**Practical 1**

**Install Selenium IDE and create a test suite containing a minimum of 4 test cases for different web page formats (e.g., HTML, XML, JSON, etc.).**

**Step 1:**

**Chrome:**

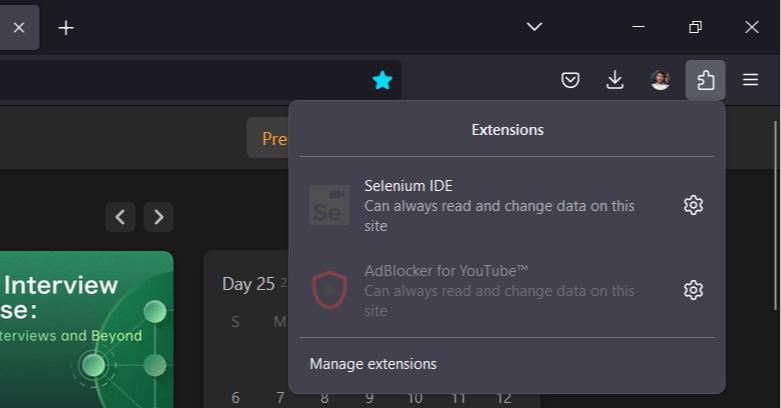
[**https://chrome.google.com/webstore/detail/selenium-ide/mooikfkahbdckldjjn**](https://chrome.google.com/webstore/detail/selenium-ide/mooikfkahbdckldjjndioackbalphokd)

[**dioackbalphokd**](https://chrome.google.com/webstore/detail/selenium-ide/mooikfkahbdckldjjndioackbalphokd)

**Firefox:**

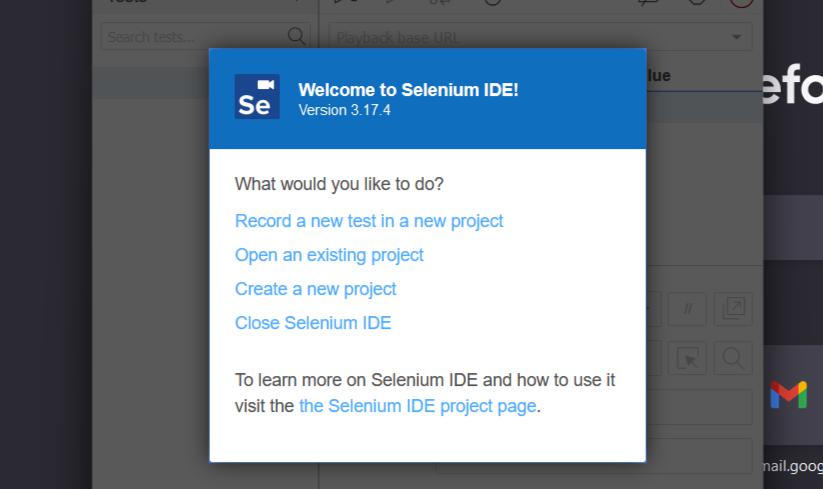
[**https://addons.mozilla.org/en-US/frefox/addon/selenium-ide/**](https://addons.mozilla.org/en-US/firefox/addon/selenium-ide/)

* **Download the Selenium IDE extension**
* **Open by clicking at extensions menu**

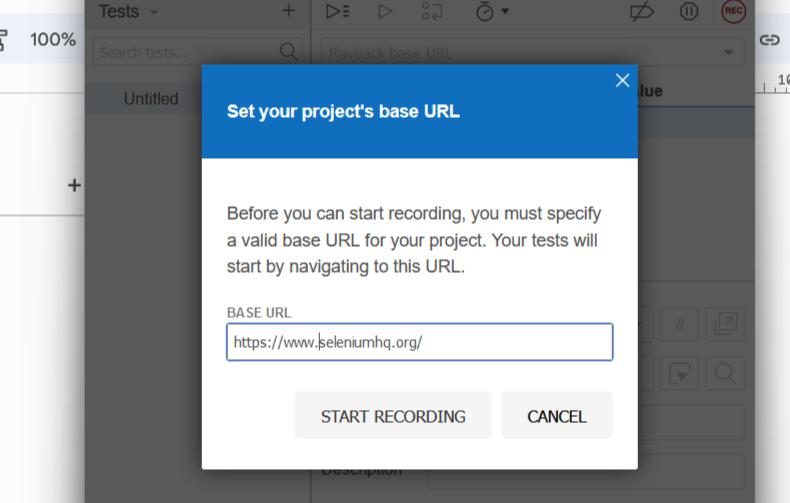
****

**Step 2:**

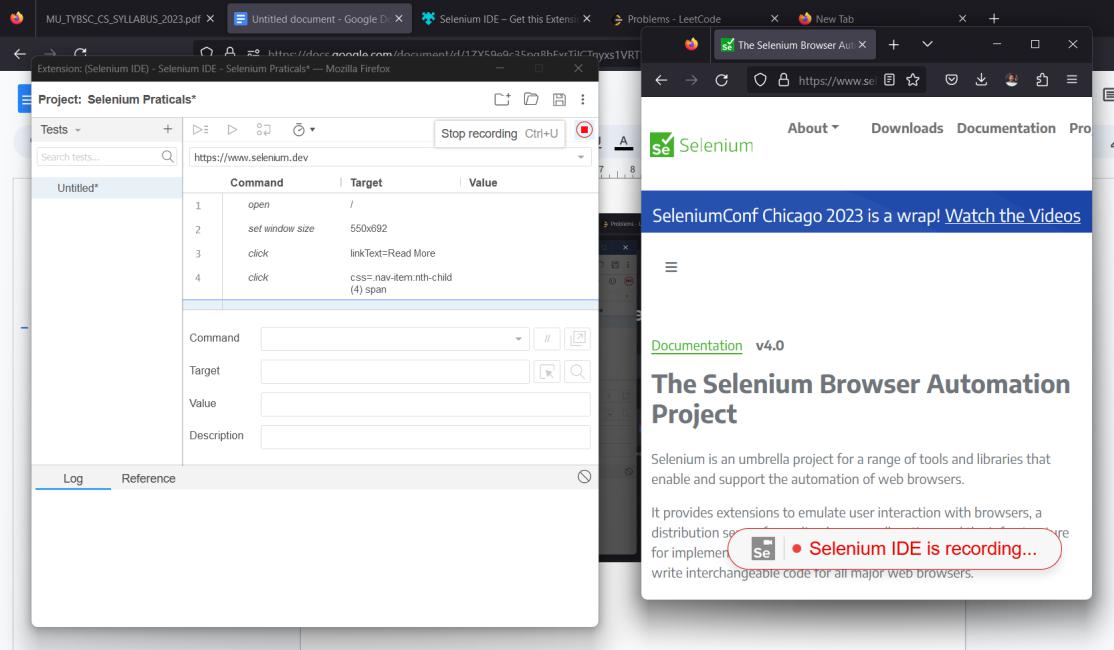
* **Click on “Record a new test in a new project”**

****

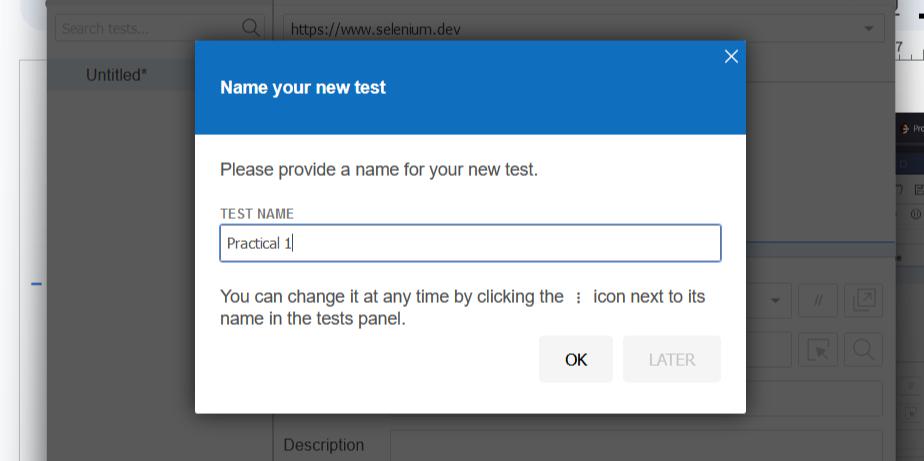
* **Type a Base Url**

****

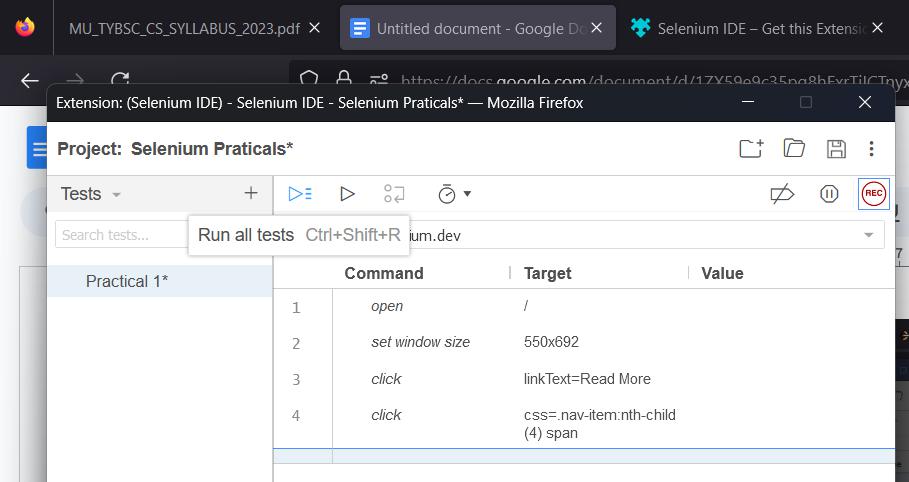
* **Now start recording and perform some actions on the viewed web page to record the tests.**

****

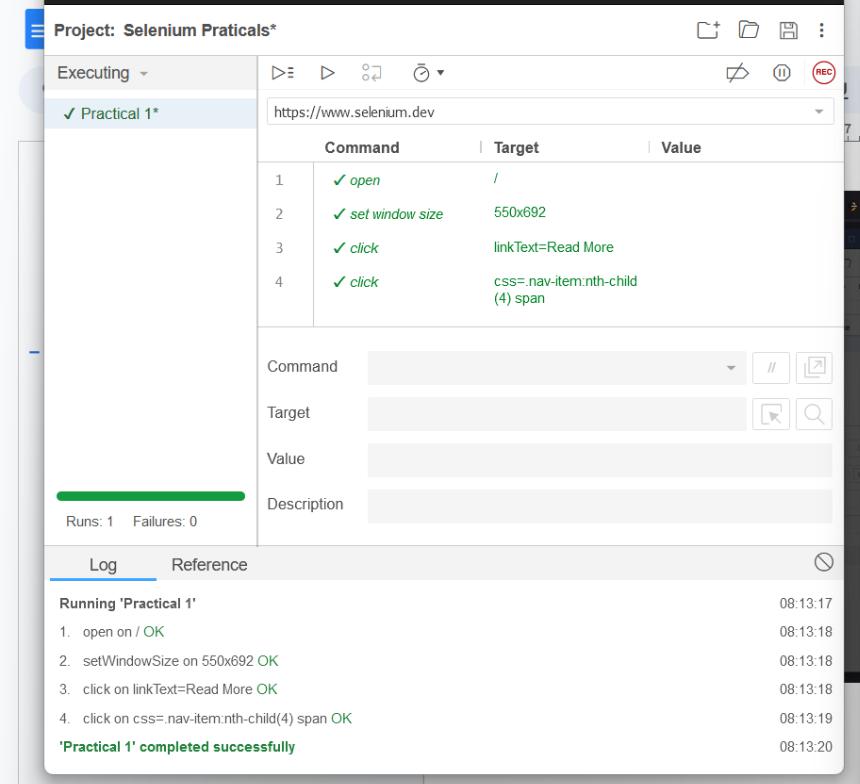
* **You will see the list of actions performed on the left side of the view.**
* **Press stop recording**
* **Give this test a name**

****

* **Now, click “Run all tests” to execute the recorded tests.**

****

* **It will execute all the recorded actions one by one ( \*No need to interfere in between ).**
* **After that it will produce a report below.**

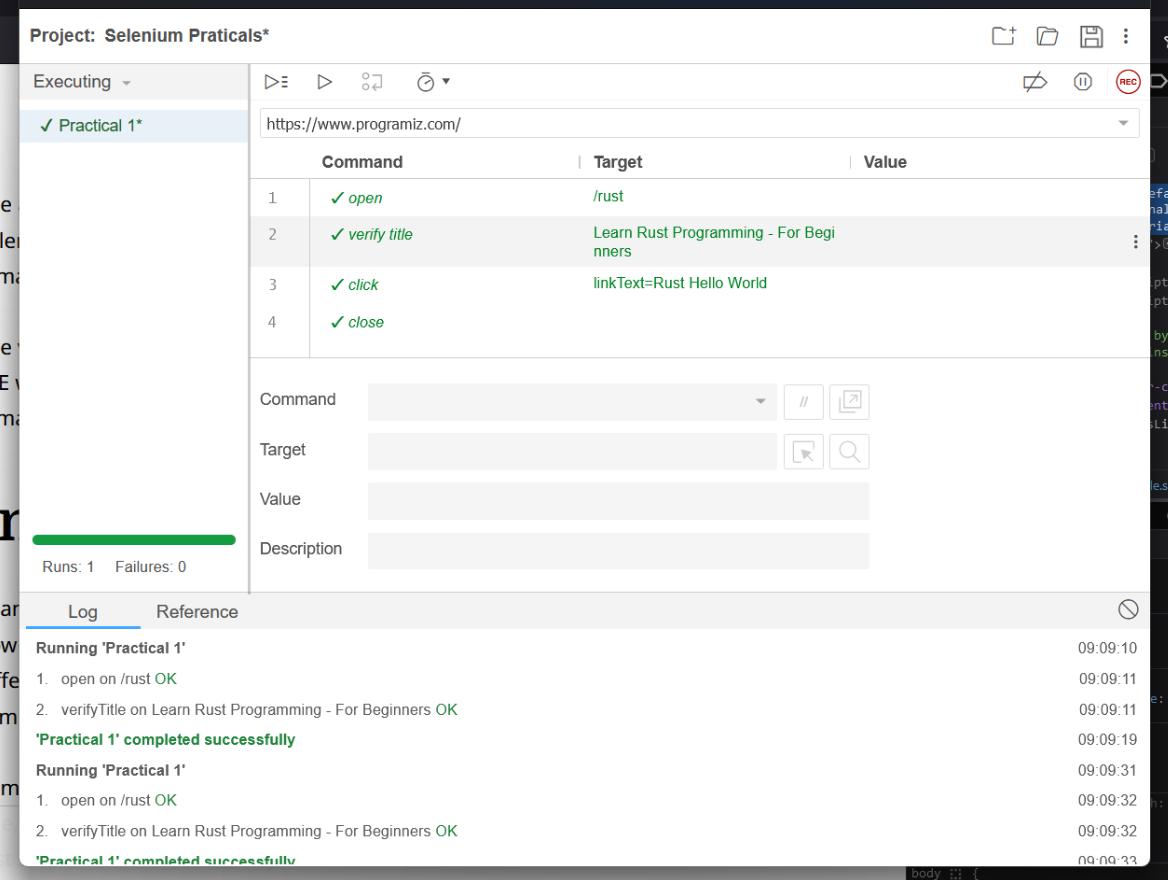
****

**Practical 2**

**Conduct a test suite for two different websites using Selenium IDE. Perform various actions like clicking links, filling forms, and verifying content.**

**A :**

* **Open the Selenium IDE**
* **Open Existing or New Project**
* **Set the Base Url or Playback Base Url to “**[**https://www.programiz.com/**](https://www.programiz.com/)**”**

****

**- Now add these following commands in the test case:**

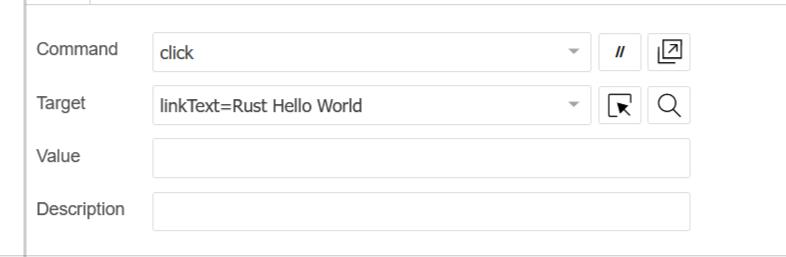
1. **Open “/rust” directory**

****

1. **Verify Title “Learn Rust Programming - For Beginners”**

****

1. **Click event to a target specifed “linkText = Rust Hello World”**

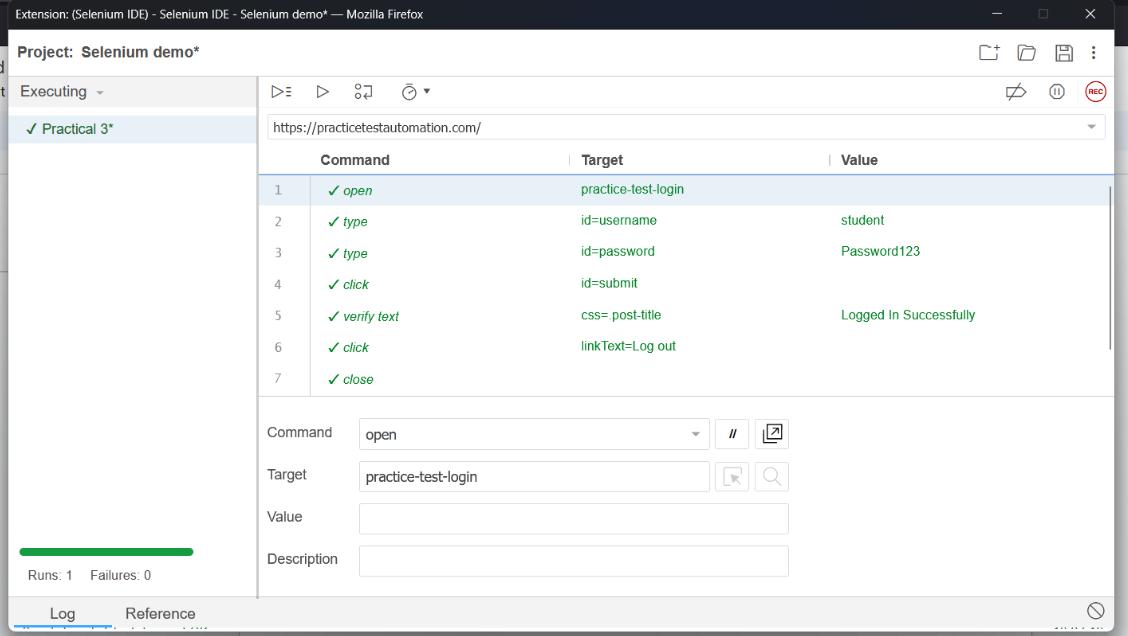
****

1. **Close event to exit the window**

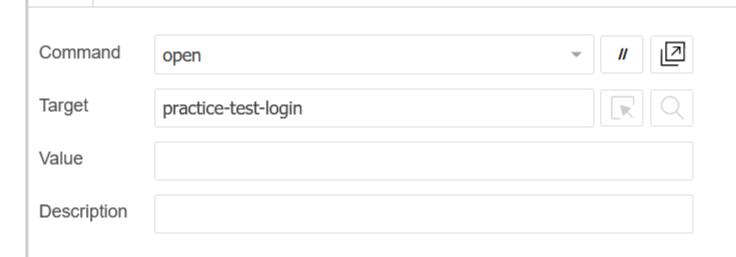
****

**B :**

* **Change Playback Base Url to “**[**https://practicetestautomation.com/**](https://practicetestautomation.com/)**”**

****

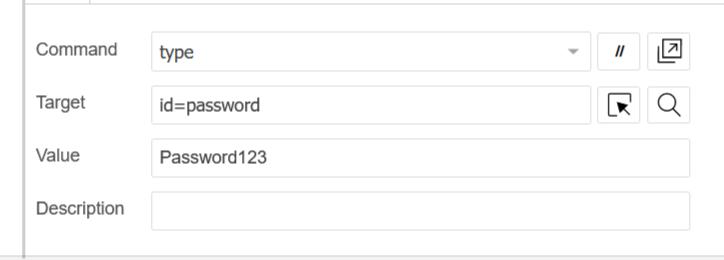
* **Add these following commands in test:**
  1. **Open “practice-test-login” directory**

****

1. **Type Target: id = username, Value: student**

****

1. **Type Target: id = password , Value: Password123**

****

1. **Click Target: id = submit**

****

1. **Verify Text Target: css =. post-title, Value: Logged In Successfully**

****

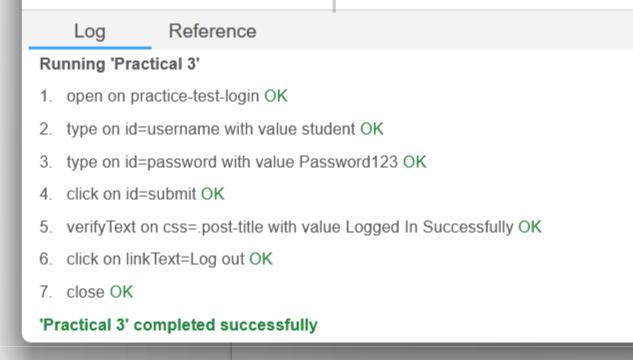
1. **Click Target: linkText = Log out**

****

1. **Close**

****

**Result:**

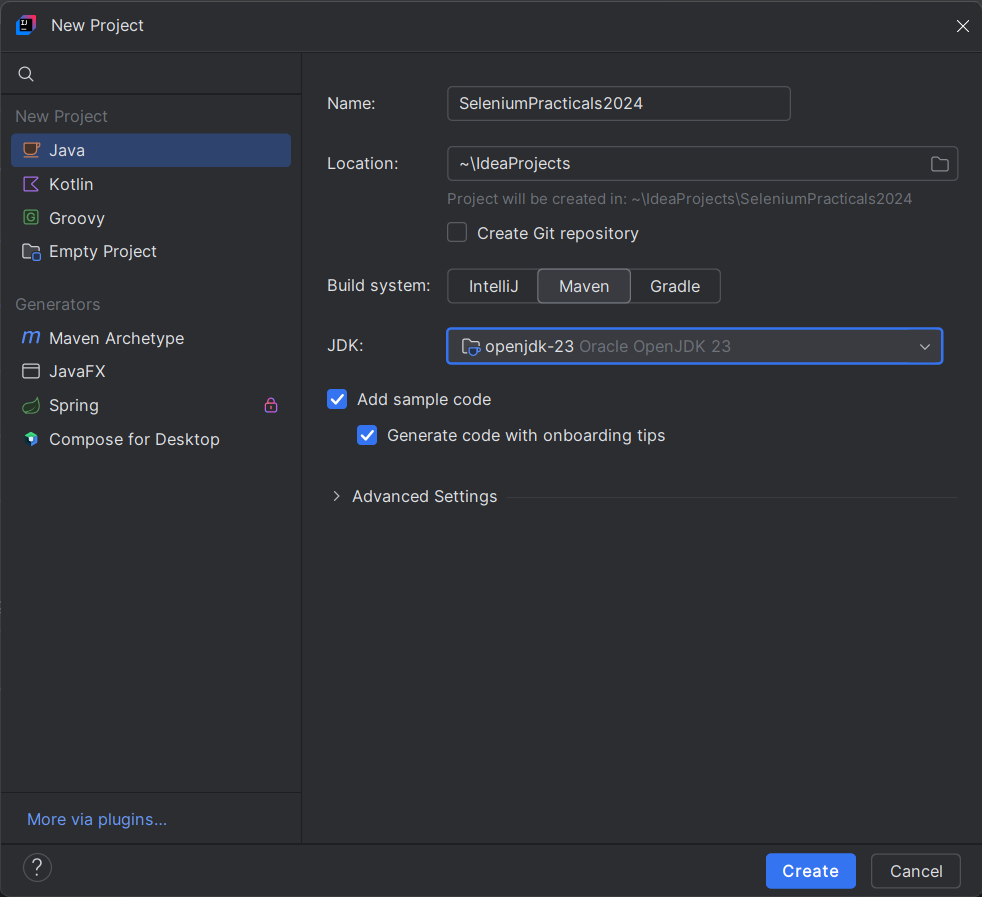
****

**Practical 3**

**Install Selenium Server (Selenium RC) and demonstrate its usage by executing a script in Java or PHP to automate browser actions.**

**Steps:**

* **Open IntelliJ IDEA IDE**
* **Create a new Maven Java Project** *(Build System: Maven)*



* **Now you will have to add this dependency to your pom.xml file**

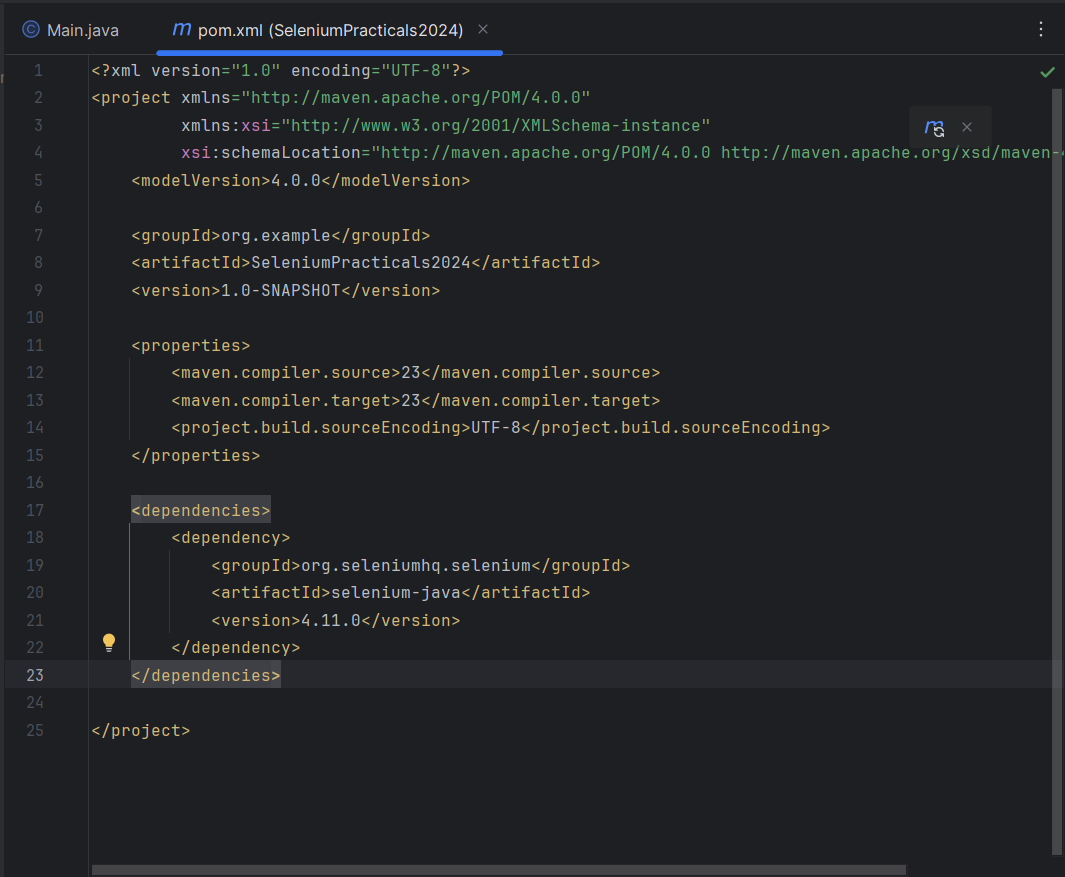
<dependencies>

<dependency>

<groupId>org.seleniumhq.selenium</groupId> <artifactId>selenium-java</artifactId> <version>4.11.0</version>

</dependency>

</dependencies>



**After adding this click on Load Marven Changes button or press Ctrl+Shift+O**

**To fetch all the dependencies** *(if not done can cause error)***.**

****

* **After it just add this code to Main.java file** package org.example;

import org.openqa.selenium.WebDriver;

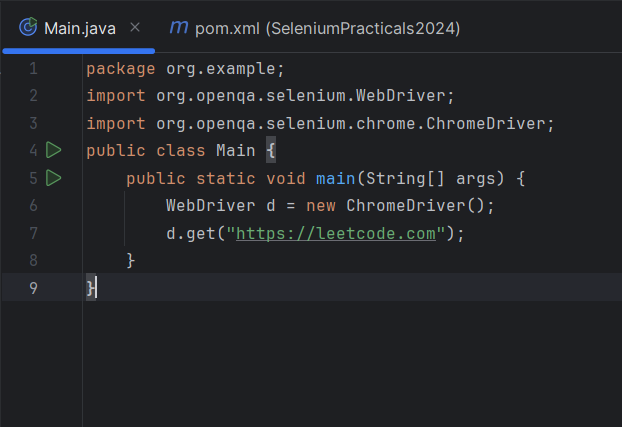
import org.openqa.selenium.chrome.ChromeDriver;

public class Main {

public static void main(String[] args) { WebDriver d = new ChromeDriver(); d.get("https://leetcode.com");

}

}

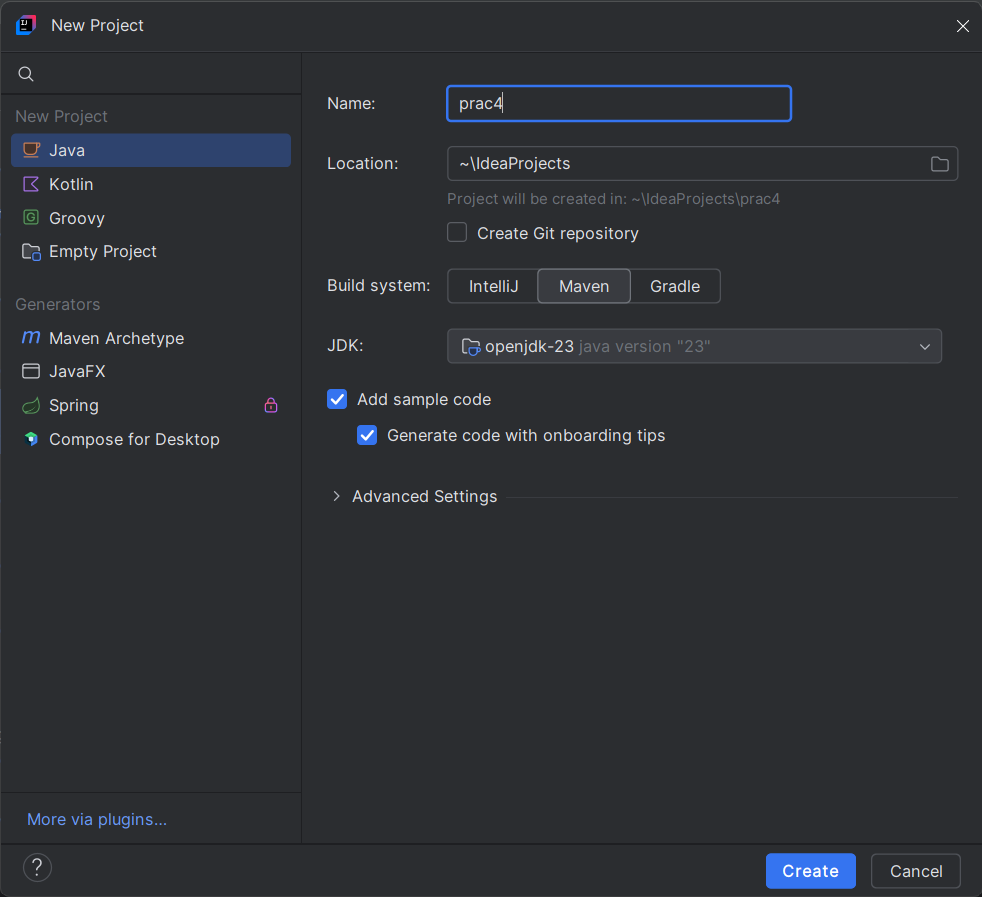


**Practical 4**

**Write a program using Selenium WebDriver to automate the login process on a specific web page. Verify successful login with appropriate assertions.**

**Steps:**

* **Open IntelliJ IDEA IDE**
* **Create a new Maven Java Project** *(Build System: Maven)*



* **Now you will have to add this dependency in your pom.xml file**

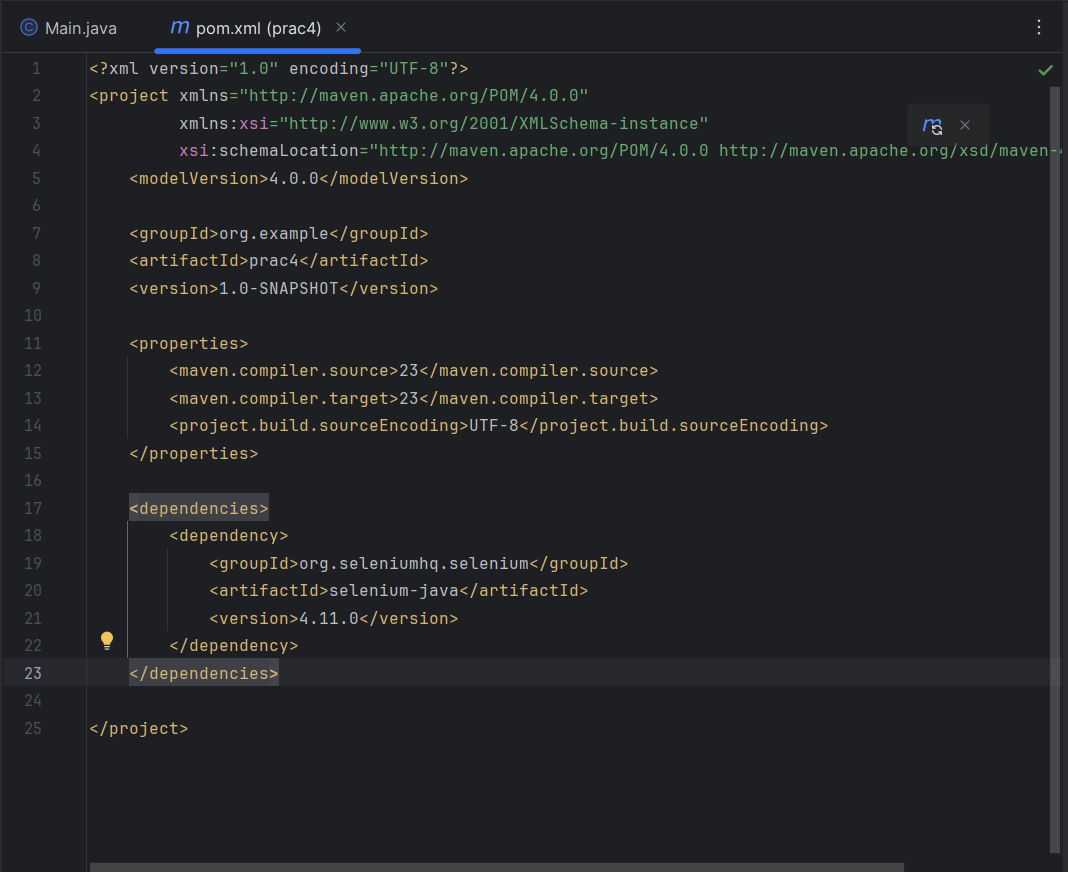
<dependencies>

<dependency>

<groupId>org.seleniumhq.selenium</groupId> <artifactId>selenium-java</artifactId> <version>4.11.0</version>

</dependency>

</dependencies>



* **After it just add this code to Main.java file**

package org.example;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.chrome.ChromeDriver;

public class Main {

public static void main(String[] args) { WebDriver driver = new ChromeDriver();

driver.get("https://practicetestautomation.com/practice-test-login/");

driver.fndElement(By.id("username")).sendKeys("student");

driver.fndElement(By.id("password")).sendKeys("Password123");

driver.fndElement(By.id("submit")).click();

assert(driver.getTitle().equals("Logged In Successfully | Practice Test

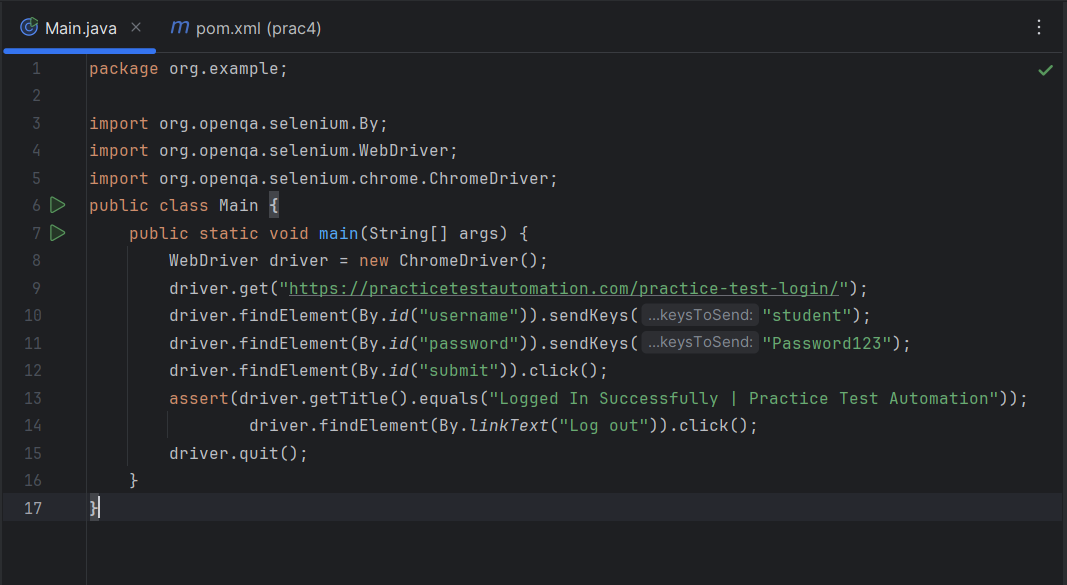
Automation"));

driver.fndElement(By.linkText("Log out")).click();

driver.quit();

}

}



* **Now run the program** *(You will be able to see an automated process to validate the tests.)*

**Practical 5**

**Write a program using Selenium WebDriver to update 10 student records in an Excel file. Perform data manipulation and verification.**

**Steps:**

* **ADD THIS DEPENDENCIES (pom.xml)**

<dependency>

<groupId>org.apache.poi</groupId> <artifactId>poi-examples</artifactId> <version>5.2.3</version>

</dependency>

<dependency>

<groupId>org.apache.logging.log4j</groupId> <artifactId>log4j-core</artifactId> <version>2.20.0</version>

</dependency>

<dependency>

<groupId>org.apache.poi</groupId>

<artifactId>poi</artifactId>

<version>5.2.3</version>

</dependency>

<dependency>

<groupId>org.apache.poi</groupId> <artifactId>poi-ooxml-lite</artifactId> <version>5.2.3</version>

</dependency>

* **ADD THIS CODE**

package org.example;  
  
import org.apache.poi.ss.usermodel.Cell;  
import org.apache.poi.ss.usermodel.Row;  
import org.apache.poi.xssf.usermodel.XSSFWorkbook;

import org.apache.poi.xssf.usermodel.XSSFSheet;

import java.io.File;  
import java.io.FileInputStream;  
import java.io.FileOutputStream;  
import java.io.IOException;  
  
public class Main {  
 static void witeData(String filepath, String filename, String  
 sheetName, String[] dataToWrite) throws IOException {  
 File file = new File(filepath + "\\" + filename);  
 FileInputStream fileInputStream = new FileInputStream(file);  
 XSSFWorkbook wb = new XSSFWorkbook(fileInputStream);  
 XSSFSheet sheet = wb.getSheet(sheetName);  
 Row row = sheet.getRow(0);  
 int rowCount = sheet.getLastRowNum() - sheet.getFirstRowNum();  
 Row newRow = sheet.createRow(rowCount + 1);  
 for (int i = 0; i < row.getLastCellNum(); i++) {  
 Cell cell = newRow.createCell(i);  
 cell.setCellValue(dataToWrite[i]);  
 }  
 fileInputStream.close();  
 FileOutputStream fileOutputStream = new FileOutputStream(file);  
 wb.write(fileOutputStream);  
 fileOutputStream.close();  
 }  
  
 public static void main(String[] args) throws IOException {  
 String[][] Sdata = {{"Student1",  
 "21"}, {"Student2", "22"}, {"Student3", "20"}, {"Student4", "21"}, {"Student5", "21"}, {  
 "Student6", "23"}, {"Student7", "19"}, {"Student8", "22"}, {"Student9", "20"}, {"Student10", "19"}};  
 for (String[] data : Sdata) {  
 witeData(System.getProperty("user.dir") + "\\src\\main\\resources", "Practical5.xlsx", "Sheet1", data);  
 }  
 }  
}

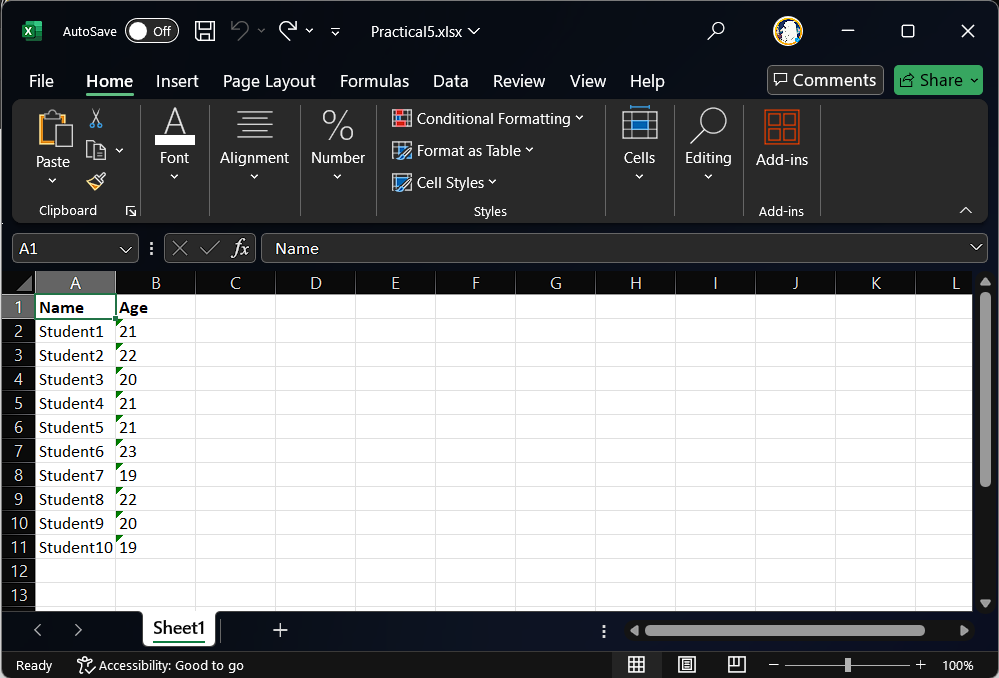
**Create an excel file named** [**Practical5.xlsx**](https://github.com/Yashv2229/TYCS-Practicals/blob/efb20bbb54e855fdd8cda2ccbec9fbe07d77b5ed/SQTA/Practical%205%20(Update%20record%20excel)/Practical5.xlsx) **in “src/main/resources/Practical5.xlsx” containing two columns Name and Age, Sheet name should be Sheet1**

Can download files from clicking on file name

****

**Now run the program**

**Output:**

****

**Practical 6**

**Write a program using Selenium WebDriver to select the number of students who have scored more than 60 in any one subject (or all subjects). Perform data extraction and analysis.**

**Steps:**

1. **Make sure that you have added all the dependencies in the pom.xml file**

*( i.e Apache POI )***.**

1. **Write this Code:**

package org.example;

import org.apache.poi.xssf.usermodel.XSSFSheet;

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

import java.io.File;

import java.io.FileInputStream;

import java.io.IOException;

public class Main {

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

File file = new File(System.getProperty("user.dir")+"\\src\\main\\resources\\Practical6.xlsx");

FileInputStream fileInputStream = new FileInputStream(file);

XSSFWorkbook wb = new XSSFWorkbook(fileInputStream);

XSSFSheet Sheet = wb.getSheet("Student\_Marks");

System.out.println("STUDENTS NAME\n");

int st=0;

for (int i=1; i<=Sheet.getLastRowNum(); i++) {

boolean \_v = false;

for (int j=1; j<Sheet.getRow(i).getLastCellNum(); j++) {

if (Sheet.getRow(i).getCell(j).getNumericCellValue() > 60) {

\_v = true;

st++;

break;

}

}

if (\_v)

System.out.println(Sheet.getRow(i).getCell(0).getStringCellValue());

}

System.out.println("\nNO. OF STUDENTS: "+st);

fileInputStream.close();

wb.close();

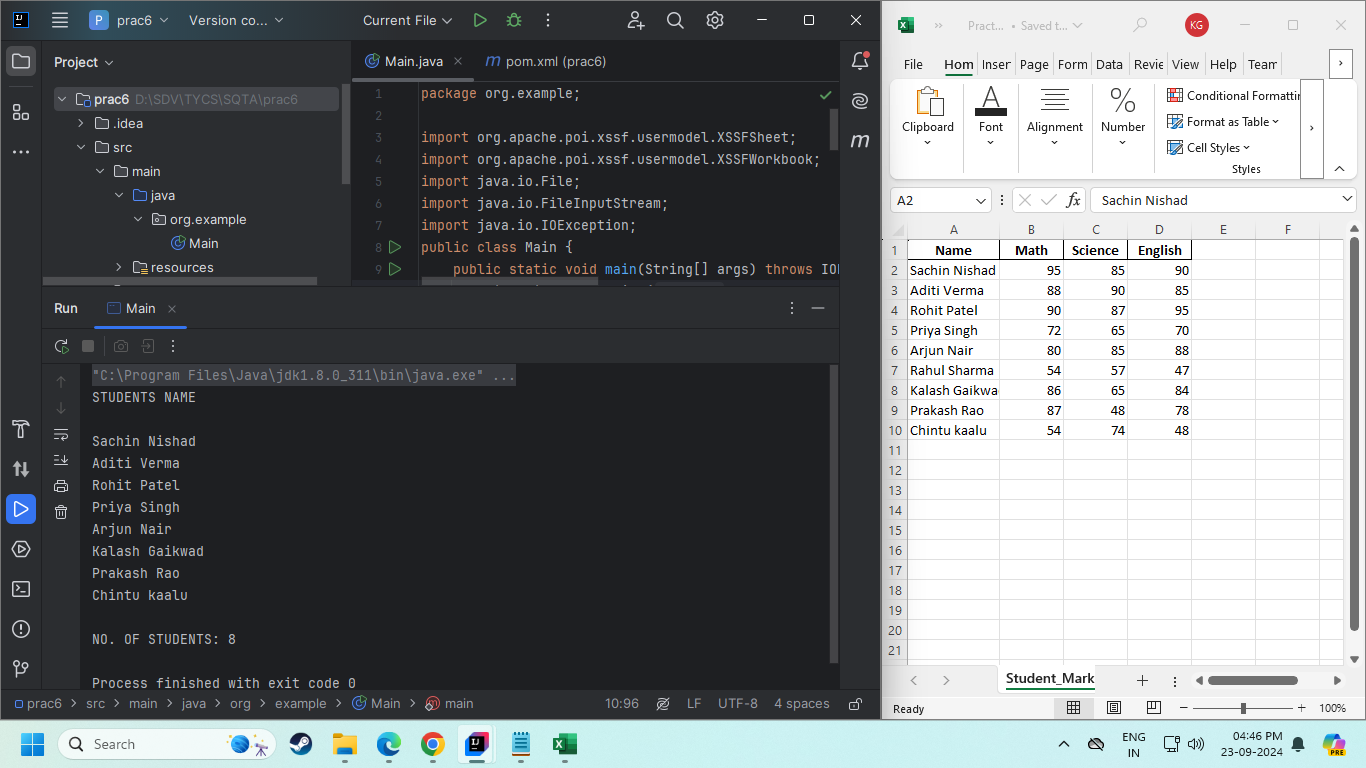
}

}

**Create an excel file named** [**Practical6.xlsx**](https://github.com/Yashv2229/TYCS-Practicals/tree/efb20bbb54e855fdd8cda2ccbec9fbe07d77b5ed/SQTA/Practical%206%20(data%20extraction%20and%20analysis)) **in “src/main/resources/Practical6.xlsx” containing columns Name, Math, Science and English, Sheet name should be Student\_Marks.**

Can download files from clicking on file name

**Output:**

****

**Practical 7**

**Write a program using Selenium WebDriver to provide the total number of objects present or available on a web page. Perform object identification and counting.**

**Steps:**

1. **Make sure you have added dependencies in the pom.xml file**

*( i.e Selenium Java )***.**

1. **ADD THIS CODE:**

package org.example;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.chrome.ChromeDriver;

import java.util.List;

public class Main {

public static void main(String[] args) {

WebDriver wd = new ChromeDriver();

wd.get("https://programiz.com");

List<WebElement> elements = wd.findElements(By.xpath("//\*"));

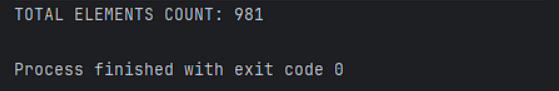
int elementsCount = elements.size();

System.out.println("TOTAL ELEMENTS COUNT: "+elementsCount);

}

}

**Output:**

****

**Practical 8**

**Write a program using Selenium WebDriver to get the number of items in a list or combo box on a web page. Perform element identification and counting.**

**Steps:**

1. **Make sure you have added dependencies in the pom.xml file**

*( i.e Selenium Java ).*

1. **ADD THIS CODE:**

package org.example;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.chrome.ChromeDriver;

import java.util.List;

public class Main {

public static void main(String[] args) {

WebDriver wd = new ChromeDriver();

wd.get("https://practicetestautomation.com/practice/");

WebElement ul = wd.findElement(By.tagName("ul"));

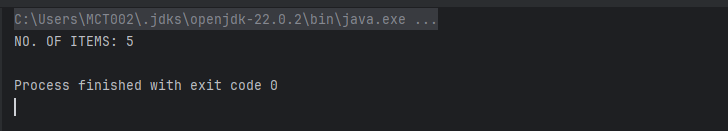
List list = ul.findElements(By.tagName("li"));

System.out.println("NO. OF ITEMS: "+list.size());

}

}

**Output:**



**Practical 9**

**Write a program using Selenium WebDriver to count the number of checkboxes on a web page, including checked and unchecked counts. Perform checkbox identification and counting.**

**Steps:**

1. **Make sure you have added dependencies in the pom.xml fle**

*( i.e Selenium Java ).*

1. **ADD THIS CODE:**

package org.example;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.chrome.ChromeDriver;

import java.util.List;

public class Main {

public static void main(String[] args) {

WebDriver wd = new ChromeDriver();

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

List checkBoxCounts =

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

System.out.println("NO. OF CHECKBOXES: "+checkBoxCounts.size());

}

}

**Output:**

