1. **Write Selenium script to demonstrate Selenium WebDriver basic commands**

package batch1;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.chrome.ChromeDriver;

public class program1 {

public static void main(String[] args) throws InterruptedException {

// TODO Auto-generated method stub

WebDriver driver=new ChromeDriver();

driver.get("https://www.google.com/");

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

String title =driver.getTitle();

System.out.println(title);

Thread.sleep(1000);

WebElement t=driver.findElement(By.tagName("textarea"));

t.sendKeys("youtube");

t.submit();

driver.findElement(By.xpath("//a[@href='https://www.youtube.com/']")).click();

driver.navigate().refresh();

String url=driver.getCurrentUrl();

System.out.println(url);

driver.manage().window().fullscreen();

driver.navigate().back();

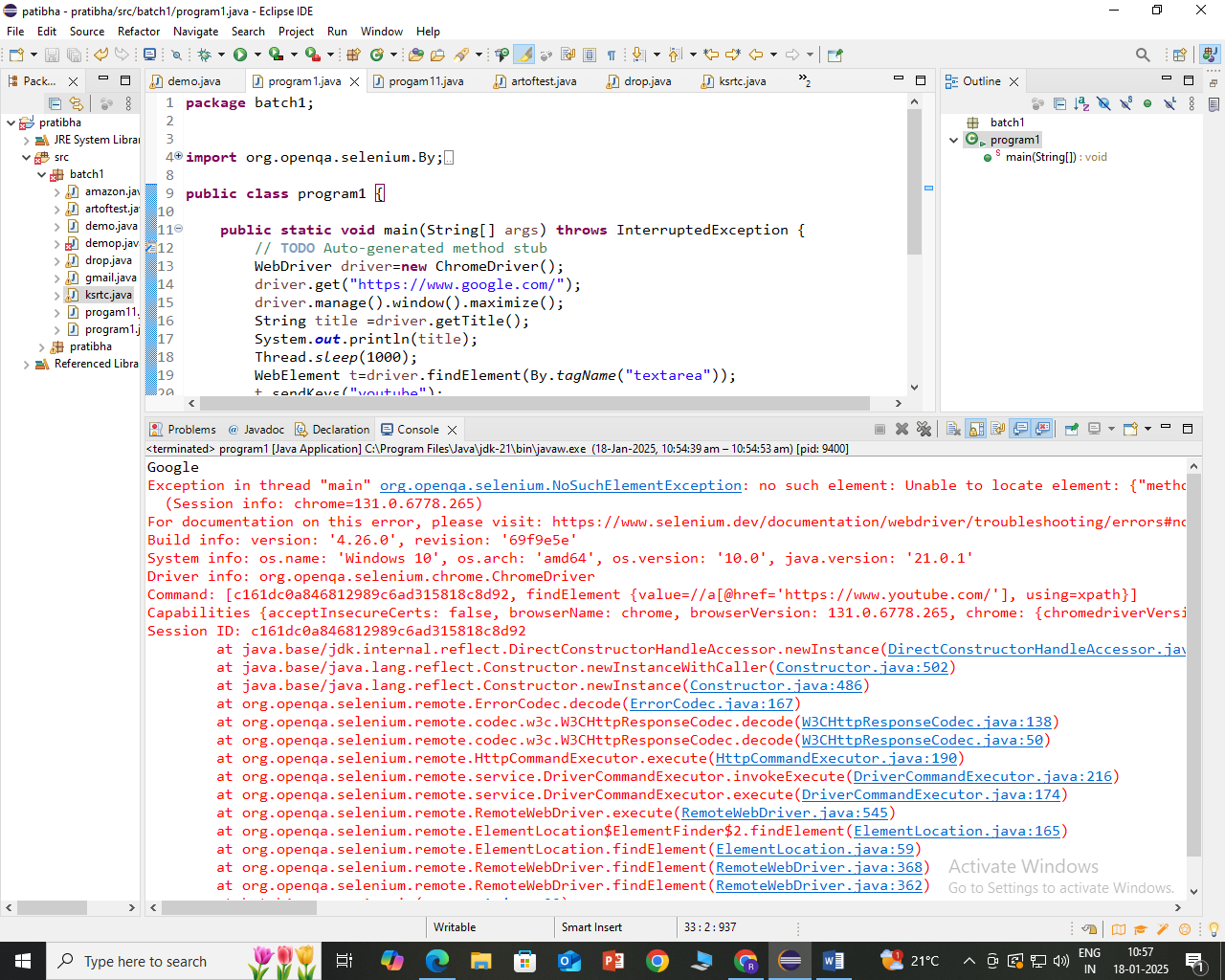
Thread.sleep(2000);

driver.close();

}

}

Output:



1. Write Selenium script to demonstrate Selenium WebDriver locators

package batch1;

import java.util.concurrent.TimeUnit;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.chrome.ChromeDriver;

public class artoftest {

public static void main(String[] args)throws InterruptedException {

// TODO Auto-generated method stub

WebDriver driver=new ChromeDriver();

driver.get("https://artoftesting.com/samplesiteforselenium");

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

driver.manage().timeouts().implicitlyWait(2,TimeUnit.SECONDS);

String sampleText=driver.findElement(By.id("fname")).getText();

System.out.println(sampleText);

Thread.sleep(3000);

driver.findElement(By.linkText("This is a link")).click();

driver.navigate().back();

driver.findElement(By.name("firstName")).sendKeys("Hello");

driver.findElement(By.name("firstName")).clear();

driver.findElement(By.xpath("//button[@id='idOfButton']")).click();

Thread.sleep(3000);

driver.findElement(By.xpath("//button[@id='dblClkBtn']")).click();

Thread.sleep(3000);

driver.findElement(By.cssSelector("#female")).click();

driver.findElement(By.cssSelector("input[value='Performance']")).click();

Thread.sleep(3000);

WebElement image=driver.findElement(By.tagName("img"));

System.out.println(image.getAttribute("alt"));

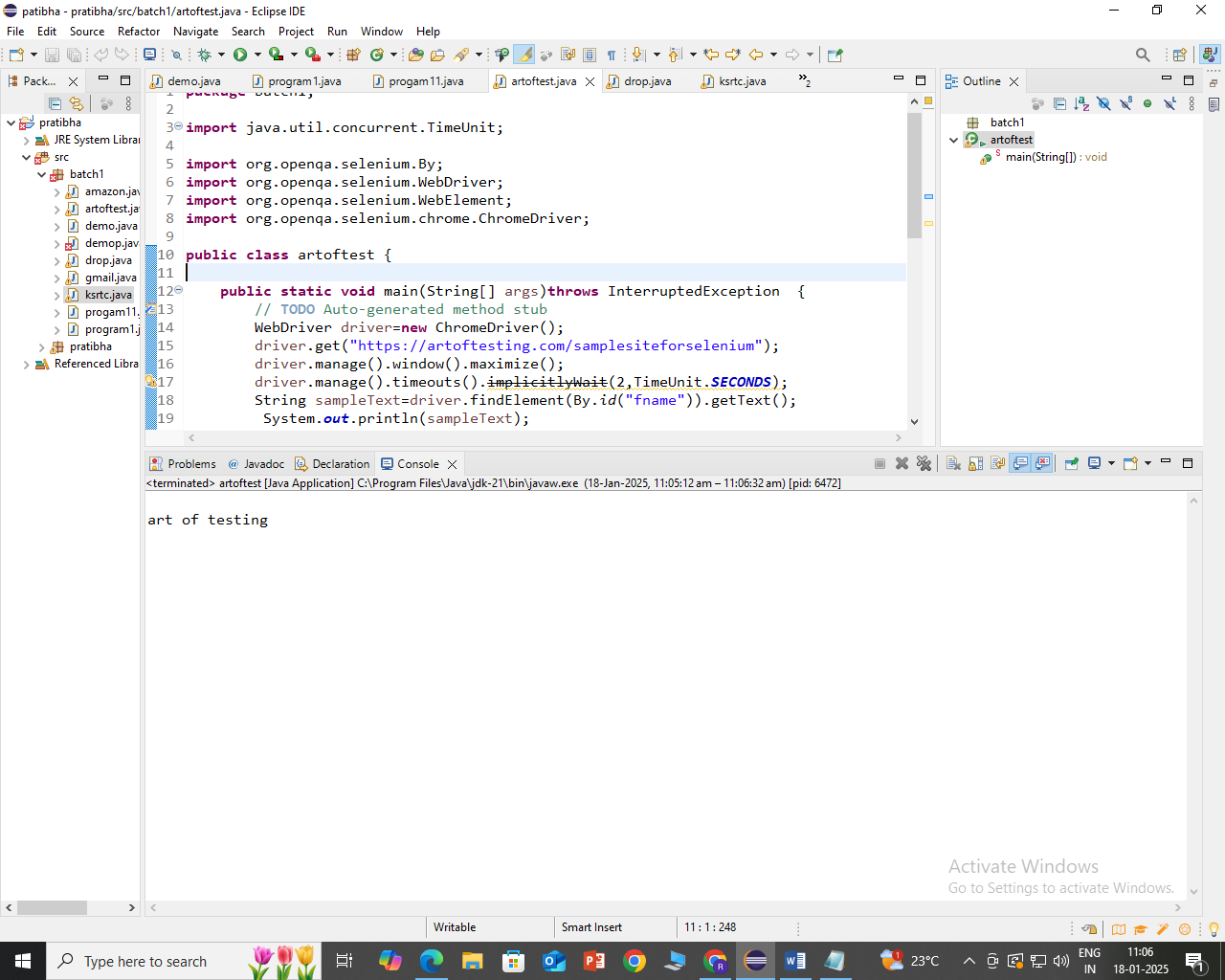
driver.findElement(By.partialLinkText("Tutorial")).click();

driver.quit();

}

}

Output:



1. Write a script to demonstrate Gmail login using Selenium WebDriver.

package batch1;

import java.time.Duration;

import java.util.concurrent.TimeUnit;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.chrome.ChromeDriver;

import org.openqa.selenium.support.ui.ExpectedConditions;

import org.openqa.selenium.support.ui.WebDriverWait;

public class gmail {

public static void main(String[] args) throws InterruptedException {

WebDriver driver=new ChromeDriver();

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

driver.navigate().to("https://gmail.com/");

String expectedTitle="Gmail";

String actualTitle=driver.getTitle();

System.out.println(actualTitle);

driver.manage().timeouts().implicitlyWait(5,TimeUnit.SECONDS);

Thread.sleep(5000);

if(actualTitle.equals(expectedTitle))

{

System.out.println("Title is matching");

}

else {

System.out.println("Title is not matching");

}

driver.findElement(By.id("identifierId")).sendKeys("laxmih2304@gmail.com");

driver.findElement(By.xpath("//span[normalize-space()='Next']")).click();

WebDriverWait wait = new WebDriverWait(driver, Duration.ofSeconds(30));

wait.until(ExpectedConditions.elementToBeClickable(By.xpath("//input[@name='Passwd']")

);

driver.findElement(By.cssSelector("input[name='Passwd']")).sendKeys("Laxmi@123");

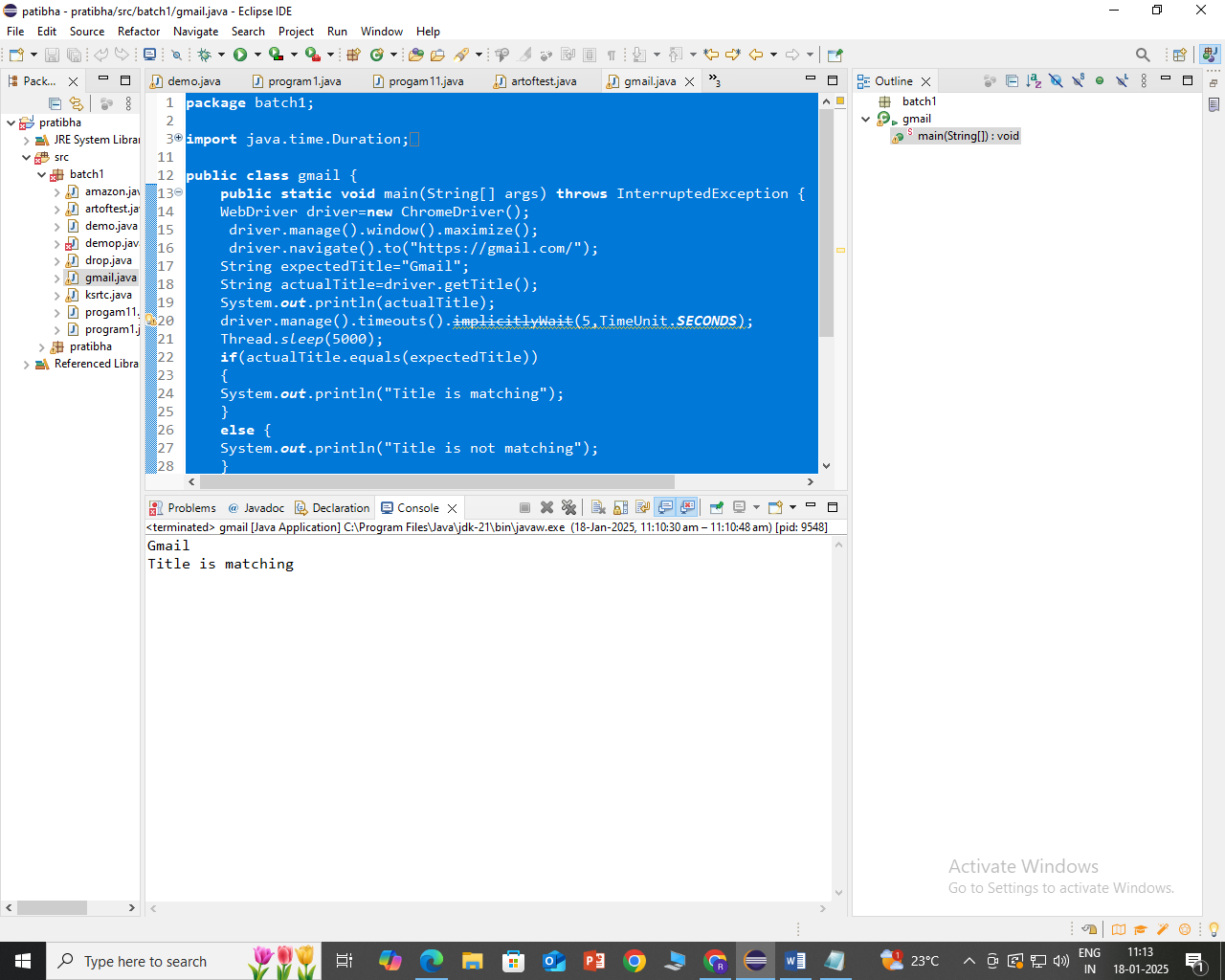
driver.findElement(By.xpath("//span[normalize-space()='Next']")).click();

driver.quit();

}

}

}}  
Output:



1. Write a script to add an item to cart of Amazon using Window handles method using Selenium WebDriver.

package batch1;

import java.util.ArrayList;

import java.util.concurrent.TimeUnit;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.chrome.ChromeDriver;

public class amazon {

public static void main(String[] args)throws InterruptedException {

WebDriver driver=new ChromeDriver();

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

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

driver.manage().timeouts().implicitlyWait(5,TimeUnit.SECONDS);

driver.manage().deleteAllCookies();

WebElement

searchbox=driver.findElement(By.id("twotabsearchtextbox"));

searchbox.sendKeys("iphone 16");

searchbox.submit();

driver.findElement(By.className("a-price-whole")).click();

ArrayList<String> wid = new

ArrayList<String>(driver.getWindowHandles());

driver.switchTo().window(wid.get(1));

driver.findElement(By.xpath("//input[@id='buy-now-button']")).click();

Thread.sleep(3000);

//driver.findElement(By.id("ap\_email")).sendKeys("8792242985");

//Thread.sleep(3000);

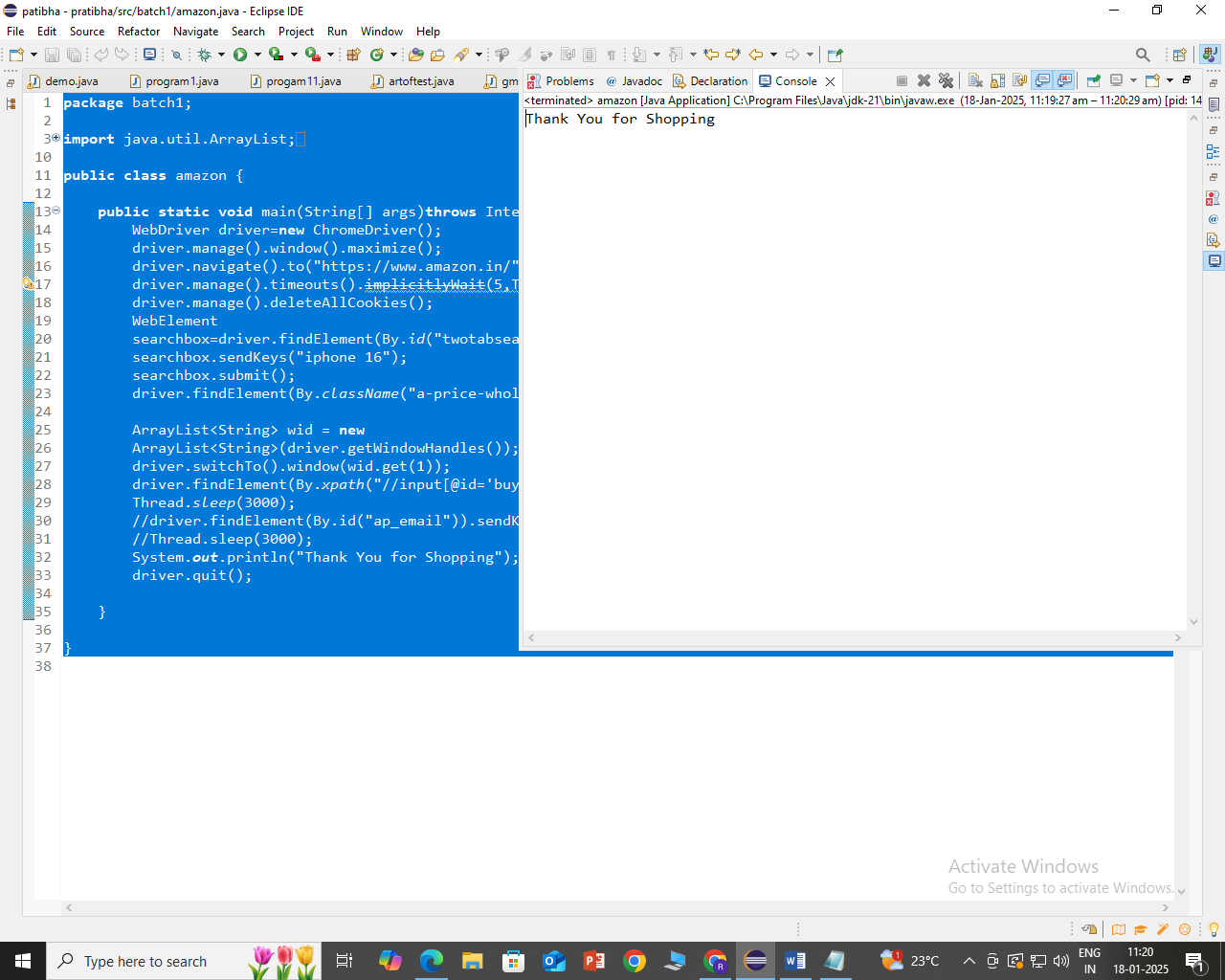
System.out.println("Thank You for Shopping");

driver.quit();

}

}

Output



1. Black Box testing: (Load Testing) Design a web page to get the count of visitors who visit your web page

<?php

// File to store the visitor count

$countFile = 'visitor\_count.txt';

// Function to increment the count and retrieve the updated value

function incrementVisitorCount() {

global $countFile;

$count = 1;

if (file\_exists($countFile)) {

$count = intval(file\_get\_contents($countFile));

$count++;

}

file\_put\_contents($countFile, $count);

return $count;

}

// Get the visitor count

$visitorCount = incrementVisitorCount();

?>

<!DOCTYPE html>

<html>

<head>

<title>Visitor Count Page</title>

<style>

body {

font-family: Arial, sans-serif;

text-align: center;

}

h1 {

margin-top: 50px;

}

#counter {

font-size: 48px;

margin-top: 20px;

}

</style>

</head>

<body>

<h1>Welcome to the Visitor Count Page</h1>

<p>Number of Visitors:</p>

<div id="counter"><?php echo $visitorCount; ?></div>

</body>

</html>

1. Black Box testing: (Functional Testing) Write a selenium script to test total number of objects present on the web page

package batch1;

import java.util.List;

import java.util.concurrent.TimeUnit;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.chrome.ChromeDriver;

public class drop {

public static void main(String[] args)throws InterruptedException {

WebDriver driver = new ChromeDriver();

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

driver.get("https://webdriveruniversity.com/Dropdown-Checkboxes-RadioButtons/index.html");

driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS);

// Get count of CheckBoxes present

List<WebElement> checkboxes =

driver.findElements(By.xpath("//input[@type='checkbox']"));

System.out.println(checkboxes.size() + " Number of CheckBoxes");

// Get count of Dropdown menus

List<WebElement> dropdown = driver.findElements(By.tagName("select"));

System.out.println(dropdown.size() + " Number of DropDown Menus");

// Get count no’ of radio buttons

List<WebElement> radioBtns =

driver.findElements(By.xpath("//input[@type='radio']"));

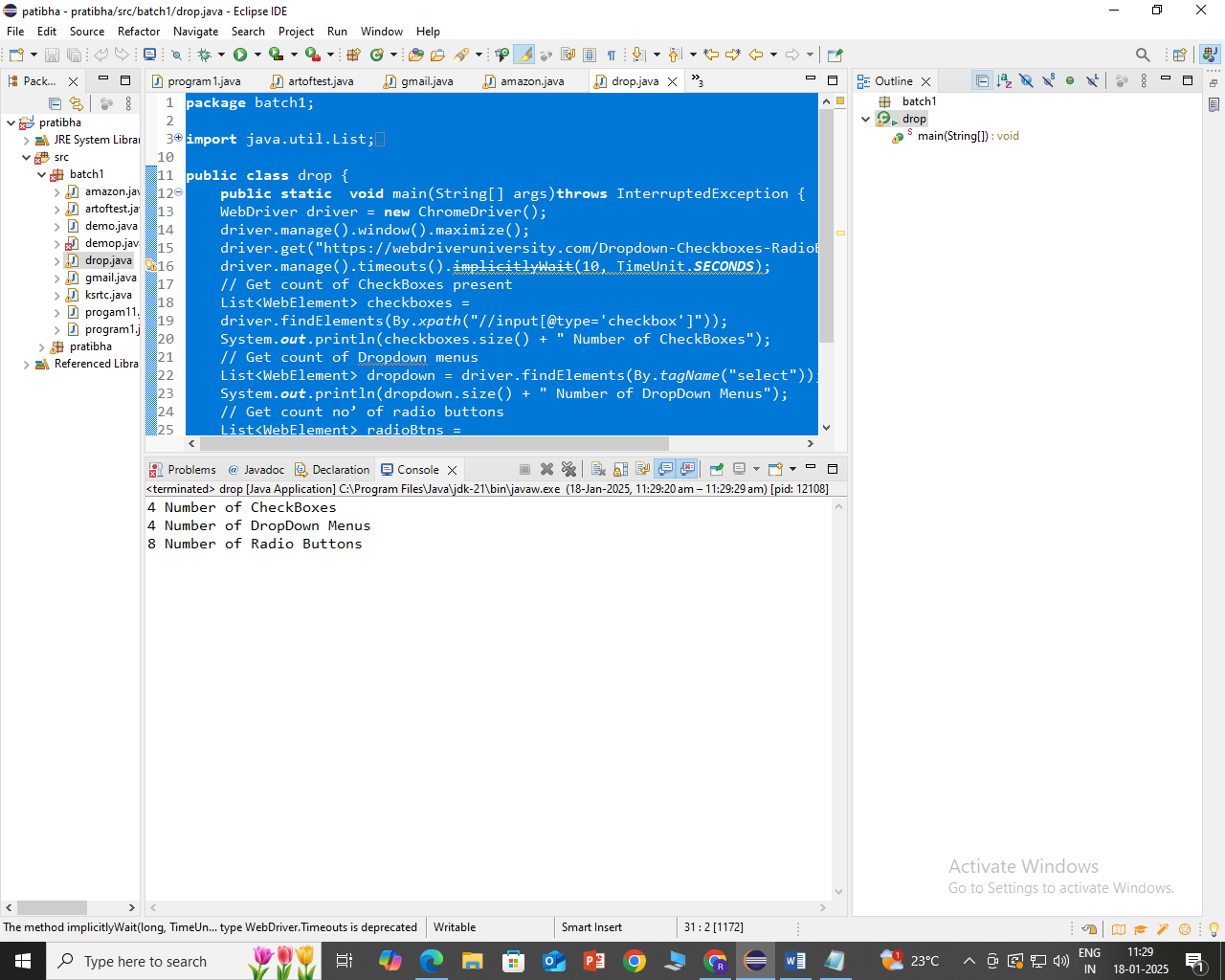
System.out.println(radioBtns.size() + " Number of Radio Buttons");

driver.close();

}

}

Output:



1. **Black Box testing: (Functional Testing) Write and test a program to get the number of list items in a list / combo box**

Drop.html

<html>

<title>Write and test a program to get the number of list items in a list / combo box.

</title>

<body>

<label >Select a Programming Language:</label>

<select name="language" id="language">

<option value="javascript">JavaScript</option>

<option value="python">Python</option>

<option value="c++" disabled>C++</option>

<option value="java" selected>Java</option>

</select>

</body>

</html>

package SeleniumWebDriver;

import java.util.List;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.chrome.ChromeDriver;

import org.openqa.selenium.support.ui.Select;

public class drop {

public static void main(String[] args) {

System.setProperty("webdriver.chrome.driver",

" F:\\ Selenium\\chromedriver-win32\\chromedriver.exe ");

WebDriver driver = new ChromeDriver();

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

// URL launch

driver.get("F:\\drop.html");

// Identify list

Select items = new Select(driver.findElement(By.name("language")));

List<WebElement> mylist = items.getOptions();

System.out.println("Number of items = " + mylist.size());

driver.quit();

}

}

OUTPUT:

Number of items = 4

1. **Demonstrate how to handle Window Pop Up in Selenium WebDriver**

package batch1;

import org.openqa.selenium.Alert;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.chrome.ChromeDriver;

public class ksrtc {

public static void main(String[] args) throws InterruptedException {

// TODO Auto-generated method stub

WebDriver driver = new ChromeDriver();

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

driver.get("https://ksrtc.in/");

driver.findElement(By.id("submitSearch")).click();

Thread.sleep(2000);

Alert simpleAlert = driver.switchTo().alert();

String alertMessage = driver.switchTo().alert().getText();

System.out.println(alertMessage);

simpleAlert.accept();

Thread.sleep(5000);

driver.close();

}

}

Output:

Please select start place.

1. **Write and test a program to count number of Check boxes on the page checked and unchecked**

Checkbox.html

<html>

<body>

<h2>Choose your Hobbies</h2>

<input type="checkbox" name="hobby" value="Reading"> Reading<br>

<input type="checkbox" name="hobby" value="Swiming"> Swiming<br>

<input type="checkbox" name="hobby" value="Travelling"> Travelling<br>

<input type="checkbox" name="hobby" value="Hiking"> Hiking<br>

<input type="checkbox" name="hobby" value="Dancing"> Dancing<br>

<br><br>

</body>

</html>

package pratibha;

import java.util.List;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.chrome.ChromeDriver;

public class checkbox {

public static void main(String[] args) throws InterruptedException {

WebDriver driver = new ChromeDriver();

driver.get("C:\\Users\\Admin\\Desktop\\checkbox.html");

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

Thread.sleep(3000);

driver.findElement(By.xpath("/html/body/input[1]")).click();

driver.findElement(By.xpath("/html/body/input[3]")).click();

List <WebElement> checkBox =

driver.findElements(By.xpath("//input[@type='checkbox']"));

int checkedCount = 0;

int uncheckedCount = 0;

for (int i=0;i<checkBox.size();i++)

{

if ( checkBox.get(i).isSelected() == true)

checkedCount++;

else

uncheckedCount++;

}

System.out.println("Number of checked checkboxes are " +checkedCount);

System.out.println("Number of unchecked checkboxes are " + uncheckedCount);

driver.close();

}

}

Output

Number of checked checkboxes are 2

Number of unchecked checkboxes are 3

**13. Write and test a program to update 10 student records into table into Excel file**

**package** testNGproject;

**import** java.io.FileInputStream;

**import** java.io.FileOutputStream;

**import** jxl.Sheet;

**import** jxl.Workbook;

**import** jxl.write.Label;

**import** jxl.write.WritableSheet;

**import** jxl.write.WritableWorkbook;

**import** org.testng.annotations.\*;

**public** **class** studentRecords {

@Test

**public** **void** testImportExport1() **throws** Exception {

// Specify input and output file paths

String inputFile = "E:\\student.xlt";

String outputFile = "E:\\ Result.xlt";

// Read the Excel file

FileInputStream fi = **new** FileInputStream(inputFile);

Workbook w = Workbook.*getWorkbook*(fi);

Sheet s = w.getSheet(0);

// Initialize 2D array to store data from the sheet

String[][] a = **new** String[s.getRows()][s.getColumns()];

// Create the output Excel file and writable sheet

FileOutputStream fo = **new** FileOutputStream(outputFile);

WritableWorkbook wwb = Workbook.*createWorkbook*(fo);

WritableSheet ws = wwb.createSheet("result1", 0);

// Write headers to the result sheet

**for** (**int** i = 0; i < s.getRows(); i++) {

**for** (**int** j = 0; j < s.getColumns(); j++) {

a[i][j] = s.getCell(j, i).getContents(); // Read contents

Label l2 = **new** Label(j, i, a[i][j]); // Write the data

ws.addCell(l2);

}

}

Label resultLabel = **new** Label(s.getColumns(), 0, "Result");

ws.addCell(resultLabel);

// Process the data and calculate the result for each student

**for** (**int** i = 1; i < s.getRows(); i++) {

**for** (**int** j = 2; j < s.getColumns(); j++) {

a[i][j] = s.getCell(j, i).getContents();

**int** score = Integer.*parseInt*(a[i][j]);

String result = (score > 35) ? "pass" : "fail";

// Write the result to the "Result" column

Label resultCell = **new** Label(s.getColumns(), i, result);

ws.addCell(resultCell);

}

}

// Write and close the output file

wwb.write();

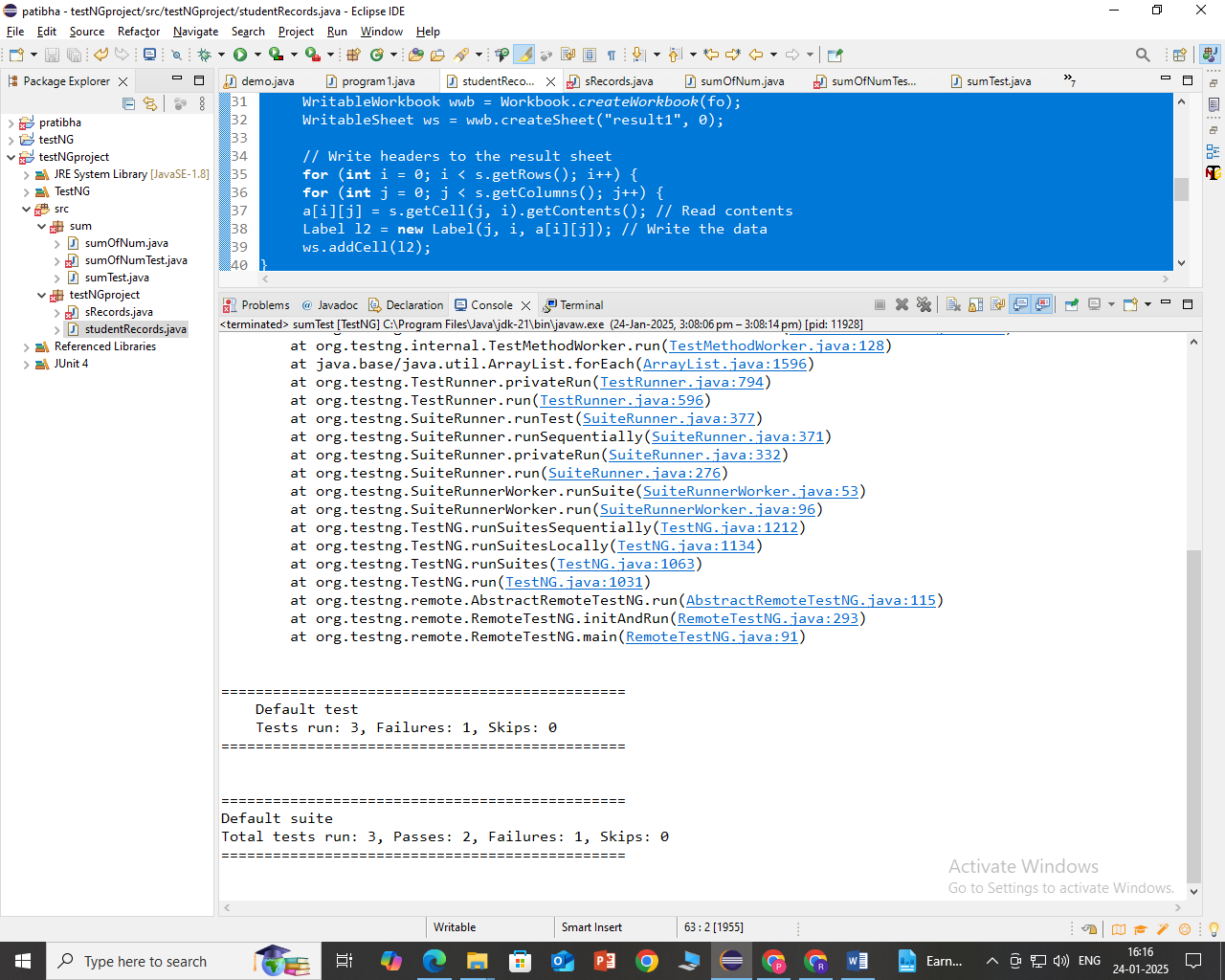
wwb.close();

System.***out***.println("Records successfully updated and saved to Result.xls");

}

}

Output:



14.

**Write a program in java to test do…while construct using TestNG**

**Java File:**

**package** sum;

**public** **class** sumOfNum {

**public** **static** **void** main(String[] args) {

**int** i = 1, sum = 0;**do**

{

sum += i; // sum=sum+i;

i++;

}

**while**(i<=10);

System.***out***.println("The sum of numbers from 1 to 10 is: "+sum);

}

**public** **static** **int** calculateSum(**int** start, **int** end) {

**int** i = start, sum = 0;

**do** {

sum += i;

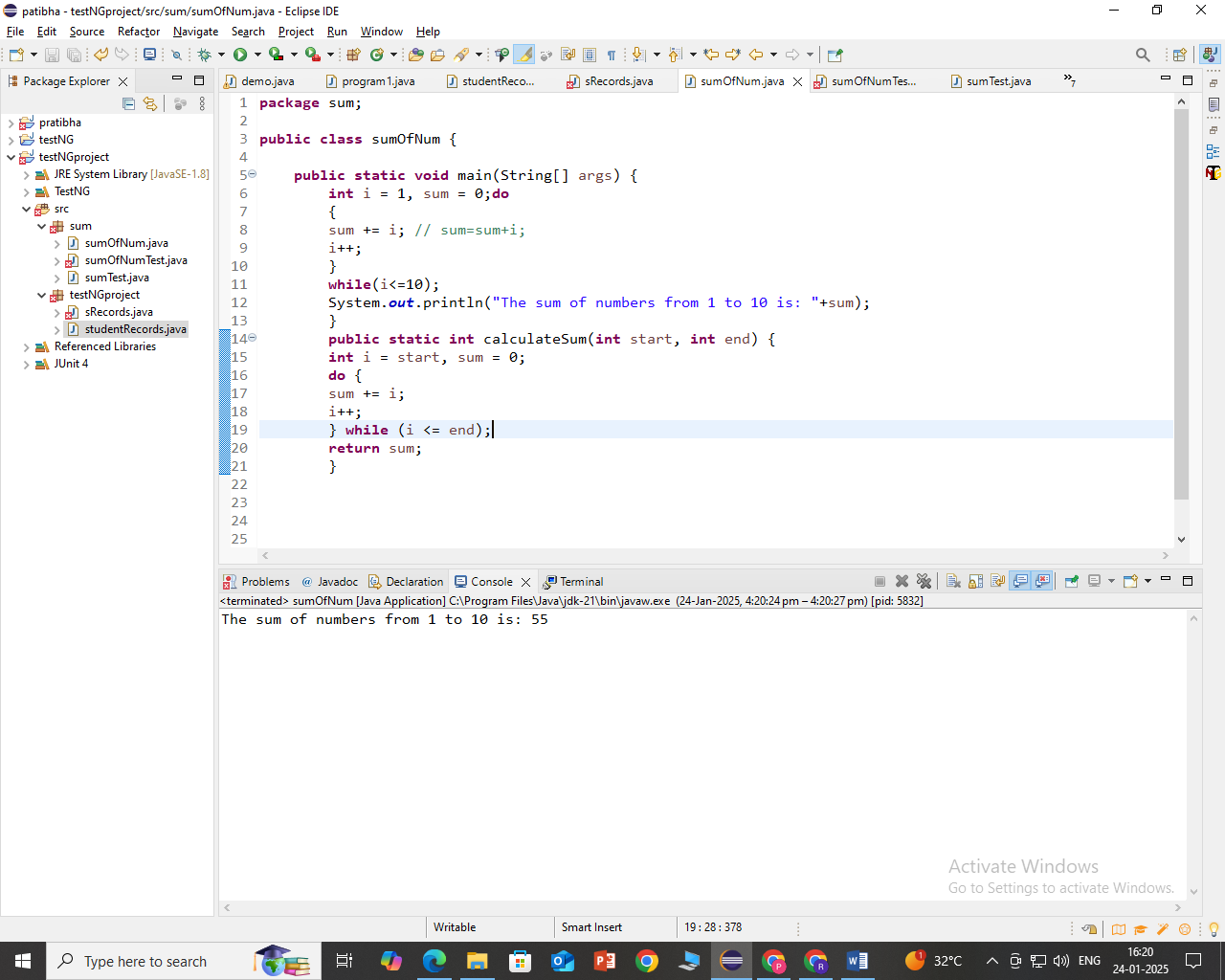
i++;

} **while** (i <= end);

**return** sum;

}

}



**TestNG File:**

**package** sum;

**import** org.junit.Assert;

**import** org.testng.annotations.Test;

**public** **class** sumTest {

@Test

**public** **void** testPositiveValuesInRange() {

**int** expectedSum = 55;

**int** actualSum=sumOfNum.*calculateSum*(1,10);

Assert.*assertEquals*(actualSum,expectedSum);

}

@Test

**public** **void** testNegativeValuesInRange() {

**int** expectedSum =0;

**int** actualSum=sumOfNum.*calculateSum*(-10,10);

Assert.*assertEquals*(actualSum,expectedSum);

}

@Test

**public** **void** testOutOfRange() {

//int expectedSum=155

**int** expectedSum = 0;

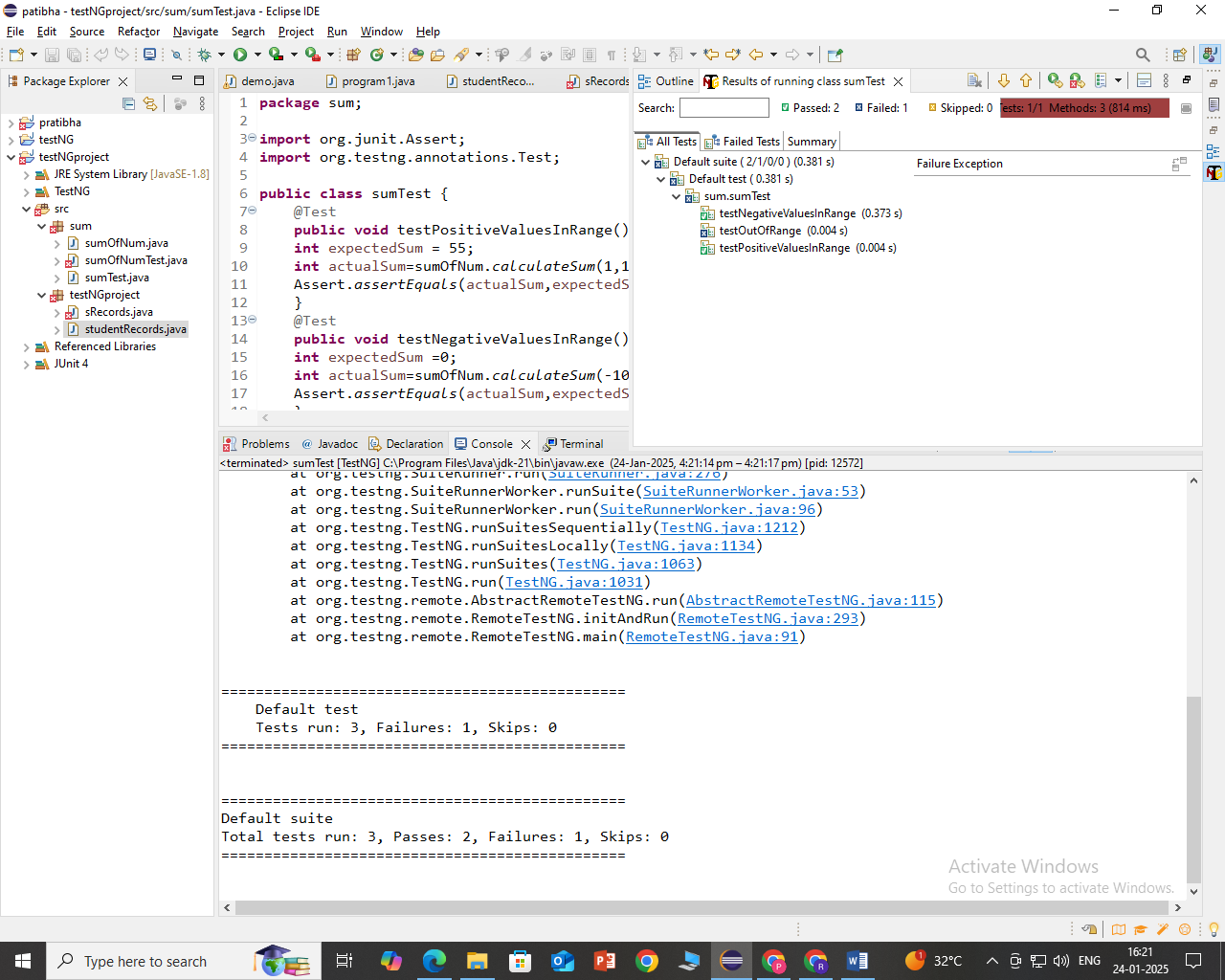
**int** actualSum =sumOfNum.*calculateSum*(11,20);

Assert.*assertEquals*(actualSum,expectedSum);

}

}

**Output of TestNG:**



**15. White Box Testing: Write a program in Java to compute the factorial of a given non-negative number using Iterative Process and Recursion and test using TestNG**

**Java class**

**package** fact1;

**public** **class** fact {

**public** **static** **void** main(String[] args) {

// **TODO** Auto-generated method stub

}

**public** **static** **int** factorialIterative(**int** n) {

**if** (n < 0) {

**throw** **new** IllegalArgumentException("n must be non-negative");

}

**int** result = 1;

**for** (**int** i = 1; i <= n; i++) {

result \*= i;

}

**return** result;

}

**public** **static** **int** factorialRecursive(**int** n) {

**if** (n < 0) {

**throw** **new** IllegalArgumentException("n must be non-negative");

}

**if** (n == 0) {

**return** 1;

} **else** {

**return** n \* *factorialRecursive*(n - 1);

}

}

}

**package** fact1;

**import** org.testng.Assert;

**import** org.testng.annotations.Test;

**public** **class** facttestng {

@Test

**public** **void** testFactorialIterativeZero() {

**int** result = fact.*factorialIterative*(0);

Assert.*assertEquals*(result, 1);

}

@Test

**public** **void** testFactorialIterativePositive() {

**int** result = fact.*factorialIterative*(5);

Assert.*assertEquals*(result, 120);

}

@Test

**public** **void** testFactorialRecursiveZero() {

**int** result = fact.*factorialRecursive*(0);

Assert.*assertEquals*(result, 1);

}

@Test

**public** **void** testFactorialRecursivePositive() {

**int** result = fact.*factorialRecursive*(4);

Assert.*assertEquals*(result, 240);

}

}

[RemoteTestNG] detected TestNG version 7.4.0

PASSED: testFactorialIterativePositive

PASSED: testFactorialRecursiveZero

PASSED: testFactorialIterativeZero

FAILED: testFactorialRecursivePositive

==============================================

Default test

Tests run: 4, Failures: 1, Skips: 0

===============================================

===============================================

Default suite

Total tests run: 4, Passes: 3, Failures: 1, Skips: 0

===============================================