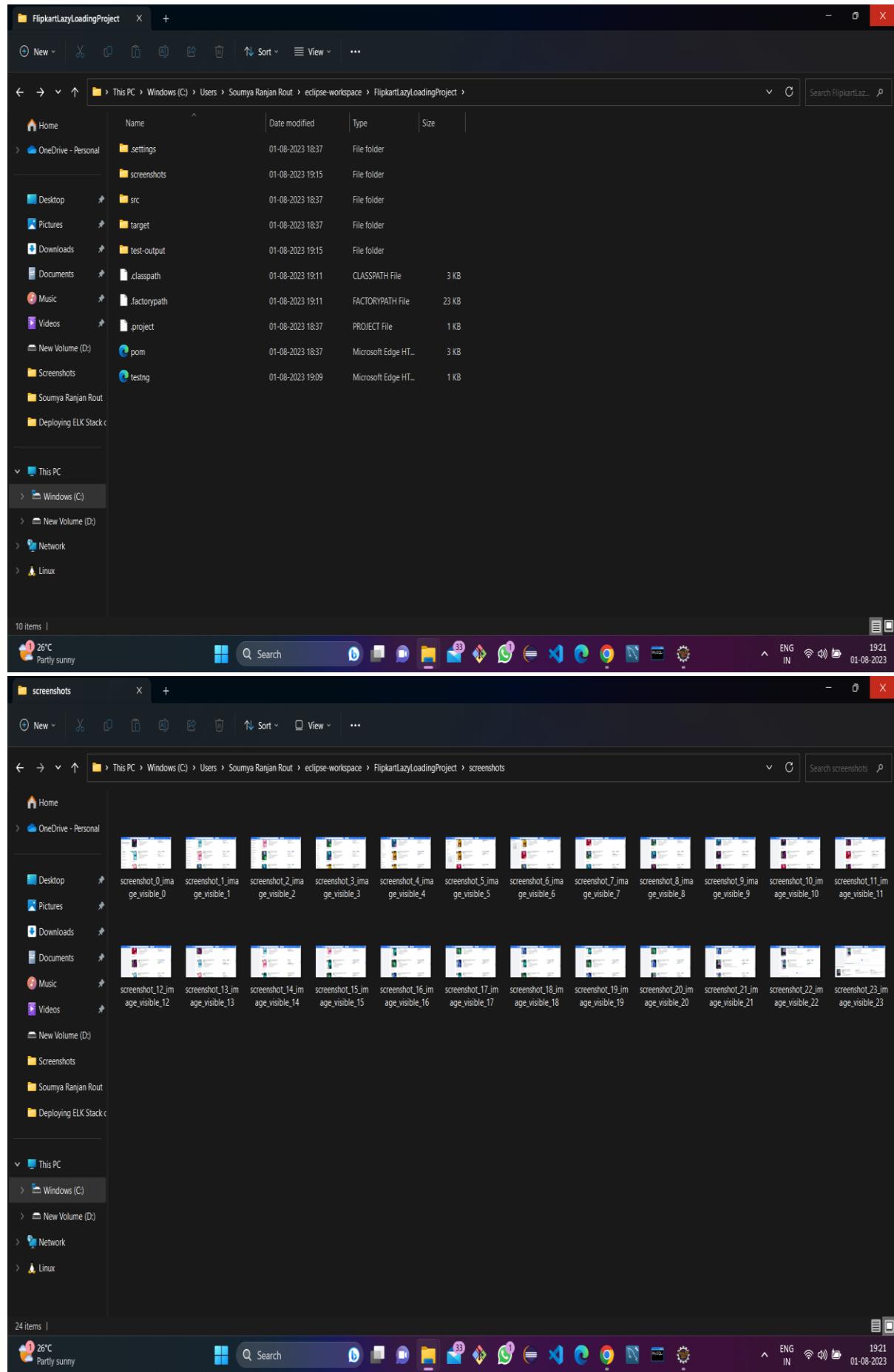
All Tests Failed Tests Summary

- Suite (7/0/0) (34.975 s)
 - Test (34.975 s)
 - tests.FlipkartLazyLoadingTest
 - testFlipkartHomepageLoadTime (3.043 s)
 - testSearchProductInMobileCategory (12.898 s)
 - testImageLoadingAndVisibility (17.427 s)
 - testScrollFeatureAndContentRefresh (0.076 s)
 - testLazyLoading (0.354 s)
 - testNavigationToBottom (0.029 s)
 - testCrossBrowserAndScreenResolution (1.148 s)

Failure Exception

Workspace and directory :





Automated Screenshots taken during run time :

Flipkart Explore Plus

iphone 13 128

PRICE

Min to ₹30,000+

BRAND

APPLE

Assured

CUSTOMER RATINGS

4★ & above

3★ & above

GST INVOICE AVAILABLE

RAM

1GB and Below

INTERNAL STORAGE

4.7 ★ 2,43,452 Ratings & 12,199 Reviews

₹69,900 11% off
Free delivery by Today
Upto ₹38,600 Off on Exchange
No Cost EMI from ₹10,334/month

128 GB ROM
15.49 cm (6.1 inch) Super Retina XDR Display
12MP + 12MP | 12MP Front Camera
A15 Bionic Chip Processor
Brand Warranty for 1 Year

APPLE iPhone 13 (Starlight, 128 GB)

4.7 ★ 2,43,452 Ratings & 12,199 Reviews

₹61,999 Assured

₹69,900 11% off
Free delivery by Today
Daily Saver
Upto ₹38,600 Off on Exchange

128 GB ROM
15.49 cm (6.1 inch) Super Retina XDR Display
12MP + 12MP | 12MP Front Camera
A15 Bionic Chip Processor
Brand Warranty for 1 Year

APPLE iPhone 13 (Pink, 128 GB)

4.7 ★ 2,43,452 Ratings & 12,199 Reviews

₹61,999 Assured

₹69,900 11% off
Free delivery by Today
Daily Saver

128 GB ROM
15.49 cm (6.1 inch) Super Retina XDR Display

Flipkart		iphone 13 128	4.7 ★ 2,43,452 Ratings & 12,199 Reviews	Login	Become a Seller	More	Cart
CUSTOMER RATINGS	^		<ul style="list-style-type: none">128 GB ROM15.49 cm (6.1 inch) Super Retina XDR Display12MP + 12MP 12MP Front CameraA15 Bionic Chip ProcessorBrand Warranty for 1 Year	₹69,900 11% off	Free delivery by Today	Daily Saver	Upto ₹38,600 Off on Exchange
<input type="checkbox"/> 4★ & above							
<input type="checkbox"/> 3★ & above							
GST INVOICE AVAILABLE	▼		<input type="checkbox"/> Add to Compare				
RAM	^						
<input type="checkbox"/> 1GB and Below							
INTERNAL STORAGE	▼		<p>♥ APPLE iPhone 13 (Pink, 128 GB)</p> <p>4.7 ★ 2,43,452 Ratings & 12,199 Reviews</p> <ul style="list-style-type: none">128 GB ROM15.49 cm (6.1 inch) Super Retina XDR Display12MP + 12MP 12MP Front CameraA15 Bionic Chip ProcessorBrand Warranty for 1 Year	₹61,999 	₹69,900 11% off	Free delivery by Today	Daily Saver
BATTERY CAPACITY	▼						
SCREEN SIZE	▼		<input type="checkbox"/> Add to Compare				
PRIMARY CAMERA	▼						
SECONDARY CAMERA	▼						
PROCESSOR BRAND	▼		<p>♥ APPLE iPhone 13 (Green, 128 GB)</p> <p>4.7 ★ 2,43,452 Ratings & 12,199 Reviews</p> <ul style="list-style-type: none">128 GB ROM15.49 cm (6.1 inch) Super Retina XDR Display	₹61,999 	₹69,900 11% off	Free delivery by Today	Daily Saver
SPECIALITY	▼						

Flipkart Explore Plus

iphone 13 128

INTERNAL STORAGE ▾

BATTERY CAPACITY ▾

SCREEN SIZE ▾

PRIMARY CAMERA ▾

SECONDARY CAMERA ▾

PROCESSOR BRAND ▾

SPECIALITY ▾

RESOLUTION TYPE ▾

OPERATING SYSTEM ▾

NETWORK TYPE ▾

SIM TYPE ▾

AVAILABILITY ▾

DISCOUNT ▾

OFFERS ▾

Buy More, Save More

iphone 13 128

4.7/5 2,43,452 Ratings & 12,199 Reviews



- 128 GB ROM
- 15.49 cm (6.1 inch) Super Retina XDR Display
- 12MP + 12MP | 12MP Front Camera
- A15 Bionic Chip Processor
- Brand Warranty for 1 Year

Add to Compare

APPLE iPhone 13 (Green, 128 GB)

4.7/5 2,43,452 Ratings & 12,199 Reviews



- 128 GB ROM
- 15.49 cm (6.1 inch) Super Retina XDR Display
- 12MP + 12MP | 12MP Front Camera
- A15 Bionic Chip Processor
- Brand Warranty for 1 Year

Add to Compare

APPLE iPhone 13 (Blue, 128 GB)

4.7/5 2,43,452 Ratings & 12,199 Reviews



- 128 GB ROM
- 15.49 cm (6.1 inch) Super Retina XDR Display

Add to Compare

₹61,999 

₹69,900 11% off

Free delivery by Today

Daily Saver

Upto ₹38,600 Off on Exchange

iphone 13 128

4.7/5 2,43,452 Ratings & 12,199 Reviews



- 128 GB ROM
- 15.49 cm (6.1 inch) Super Retina XDR Display
- 12MP + 12MP | 12MP Front Camera
- A15 Bionic Chip Processor
- Brand Warranty for 1 Year

Add to Compare

APPLE iPhone 13 (Blue, 128 GB)

4.7/5 2,43,452 Ratings & 12,199 Reviews



- 128 GB ROM
- 15.49 cm (6.1 inch) Super Retina XDR Display
- 12MP + 12MP | 12MP Front Camera
- A15 Bionic Chip Processor
- Brand Warranty for 1 Year

Add to Compare

APPLE iPhone 13 Pro Max (Gold, 128 GB)

4.6/5 1,914 Ratings & 183 Reviews



- 128 GB ROM
- 17.02 cm (6.7 inch) Super Retina XDR Display

₹61,999 

₹69,900 11% off

Free delivery by Today

Daily Saver

Upto ₹38,600 Off on Exchange

iphone 13 128

4.7/5 2,43,452 Ratings & 12,199 Reviews



- 128 GB ROM
- 15.49 cm (6.1 inch) Super Retina XDR Display
- 12MP + 12MP | 12MP Front Camera
- A15 Bionic Chip Processor
- Brand Warranty for 1 Year

Add to Compare

APPLE iPhone 13 Pro Max (Gold, 128 GB)

4.6/5 1,914 Ratings & 183 Reviews



- 128 GB ROM
- 17.02 cm (6.7 inch) Super Retina XDR Display
- 12MP + 12MP + 12MP | 12MP Front Camera
- A15 Bionic Chip Processor
- Brand Warranty for 1 Year

Add to Compare

APPLE iPhone 13 Pro (Gold, 128 GB)

4.7/5 1,471 Ratings & 121 Reviews



- 128 GB ROM
- 15.49 cm (6.1 inch) Super Retina XDR Display

₹1,29,900 

₹60,000 11% off

Free delivery by Today

Daily Saver

Upto ₹35,600 Off on Exchange

Bank Offer

₹1,19,900 

Free delivery by Today

Upto ₹35,600 Off on Exchange

Bank Offer

Flipkart Explore Plus ▾

iphone 13 128

4.0 ▾ 1,714 Ratings & 103 Reviews

Back to top

Free delivery by Today
Upto ₹35,600 Off on Exchange
Bank Offer

OFFERS

- Buy More, Save More
- No Cost EMI
- Special Price

TYPE

CLOCK SPEED

FEATURES

NUMBER OF CORES

Need help?
[Help me decide](#) ▾

APPLE iPhone 13 Pro (Gold, 128 GB) ₹1,19,900
4.7 ★ 1,471 Ratings & 121 Reviews
Free delivery by Today
Upto ₹35,600 Off on Exchange
Bank Offer

Add to Compare

APPLE iPhone 13 ((PRODUCT)RED, 128 GB) ₹61,999
4.7 ★ 2,43,452 Ratings & 12,199 Reviews
₹69,900 11% off
Free delivery by Today
Upto ₹35,600 Off on Exchange
Bank Offer

Add to Compare

APPLE iPhone 13 ((PRODUCT)RED, 128 GB) ₹61,999
4.7 ★ 2,43,452 Ratings & 12,199 Reviews
₹69,900 11% off
Free delivery by Today
Upto ₹35,600 Off on Exchange
Bank Offer

Add to Compare

APPLE iPhone 13 mini (Green, 128 GB) ₹61,999
4.5 ★ 3,448 Ratings & 298 Reviews
₹64,900 4% off
Free delivery by Today
Upto ₹38,600 Off on Exchange

Add to Compare

APPLE iPhone 13 mini (Green, 128 GB) ₹61,999
4.5 ★ 3,448 Ratings & 298 Reviews
₹64,900 4% off
Free delivery by Today
Upto ₹38,600 Off on Exchange
No Cost EMI from ₹1,034/month

Add to Compare

APPLE iPhone 13 mini (Blue, 128 GB) ₹61,999
4.5 ★ 3,448 Ratings & 298 Reviews
₹64,900 4% off
Free delivery by Today
Upto ₹38,600 Off on Exchange

Add to Compare

Flipkart Explore Plus+  [Login](#) [Become a Seller](#) [More](#) 

iphone 13 128


Currently unavailable

4.5 ★ 3,448 Ratings & 298 Reviews [Back to top](#)

- 128 GB ROM
- 13.72 cm (5.4 inch) Super Retina XDR Display
- 12MP + 12MP | 12MP Front Camera
- A15 Bionic Chip Processor
- Brand Warranty for 1 Year

₹64,900 4% off
Free delivery by **Today**
Upto ₹38,600 Off on Exchange
No Cost EMI from ₹10,334/month

Add to Compare

APPLE iPhone 13 mini (Blue, 128 GB)  **₹61,999**

4.5 ★ 3,448 Ratings & 298 Reviews

- 128 GB ROM
- 13.72 cm (5.4 inch) Super Retina XDR Display
- 12MP + 12MP | 12MP Front Camera
- A15 Bionic Chip Processor
- Brand Warranty for 1 Year

₹64,900 4% off
Free delivery by **Today**
Upto ₹38,600 Off on Exchange
No Cost EMI from ₹10,334/month


Currently unavailable

Add to Compare

APPLE iPhone 13 mini (Midnight, 128 GB)  **₹61,999**

4.5 ★ 3,448 Ratings & 298 Reviews

- 128 GB ROM
- 13.72 cm (5.4 inch) Super Retina XDR Display

₹64,900 4% off
Free delivery by **Today**
Upto ₹38,600 Off on Exchange


Currently unavailable

Add to Compare

Flipkart Explore Plus+  [Login](#) [Become a Seller](#) [More](#) 

iphone 13 128


Currently unavailable

4.5 ★ 3,448 Ratings & 298 Reviews [Back to top](#)

- 128 GB ROM
- 13.72 cm (5.4 inch) Super Retina XDR Display
- 12MP + 12MP | 12MP Front Camera
- A15 Bionic Chip Processor
- Brand Warranty for 1 Year

₹64,900 4% off
Free delivery by **Today**
Upto ₹38,600 Off on Exchange
No Cost EMI from ₹10,334/month

Add to Compare

APPLE iPhone 13 mini (Midnight, 128 GB)  **₹61,999**

4.5 ★ 3,448 Ratings & 298 Reviews

- 128 GB ROM
- 13.72 cm (5.4 inch) Super Retina XDR Display
- 12MP + 12MP | 12MP Front Camera
- A15 Bionic Chip Processor
- Brand Warranty for 1 Year

₹64,900 4% off
Free delivery by **Today**
Upto ₹38,600 Off on Exchange
No Cost EMI from ₹10,334/month


Currently unavailable

Add to Compare

APPLE iPhone 13 Pro Max (Graphite, 128 GB)  **₹1,29,900**

4.6 ★ 1,914 Ratings & 183 Reviews

- 128 GB ROM
- 17.02 cm (6.7 inch) Super Retina XDR Display

₹64,900 4% off
Free delivery by **Today**
Upto ₹35,600 Off on Exchange
Bank Offer


Currently unavailable

Add to Compare

Flipkart Explore Plus+  [Login](#) [Become a Seller](#) [More](#) 

iphone 13 128


Currently unavailable

4.5 ★ 3,448 Ratings & 298 Reviews [Back to top](#)

- 128 GB ROM
- 13.72 cm (5.4 inch) Super Retina XDR Display
- 12MP + 12MP | 12MP Front Camera
- A15 Bionic Chip Processor
- Brand Warranty for 1 Year

₹64,900 4% off
Free delivery by **Today**
Upto ₹38,600 Off on Exchange
No Cost EMI from ₹10,334/month

Add to Compare

APPLE iPhone 13 Pro Max (Graphite, 128 GB)  **₹1,29,900**

4.6 ★ 1,914 Ratings & 183 Reviews

- 128 GB ROM
- 17.02 cm (6.7 inch) Super Retina XDR Display
- 12MP + 12MP + 12MP | 12MP Front Camera
- A15 Bionic Chip Processor
- Brand Warranty for 1 Year

₹64,900 4% off
Free delivery by **Today**
Upto ₹35,600 Off on Exchange
Bank Offer


Currently unavailable

Add to Compare

APPLE iPhone 13 mini ((PRODUCT)RED, 128 GB)  **₹61,999**

4.5 ★ 3,448 Ratings & 298 Reviews

- 128 GB ROM
- 13.72 cm (5.4 inch) Super Retina XDR Display

₹64,900 4% off
Free delivery by **Today**
Upto ₹38,600 Off on Exchange


Currently unavailable

Add to Compare

Flipkart Explore Plus iPhone 13 128

1,714 Ratings & 163 Reviews

4.0 ★ Back to top

Currently unavailable

Add to Compare

APPLE iPhone 13 mini ((PRODUCT)RED, 128 GB)

4.5 ★ 3,448 Ratings & 298 Reviews

- 128 GB ROM
- 17.02 cm (6.7 inch) Super Retina XDR Display
- 12MP + 12MP + 12MP | 12MP Front Camera
- A15 Bionic Chip Processor
- Brand Warranty for 1 Year

₹61,999 Assured

₹64,900 4% off

Free delivery by Today

Upto ₹35,600 Off on Exchange

Bank Offer

APPLE iPhone 13 Pro (Graphite, 128 GB)

4.7 ★ 1,471 Ratings & 121 Reviews

- 128 GB ROM
- 13.72 cm (5.4 inch) Super Retina XDR Display
- 12MP + 12MP | 12MP Front Camera
- A15 Bionic Chip Processor
- Brand Warranty for 1 Year

₹1,19,900 Assured

₹64,900 4% off

Free delivery by Today

Upto ₹35,600 Off on Exchange

Bank Offer

Flipkart Explore Plus iPhone 13 128

1,714 Ratings & 163 Reviews

4.0 ★ Back to top

Currently unavailable

Add to Compare

APPLE iPhone 13 Pro (Graphite, 128 GB)

4.7 ★ 1,471 Ratings & 121 Reviews

- 128 GB ROM
- 13.72 cm (5.4 inch) Super Retina XDR Display
- 12MP + 12MP | 12MP Front Camera
- A15 Bionic Chip Processor
- Brand Warranty for 1 Year

₹1,19,900 Assured

₹64,900 4% off

Free delivery by Today

Upto ₹35,600 Off on Exchange

Bank Offer

Flipkart Explore Plus iPhone 13 128

1,714 Ratings & 121 Reviews

4.5 ★ Back to top

Currently unavailable

Add to Compare

APPLE iPhone 13 mini (Starlight, 128 GB)

4.5 ★ 3,448 Ratings & 298 Reviews

- 128 GB ROM
- 13.72 cm (5.4 inch) Super Retina XDR Display

₹61,999 Assured

₹64,900 4% off

Free delivery by Today

Upto ₹35,600 Off on Exchange

Bank Offer

APPLE iPhone 13 mini (Starlight, 128 GB)

4.5 ★ 3,448 Ratings & 298 Reviews

- 128 GB ROM
- 15.49 cm (6.1 inch) Super Retina XDR Display
- 12MP + 12MP + 12MP | 12MP Front Camera
- A15 Bionic Chip Processor
- Brand Warranty for 1 Year

₹61,999 Assured

₹64,900 4% off

Free delivery by Today

Upto ₹35,600 Off on Exchange

Bank Offer

APPLE iPhone 13 mini (Pink, 128 GB)

4.5 ★ 3,448 Ratings & 298 Reviews

- 128 GB ROM
- 13.72 cm (5.4 inch) Super Retina XDR Display

₹61,999

₹64,900 4% off

Free delivery by Today

Upto ₹35,600 Off on Exchange

Flipkart Explore Plus+ **iphone 13 128** Search Login Become a Seller More Cart



Currently unavailable

Add to Compare

4.5 ★ 3,749 Ratings & 298 Reviews Back to top

- 128 GB ROM
- 13.72 cm (5.4 inch) Super Retina XDR Display
- 12MP + 12MP | 12MP Front Camera
- A15 Bionic Chip Processor
- Brand Warranty for 1 Year

₹64,900 4% off
Free delivery by Today
Upto ₹38,600 Off on Exchange
No Cost EMI from ₹10,334/month



Currently unavailable

Add to Compare

4.5 ★ 3,448 Ratings & 298 Reviews Back to top

APPLE iPhone 13 mini (Pink, 128 GB)

- 128 GB ROM
- 13.72 cm (5.4 inch) Super Retina XDR Display
- 12MP + 12MP | 12MP Front Camera
- A15 Bionic Chip Processor
- Brand Warranty for 1 Year

₹61,999
₹64,900 4% off
Free delivery by Today
Upto ₹38,600 Off on Exchange
No Cost EMI from ₹10,334/month



Currently unavailable

Add to Compare

4.5 ★ 3,749 Ratings & 298 Reviews Back to top

APPLE iPhone 13 Pro Max (Silver, 128 GB)

- 128 GB ROM
- 17.02 cm (6.7 inch) Super Retina XDR Display

₹1,29,900 Assured
Free delivery by Today
Upto ₹35,600 Off on Exchange
Bank Offer



Currently unavailable

Add to Compare

4.5 ★ 3,749 Ratings & 298 Reviews Back to top

APPLE iPhone 13 Pro Max (Silver, 128 GB)

- 128 GB ROM
- 17.02 cm (6.7 inch) Super Retina XDR Display
- 12MP + 12MP + 12MP | 12MP Front Camera
- A15 Bionic Chip Processor
- Brand Warranty for 1 Year

₹1,29,900 Assured
Free delivery by Today
Upto ₹35,600 Off on Exchange
Bank Offer



Currently unavailable

Add to Compare

4.5 ★ 3,749 Ratings & 298 Reviews Back to top

APPLE iPhone 13 Pro Max (Alpine Green, 128 GB)

- 128 GB ROM
- 17.02 cm (6.7 inch) Super Retina XDR Display

₹1,29,900 Assured
Free delivery by Today
Upto ₹35,600 Off on Exchange
Bank Offer



Currently unavailable

Add to Compare

4.5 ★ 3,749 Ratings & 298 Reviews Back to top

APPLE iPhone 13 128

- 128 GB ROM
- 17.02 cm (6.7 inch) Super Retina XDR Display
- 12MP + 12MP + 12MP | 12MP Front Camera
- A15 Bionic Chip Processor
- Brand Warranty for 1 Year

₹64,900 4% off
Free delivery by Today
Upto ₹38,600 Off on Exchange
Bank Offer



Currently unavailable

Add to Compare

4.5 ★ 3,749 Ratings & 298 Reviews Back to top

APPLE iPhone 13 Pro Max (Alpine Green, 128 GB)

- 128 GB ROM
- 17.02 cm (6.7 inch) Super Retina XDR Display
- 12MP + 12MP + 12MP | 12MP Front Camera
- A15 Bionic Chip Processor
- Brand Warranty for 1 Year

₹1,29,900 Assured
Free delivery by Today
Upto ₹35,600 Off on Exchange
Bank Offer



Currently unavailable

Add to Compare

4.5 ★ 3,749 Ratings & 298 Reviews Back to top

APPLE iPhone 13 Pro Max (Sierra Blue, 128 GB)

- 128 GB ROM
- 17.02 cm (6.7 inch) Super Retina XDR Display

₹1,29,900 Assured
Free delivery by Today
Upto ₹35,600 Off on Exchange
Bank Offer

Flipkart Explore Plus

iphone 13 128

4.0 ★ 1,714 Ratings & 103 Reviews

Currently unavailable

Add to Compare

Back to top

128 GB ROM
17.02 cm (6.7 inch) Super Retina XDR Display
12MP + 12MP + 12MP | 12MP Front Camera
A15 Bionic Chip Processor
Brand Warranty for 1 Year

Free delivery by Today
Upto ₹35,600 Off on Exchange
Bank Offer

APPLE iPhone 13 Pro Max (Sierra Blue, 128 GB)

4.6 ★ 1,914 Ratings & 183 Reviews

Currently unavailable

Add to Compare

128 GB ROM
17.02 cm (6.7 inch) Super Retina XDR Display
12MP + 12MP + 12MP | 12MP Front Camera
A15 Bionic Chip Processor
Brand Warranty for 1 Year

₹1,29,900 Assured

Free delivery by Today
Upto ₹35,600 Off on Exchange
Bank Offer

APPLE iPhone 13 Pro (Silver, 128 GB)

4.7 ★ 1,471 Ratings & 121 Reviews

Currently unavailable

Add to Compare

128 GB ROM
15.49 cm (6.1 inch) Super Retina XDR Display

₹1,19,900 Assured

Free delivery by Today
Upto ₹35,600 Off on Exchange
Bank Offer

Flipkart Explore Plus

iphone 13 128

4.0 ★ 1,714 Ratings & 103 Reviews

Currently unavailable

Add to Compare

128 GB ROM
17.02 cm (6.7 inch) Super Retina XDR Display
12MP + 12MP + 12MP | 12MP Front Camera
A15 Bionic Chip Processor
Brand Warranty for 1 Year

Free delivery by Today
Upto ₹35,600 Off on Exchange
Bank Offer

APPLE iPhone 13 Pro (Silver, 128 GB)

4.7 ★ 1,471 Ratings & 121 Reviews

Currently unavailable

Add to Compare

128 GB ROM
15.49 cm (6.1 inch) Super Retina XDR Display
12MP + 12MP + 12MP | 12MP Front Camera
A15 Bionic Chip Processor
Brand Warranty for 1 Year

₹1,19,900 Assured

Free delivery by Today
Upto ₹35,600 Off on Exchange
Bank Offer

APPLE iPhone 13 Pro (Alpine Green, 128 GB)

4.7 ★ 1,471 Ratings & 121 Reviews

Currently unavailable

Add to Compare

128 GB ROM
15.49 cm (6.1 inch) Super Retina XDR Display

₹1,19,900 Assured

Free delivery by Today
Upto ₹35,600 Off on Exchange
Bank Offer

Flipkart Explore Plus

iphone 13 128

4.0 ★ 1,714 Ratings & 121 Reviews

Currently unavailable

Add to Compare

128 GB ROM
15.49 cm (6.1 inch) Super Retina XDR Display
12MP + 12MP + 12MP | 12MP Front Camera
A15 Bionic Chip Processor
Brand Warranty for 1 Year

Free delivery by Today
Upto ₹35,600 Off on Exchange
Bank Offer

APPLE iPhone 13 Pro (Alpine Green, 128 GB)

4.7 ★ 1,471 Ratings & 121 Reviews

Currently unavailable

Add to Compare

128 GB ROM
15.49 cm (6.1 inch) Super Retina XDR Display
12MP + 12MP + 12MP | 12MP Front Camera
A15 Bionic Chip Processor
Brand Warranty for 1 Year

₹1,19,900 Assured

Free delivery by Today
Upto ₹35,600 Off on Exchange
Bank Offer

APPLE iPhone 13 Pro (Sierra Blue, 128 GB)

4.7 ★ 1,471 Ratings & 121 Reviews

Currently unavailable

Add to Compare

128 GB ROM
15.49 cm (6.1 inch) Super Retina XDR Display

₹1,19,900 Assured

Free delivery by Today
Upto ₹35,600 Off on Exchange
Bank Offer

Flipkart Explore Plus Q

Login Become a Seller More Cart



4.7 / 5 1,471 Ratings & 121 Reviews ^ Back to top

- 128 GB ROM
- 15.49 cm (6.1 inch) Super Retina XDR Display
- 12MP + 12MP + 12MP | 12MP Front Camera
- A15 Bionic Chip Processor
- Brand Warranty for 1 Year

Currently unavailable

Add to Compare



4.7 / 5 1,471 Ratings & 121 Reviews

APPLE iPhone 13 Pro (Sierra Blue, 128 GB)

₹1,19,900 Assured

Free delivery by Today
Upto ₹35,600 Off on Exchange
Bank Offer

- 128 GB ROM
- 15.49 cm (6.1 inch) Super Retina XDR Display
- 12MP + 12MP + 12MP | 12MP Front Camera
- A15 Bionic Chip Processor
- Brand Warranty for 1 Year

Currently unavailable

Add to Compare



4.6 / 5 43,516 Ratings & 1,670 Reviews

APPLE iPhone 14 (Midnight, 128 GB)

₹68,999 Assured

₹79,900 13% off
Free delivery by Today
Upto ₹38,600 Off on Exchange

- 128 GB ROM
- 15.49 cm (6.1 inch) Super Retina XDR Display

Flipkart Explore Plus Q

Login Become a Seller More Cart



4.7 / 5 1,471 Ratings & 121 Reviews ^ Back to top

- 128 GB ROM
- 15.49 cm (6.1 inch) Super Retina XDR Display
- 12MP + 12MP + 12MP | 12MP Front Camera
- A15 Bionic Chip Processor
- Brand Warranty for 1 Year

Currently unavailable

Add to Compare



4.6 / 5 43,516 Ratings & 1,670 Reviews

APPLE iPhone 14 (Midnight, 128 GB)

₹68,999 Assured

₹79,900 13% off
Free delivery by Today
Upto ₹38,600 Off on Exchange
Bank Offer

- 128 GB ROM
- 15.49 cm (6.1 inch) Super Retina XDR Display
- 12MP + 12MP + 12MP | 12MP Front Camera
- A15 Bionic Chip, 6 Core Processor Processor
- 1 Year Warranty for Phone and 6 Months Warranty for In-Box Accessories

Add to Compare



4.6 / 5 43,516 Ratings & 1,670 Reviews

APPLE iPhone 14 (Purple, 128 GB)

₹68,999 Assured

₹79,900 13% off
Free delivery by Today
Upto ₹38,600 Off on Exchange

- 128 GB ROM
- 15.49 cm (6.1 inch) Super Retina XDR Display

Flipkart Explore Plus+ Q Login Become a Seller More Cart



4.6 ★ 43,516 Ratings & 1,670 reviews ^ Back to top

- 128 GB ROM
- 15.49 cm (6.1 inch) Super Retina XDR Display
- 12MP + 12MP | 12MP Front Camera
- A15 Bionic Chip, 6 Core Processor Processor
- 1 Year Warranty for Phone and 6 Months Warranty for In-Box Accessories

₹79,900 13% off
Free delivery by Today
Upto ₹38,600 Off on Exchange
Bank Offer

Add to Compare



APPLE iPhone 14 (Purple, 128 GB) 4.6 ★ 43,516 Ratings & 1,670 Reviews

- 128 GB ROM
- 15.49 cm (6.1 inch) Super Retina XDR Display
- 12MP + 12MP | 12MP Front Camera
- A15 Bionic Chip, 6 Core Processor Processor
- 1 Year Warranty for Phone and 6 Months Warranty for In-Box Accessories

₹68,999 Assured
₹79,900 13% off
Free delivery by Today
Upto ₹38,600 Off on Exchange
Bank Offer

Add to Compare

Page 1 of 5

1 2 3 4 5 NEXT

Did you find what you were looking for?

 Yes No

Flipkart Explore Plus+ Q Login Become a Seller More Cart



4.6 ★ 43,516 Ratings & 1,670 reviews ^ Back to top

- 128 GB ROM
- 15.49 cm (6.1 inch) Super Retina XDR Display
- 12MP + 12MP | 12MP Front Camera
- A15 Bionic Chip, 6 Core Processor Processor
- 1 Year Warranty for Phone and 6 Months Warranty for In-Box Accessories

₹79,900 13% off
Free delivery by Today
Upto ₹38,600 Off on Exchange
Bank Offer

Add to Compare

Page 1 of 5

1 2 3 4 5 NEXT

Did you find what you were looking for?

Yes No

Reviews for Popular Mobiles



1. APPLE iPhone 14 (Midnight, ...)

4.6 ★ 43,516 Ratings & 1,670 Reviews

₹68,999 13% off

128 GB ROM

15.49 cm (6.1 inch) Super Retina XDR Display

12MP + 12MP | 12MP Front Camera

Most Helpful Review

5★ Great product

Nice camera but battery drain fast specially on video recording

Tara singh mehra 4 months ago

Recent Review

5★ Worth every penny

1st iPhone, not regretting the decision of switching from android. Overall is good, Completely satisfied.

Mohammad Salman Certified Buyer 2 days ago