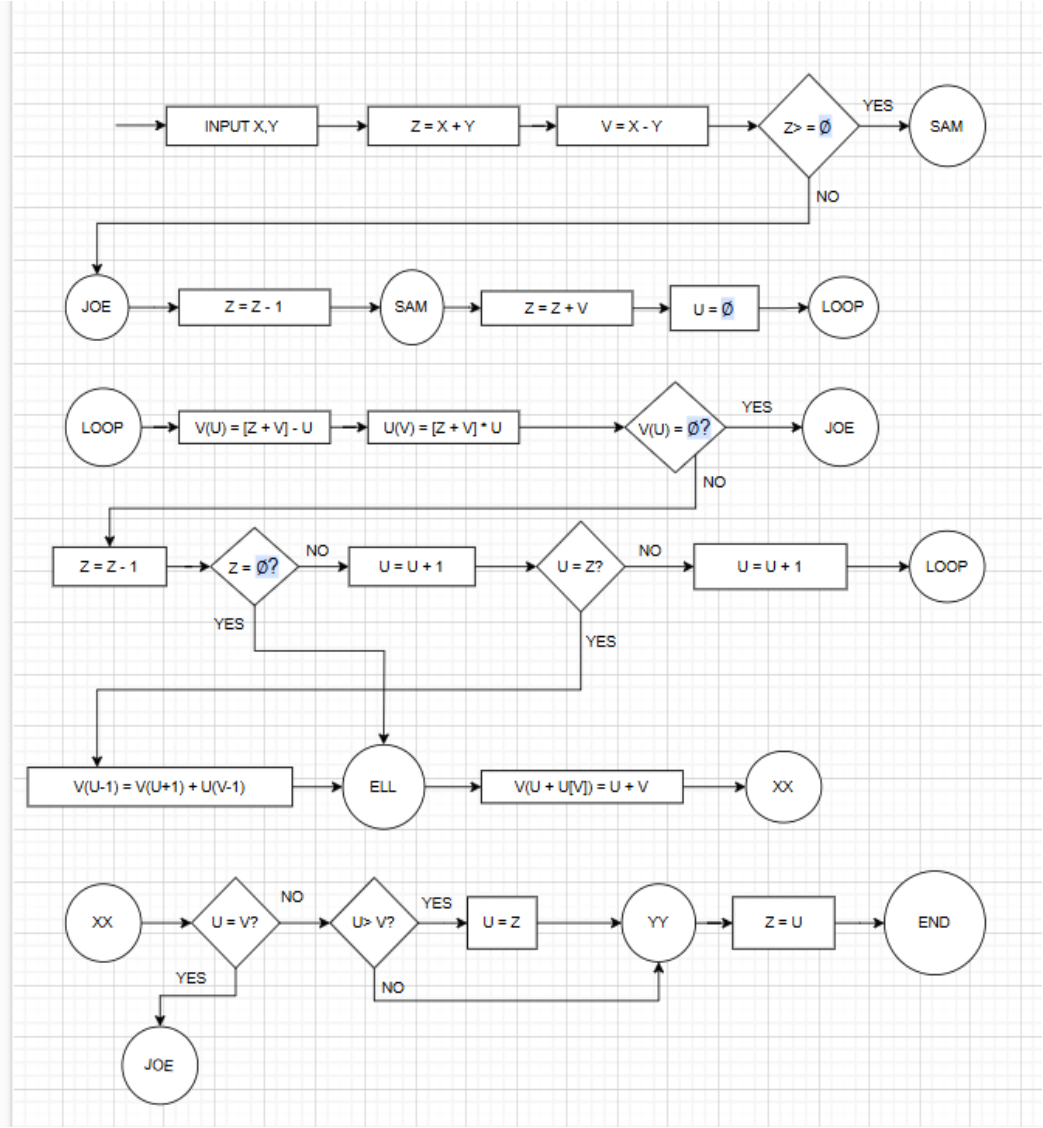
PRACTICAL-1

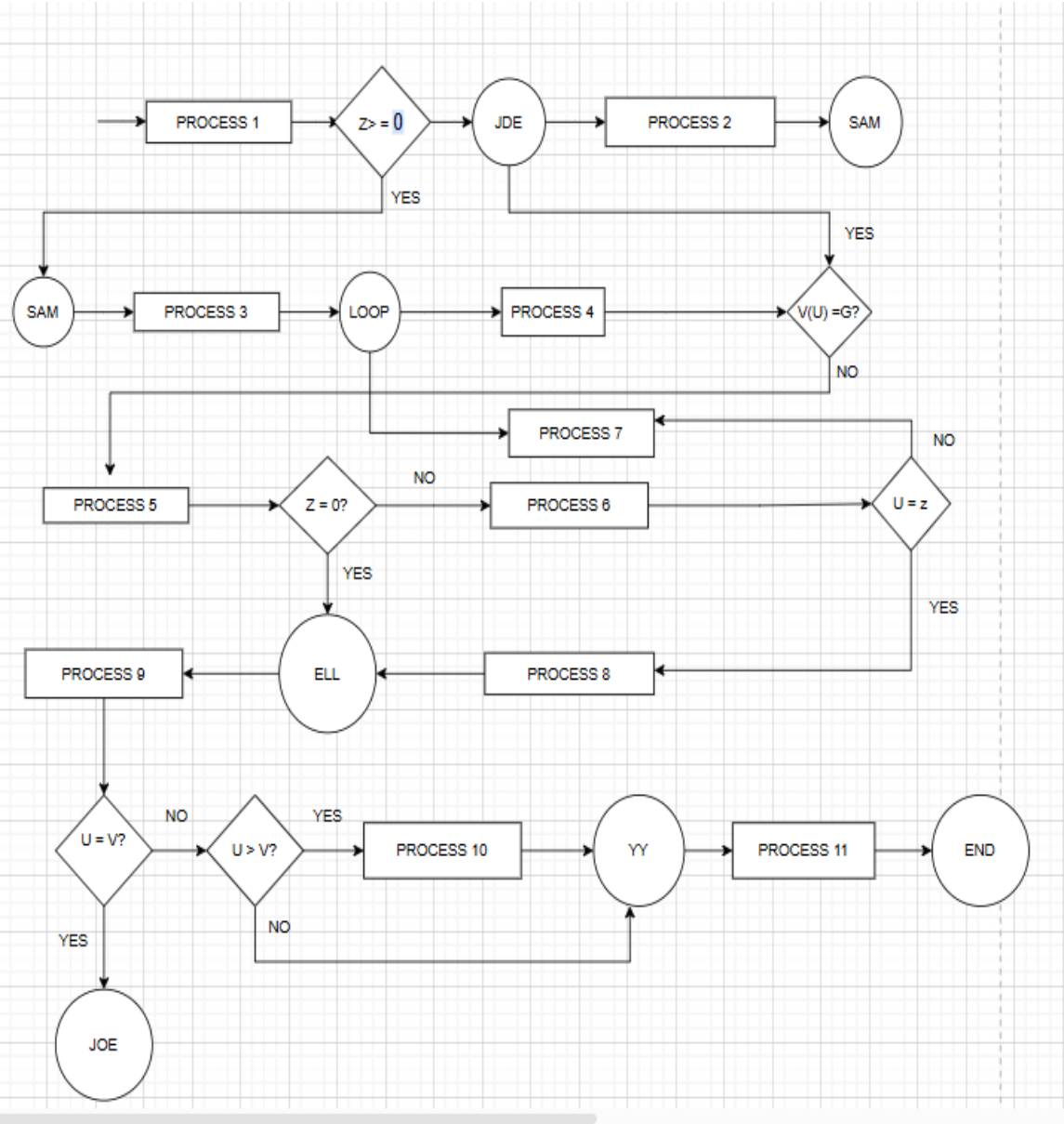
Aim: Flow graphs and Path Testing using Some elements

Output:

one-to-one flowchart :



Control Flowgraph :



Practical :- 2

Aim: - Installation of Selenium IDE & case test using Selenium IDE

To set up Selenium with TestNG, Eclipse, and Chrome Driver for automation testing in Java, follow these steps:

Steps:

Install Java Development kit (JDK)

Ensure that you have the JDK installed on your system. You can download it from the Oracle website or install it using a package manager like apt-get for Linux.

To verify if JDK is installed, open the terminal (Command Prompt for Windows) and run:

Java -version

Install Eclipse IDE

Download Eclipse: Go to the Eclipse Downloads page and download the "Eclipse IDE for Java Developers".

Install Eclipse: Follow the installation instructions based on your operating system.

After installing Eclipse Create a new project -> Mavan Project -> Add Group Id and Artifact ID and create a simple mavan project.

To set up selenium go to Mavan Repository(https://mvnrepository.com/) and search for selenium java -> copy the dependency for version 3.141.5(https://mvnrepository.com/artifact/org.seleniumhq.selenium/selenium-java/3.141.5) and paste it in the pom.xml file of your mavan project.

To set up testng go to Mavan Repository(https://mvnrepository.com/) and search for testng -> copy the dependency for version 6.14.3(https://mvnrepository.com/artifact/org.testng/testng) and paste it in the pom.xml file of your mavan project.

Install TestNG Plugin for Eclipse:

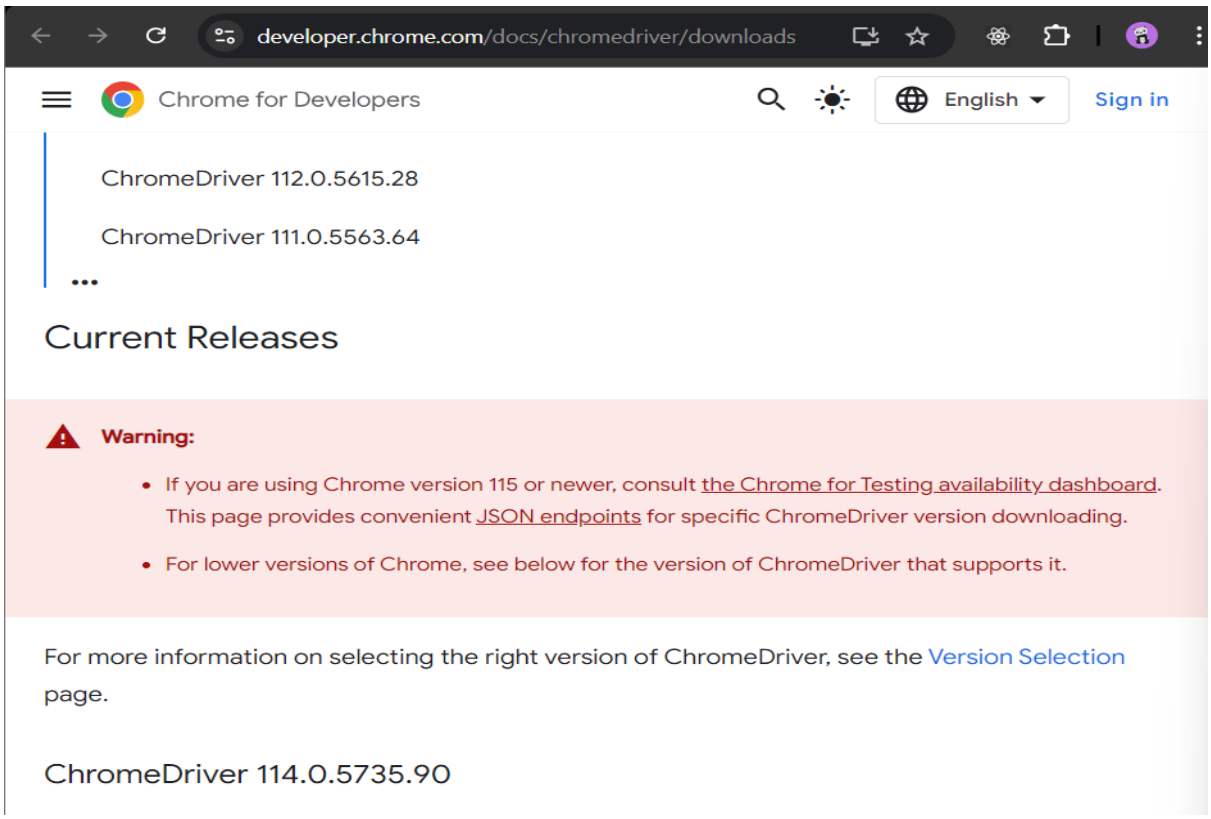
Go to Help > Eclipse Marketplace.

Search for "TestNG" and click Go.

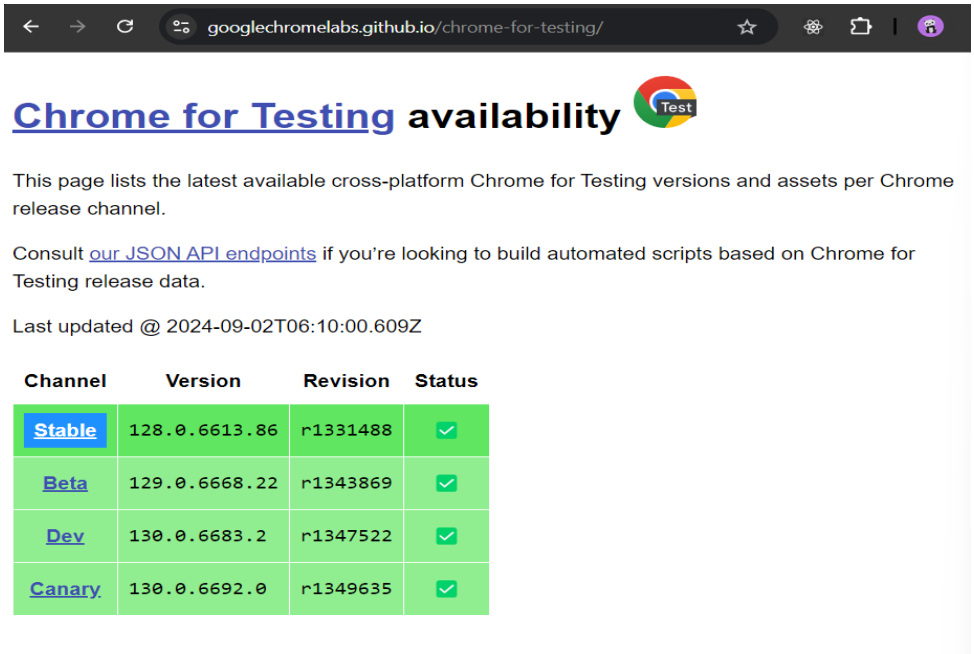
Install the TestNG plugin and restart Eclipse.

Install Chrome Driver

Search for chrome driver -> Click on chrome for testing availability dashboard.



Click on stable



Copy the URL according to your device and paste it in the browser to download it

		win64.zip	
chromedriver	linux64	https://storage.googleapis.com/chrome-for-testing-public/128.0.6613.86/linux64/chromedriver-linux64.zip	200
chromedriver	mac-arm64	https://storage.googleapis.com/chrome-for-testing-public/128.0.6613.86/mac-arm64/chromedriver-mac-arm64.zip	200
chromedriver	mac-x64	https://storage.googleapis.com/chrome-for-testing-public/128.0.6613.86/mac-x64/chromedriver-mac-x64.zip	200
chromedriver	win32	https://storage.googleapis.com/chrome-for-testing-public/128.0.6613.86/win32/chromedriver-win32.zip	200
chromedriver	win64	https://storage.googleapis.com/chrome-for-testing-public/128.0.6613.86/win64/chromedriver-win64.zip	200

After the file is downloaded Extract the file and copy the chromedriver.exe file. -> open eclipse creates a new folder with name “drivers” and paste the copied chromedriver.exe file in it.

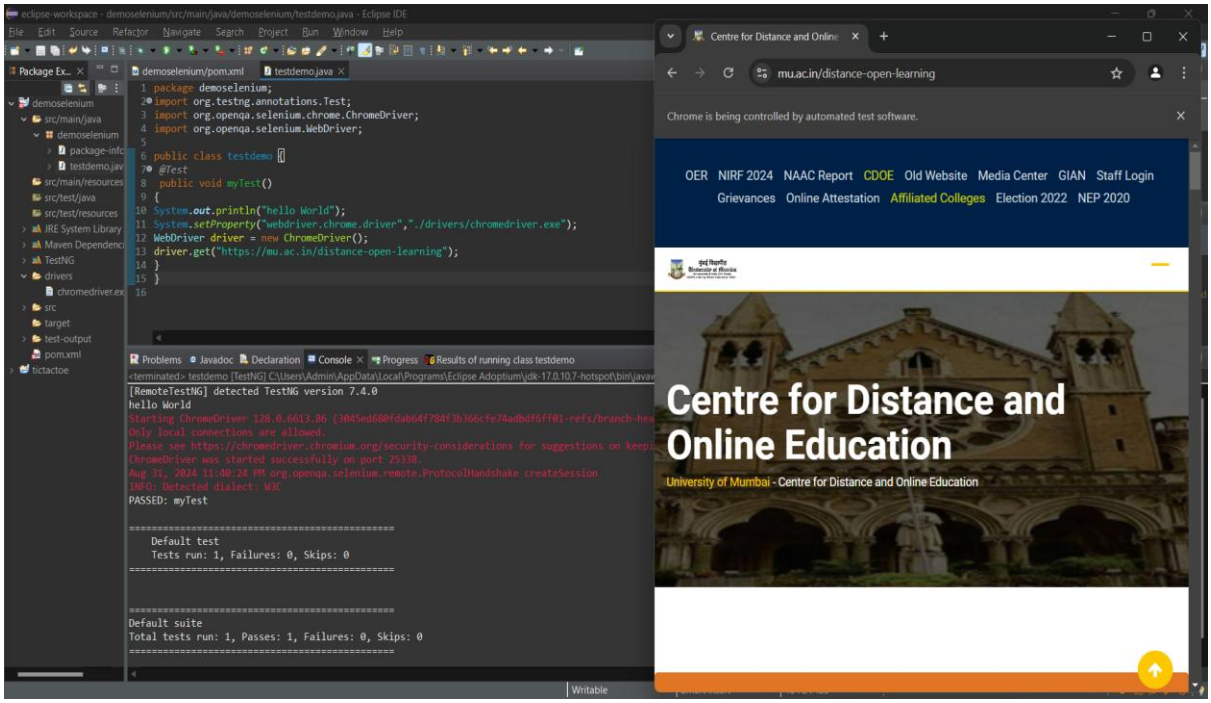
Create a Package and class in the project -> add a new function -> add a system.out. println(“Hello World”);

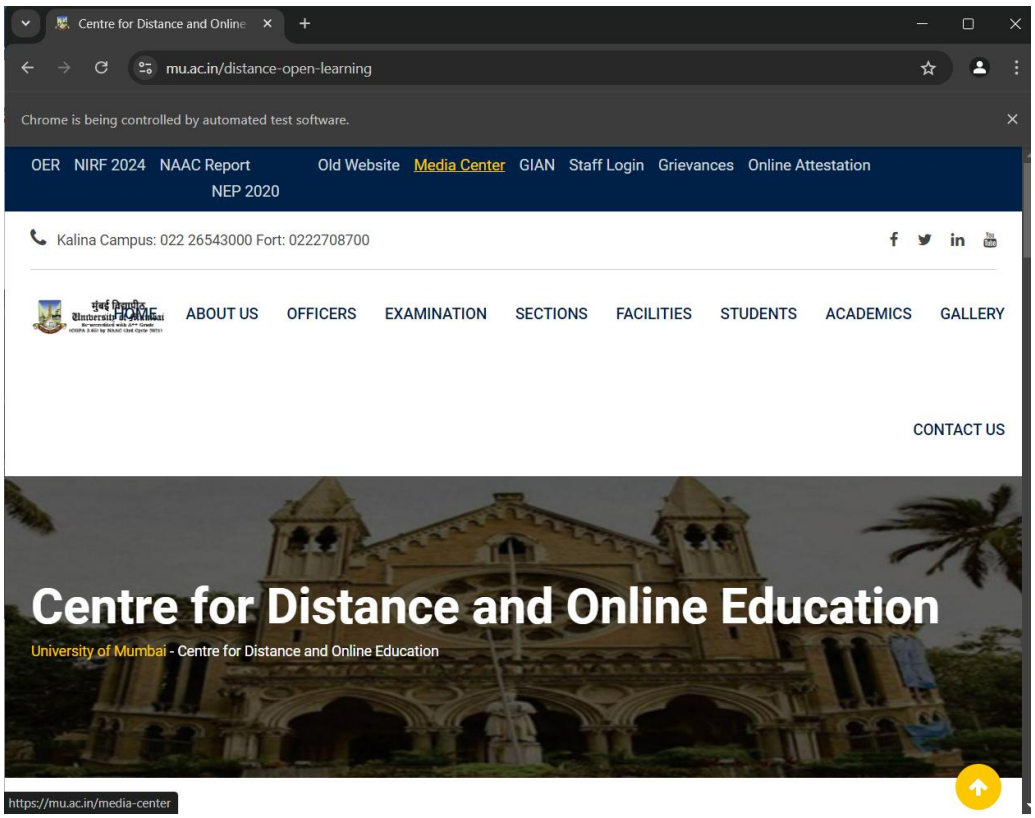
And the test it as Run as -> Testng Test. Check your console for the output.

```
package demoselenium;

import org.testng.annotations.Test;

import org.openqa.selenium.chrome.ChromeDriver;

import org.openqa.selenium.WebDriver;

public class testdemo {

 @Test

 public void myTest()

{

System.out.println("hello World");

System.setProperty("webdriver.chrome.driver","./drivers/chromedriver.exe");
```

```
WebDriver driver = new ChromeDriver();

driver.get("https://mu.ac.in/distance-open-learning");

}}
```

OUTPUT:-





Practical 3

Aim: Implementing handling multiple frames

CODE:-

```
package demoselenium;

import java.util.ArrayList;

import java.util.List;

import java.util.Set;

import org.testng.annotations.Test;

import org.openqa.selenium.chrome.ChromeDriver;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;


public class testdemo {

 @Test

 public void myTest()

{

//System.out.println("hello World");

System.setProperty("webdriver.chrome.driver","./drivers/chromedriver.exe");

WebDriver driver = new ChromeDriver();

driver.get("https:kjsit.somaiya.edu.in/en/");

System.out.println("Parent:"+driver.getTitle());

String parentwindow = driver.getWindowHandle();

System.out.println(parentwindow);

driver.findElement(By.linkText("B.Tech.(IT)")).click();


Set<String>Windowhandles = driver.getWindowHandles();

List<String>Windows = new ArrayList<String>(Windowhandles);

driver.switchTo().window(Windows.get(1));
```

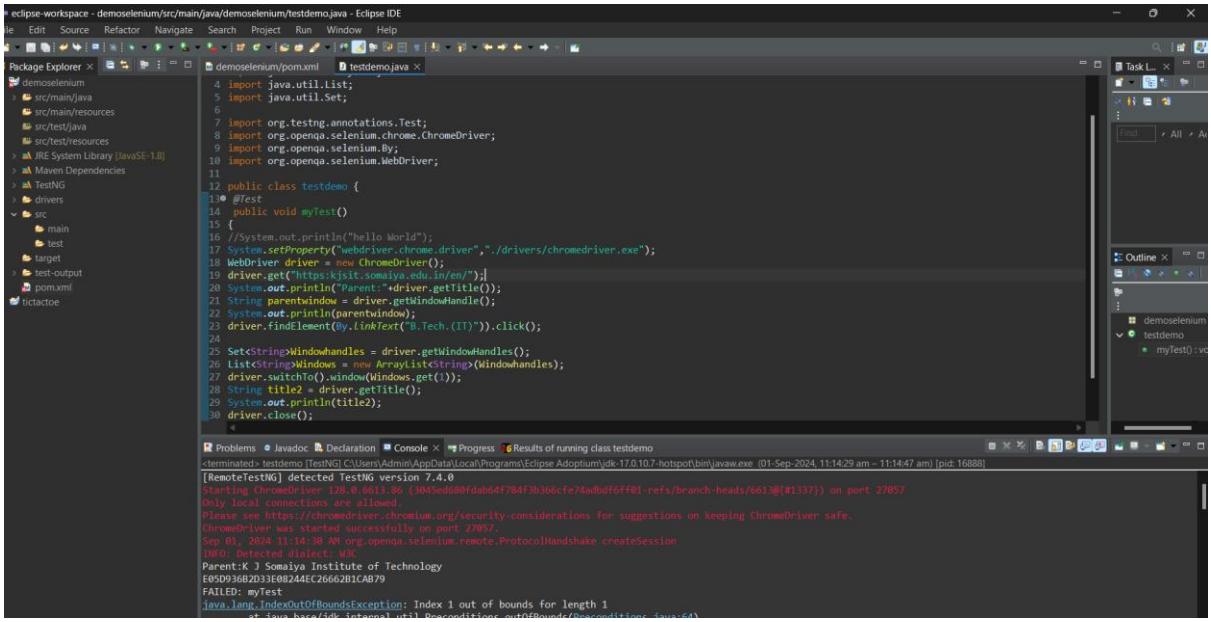
String title2 = driver.getTitle();

System.***out***.println(title2);

driver.close();

}}

OUTPUT:-




Bachelor of Technology Informa
kjsit.somaiya.edu.in/en/programme/information-technology-engineering
Chrome is being controlled by automated test software.
SOMAIYA VIDYAVIHAR
K J Somaiya Institute of Technology
somaiya.edu.in
STUDENT FACULTY STAFF
ALUMNI EMERGENCY CAREER GRIEVANCE
Somaiya TRUST
About | Programmes | Admissions | Academics | Research | Placement | Industry Connect | IQAC | Student Life
Bachelor of Technology
Information Technology | B.Tech (IT)
Information Technology is at the forefront of innovation and is a driving force in the digital transformation of the world. Therefore, graduates of Information Technology degrees take up stimulating employment with national and global industry leaders. The Bachelor of Engineering in Information Technology (B.Tech. I.T.) programme provides the skills required succeed at the cutting edge of the digital economy.
Considering the future aspects of this rapidly developing sector, the programme is designed to ensure that graduates possess the ability to design and implement up to date
Key Information
Duration
4 Years
Programme Code
ST2207
Course Type
Bachelor Degree
Mode of study
Full time
Campus
Sion - Mumbai
Institute

Practical 4

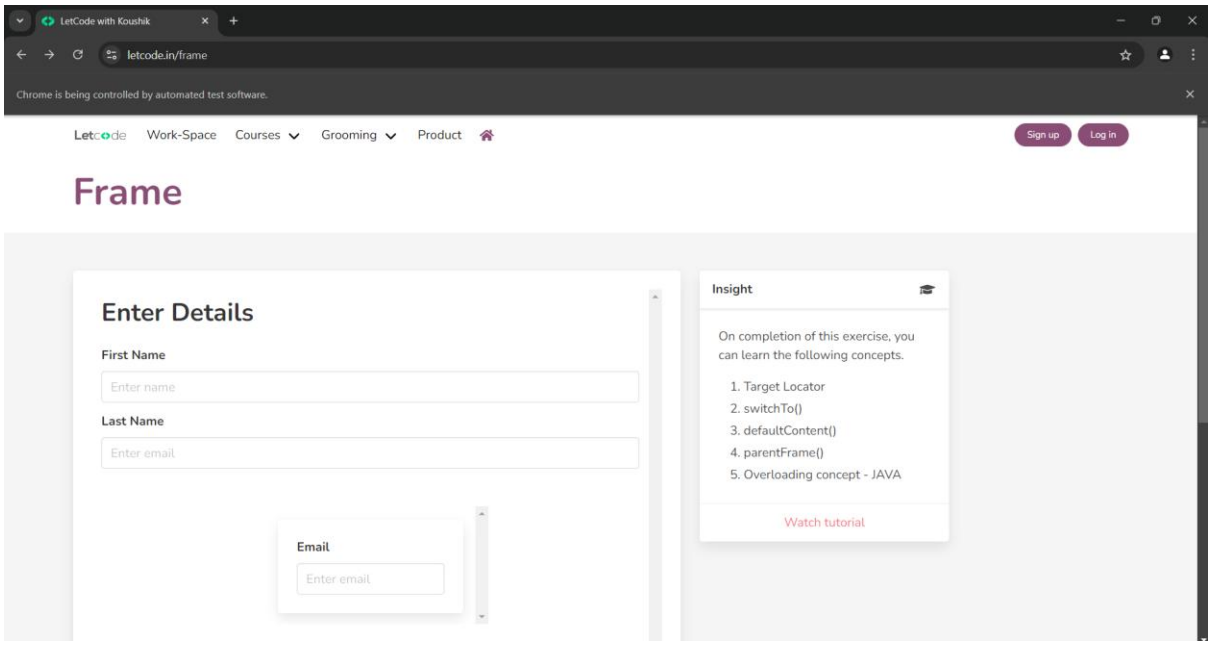
Aim: Implementing Selenium WebDriver- find element command, Locator(id,css selector, Xpath), Input Box, Buttons, Submit Buttons.

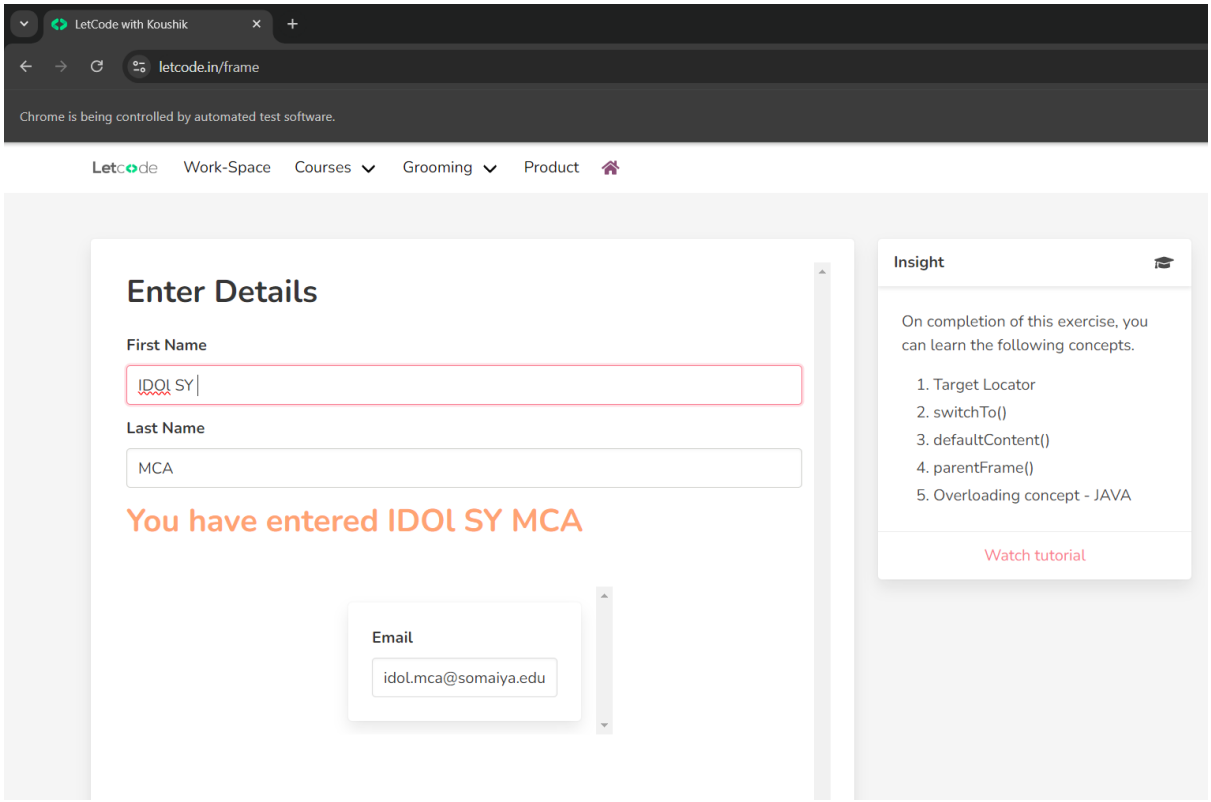
CODE:-

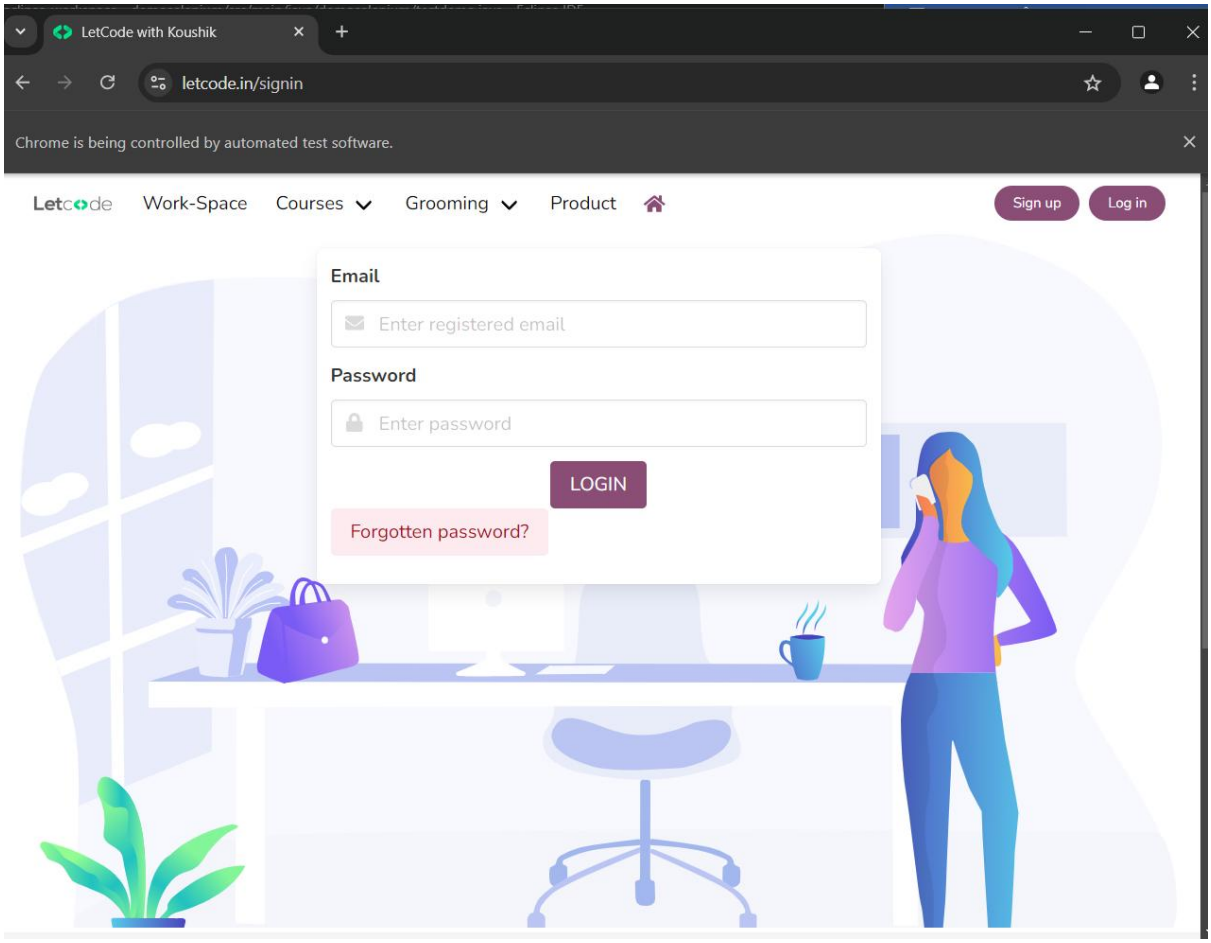
```
package demoselenium;

//import java.util.ArrayList;

//import java.util.List;

//import java.util.Set;

//import org.testng.annotations.Test;

import org.openqa.selenium.chrome.ChromeDriver;

import org.openqa.selenium.By;

//import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;


public class testdemo {

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

		System.setProperty("webdriver.chrome.driver", "./drivers/chromedriver.exe");

		ChromeDriver driver = new ChromeDriver();

		driver.get("https://letcode.in/frame");

		//frame

		//driver.switchTo().frame("firstFr");

		//driver.switchTo().frame(0);

		WebElement frameEle = driver.findElementByXPath("//*[@id=\"firstFr\"]");

		driver.switchTo().frame(frameEle);

		driver.findElement(By.name("fname")).sendKeys("IDOL");

		driver.findElement(By.name("lname")).sendKeys("MCA");

		Thread.sleep(2000);

		//nested frame
```

```
WebElement innerFrames = driver.findElement(By.cssSelector("iframe.has-background-white"));

driver.switchTo().frame(innerFrames);

driver.findElement(By.name("email")).sendKeys("idol.mca@somaiya.edu");

Thread.sleep(2000);

//parent Frame

driver.switchTo().parentFrame();

driver.findElement(By.name("fname")).sendKeys(" SY ");

Thread.sleep(2000);

//default frame

driver.switchTo().defaultContent();

driver.findElement(By.linkText("Log in")).click();

Thread.sleep(2000);

//                driver.quit();      }}
```

OUTPUT:-




LetCode with Koushik
letcode.in/frame
Chrome is being controlled by automated test software.
Letcode
Work-Space
Courses
Grooming
Product
Enter Details
First Name
IDOl SY
Last Name
MCA
You have entered IDOl SY MCA
Email
idol.mca@somaiya.edu
Insight
On completion of this exercise, you can learn the following concepts.
1. Target Locator
2. switchTo()
3. defaultContent()
4. parentFrame()
5. Overloading concept - JAVA
Watch tutorial


LetCode with Koushik
letcode.in/signin
Chrome is being controlled by automated test software.
Letcode
Work-Space
Courses
Grooming
Product
Sign up
Log in
Email
Enter registered email
Password
Enter password
LOGIN
Forgotten password?

Practical 5

Aim: Implementing methods in TestNG file.

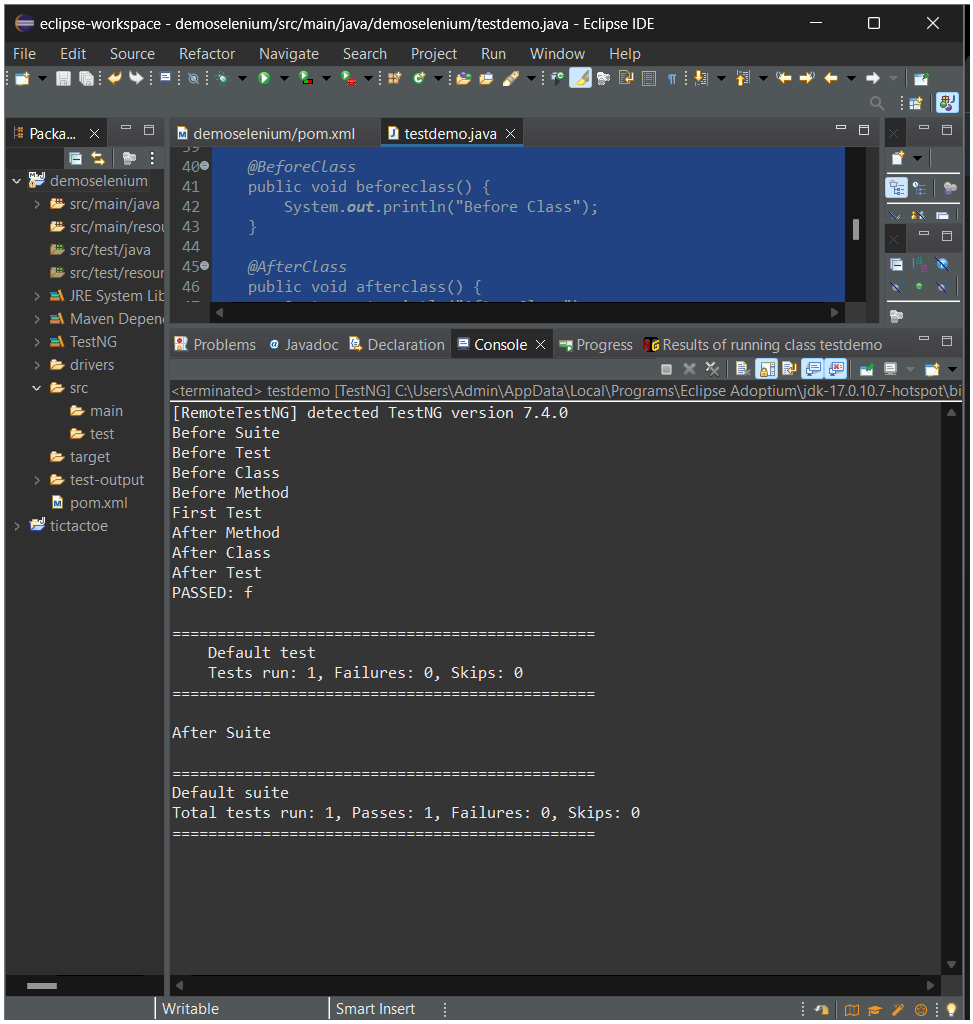
CODE:-

```
package demoselenium;
//import java.util.ArrayList;
//import java.util.List;
//import java.util.Set;
//import org.testng.annotations.Test;
//import org.openqa.selenium.chrome.ChromeDriver;
import org.testng.annotations.AfterClass;
import org.testng.annotations.AfterMethod;
import org.testng.annotations.AfterSuite;
import org.testng.annotations.AfterTest;
import org.testng.annotations.BeforeClass;
import org.testng.annotations.BeforeMethod;
import org.testng.annotations.BeforeSuite;
import org.testng.annotations.BeforeTest;
import org.testng.annotations.Test;
//import org.openqa.selenium.By;
//import org.openqa.selenium.WebDriver;
//import org.openqa.selenium.WebElement;

public class testdemo {
	@Test
	public void f() {
		System.out.println("First Test");
	}
	@BeforeMethod
	public void beforemethod() {
```

```
        System.out.println("Before Method");
}

@AfterMethod
public void aftermethod() {
        System.out.println("After Method");
}

@BeforeClass
public void beforeclass() {
        System.out.println("Before Class");
}

@AfterClass
public void afterclass() {
        System.out.println("After Class");
}
 @BeforeTest
public void beforeTest()
 {
         System.out.println("Before Test");
 }
 @AfterTest
 public void afterTest()
 {
         System.out.println("After Test");
 }
 @BeforeSuite
 public void beforeSuite()
```

```
{
        System.out.println("Before Suite");
}

@AfterSuite
public void afterSuite()
{
        System.out.println("After Suite");
}}
```

OUTPUT:-



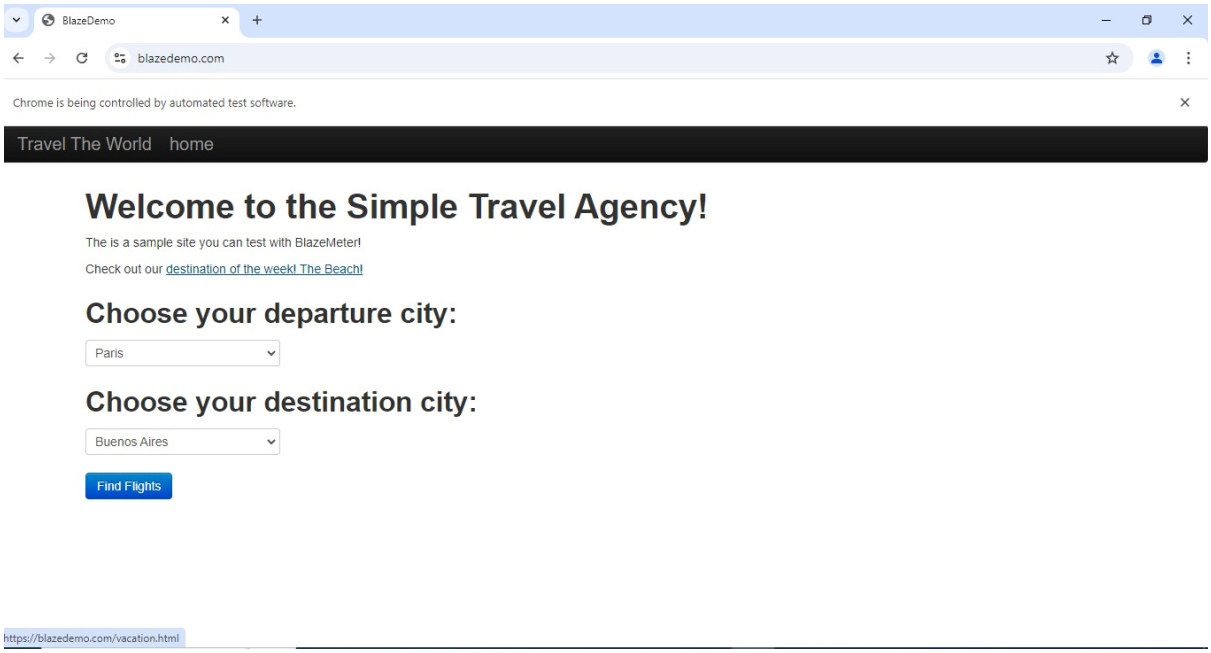
Practical 6

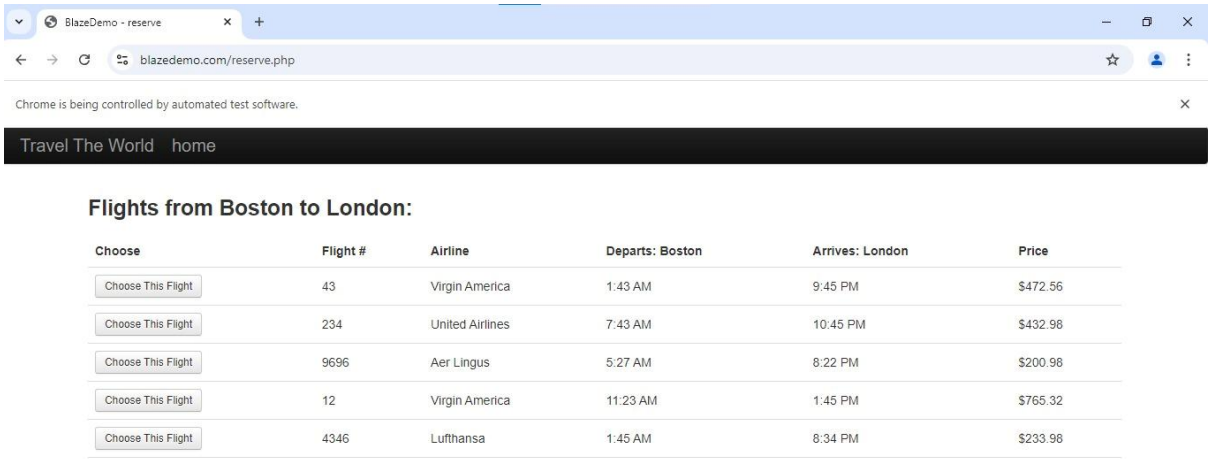
Aim: Select Value from DropDown using Selenium Webdriver.

CODE:-

```
package lamdaDemo;
import org.openqa.selenium.WebDriver.Options;
import org.openqa.selenium.WebDriver.Window;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.support.ui.Select;
public class dropdown {
	public static void main(String args[]) throws InterruptedException {
		System.setProperty("webdriver.chrome.driver", "./drivers/chromedriver.exe");
		ChromeDriver driver = new ChromeDriver();
		Options manage = driver.manage();
		Window = manage.window();
		window.maximize();
		driver.get("https://blazedemo.com/");
		Thread.sleep(1000);
		Select departure = new Select(driver.findElementByName("fromPort"));
		Select destination = new Select(driver.findElementByName("toPort"));
		departure.selectByVisibleText("Boston");
		destination.selectByVisibleText("London");
		Thread.sleep(2000);
		driver.findElementByCssSelector("body > div.container > form > div >
input").click();
		Thread.sleep(2000);
		driver.close();   }}
```

OUTPUT:-


BlazeDemo
blazedemo.com
Chrome is being controlled by automated test software.
Travel The World home
Welcome to the Simple Travel Agency!
The is a sample site you can test with BlazeMeter!
Check out our destination of the week! The Beach!
Choose your departure city:
Paris
Choose your destination city:
Buenos Aires
Find Flights
https://blazedemo.com/vacation.html


BlazeDemo - reserve
blazedemo.com/reserve.php
Chrome is being controlled by automated test software.
Travel The World home
Flights from Boston to London:
Choose
Flight #
Airline
Departs: Boston
Arrives: London
Price
Choose This Flight
43
Virgin America
1:43 AM
9:45 PM
$472.56
Choose This Flight
234
United Airlines
7:43 AM
10:45 PM
$432.98
Choose This Flight
9696
Aer Lingus
5:27 AM
8:22 PM
$200.98
Choose This Flight
12
Virgin America
11:23 AM
1:45 PM
$765.32
Choose This Flight
4346
Lufthansa
1:45 AM
8:34 PM
$233.98

Practical 7

Aim: Demonstrate CheckBox and Radio Button in Selenium WebDriver and Testng.

CODE:-

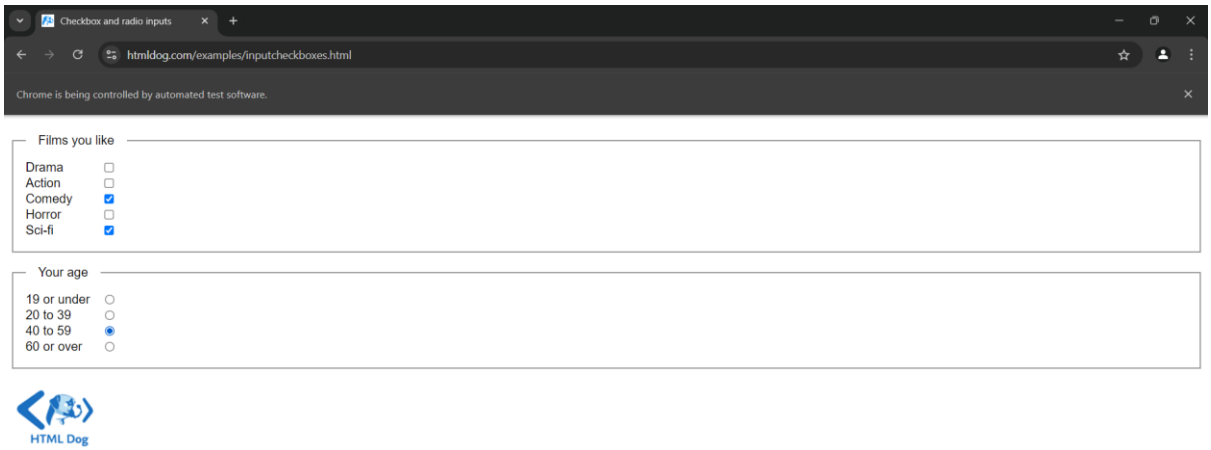
```
package demoselenium;
//import org.testng.annotations.Test;
//import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebDriver.Options;
import org.openqa.selenium.WebDriver.Window;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
public class testdemo {
	public static void main(String args[]) throws InterruptedException {
		System.setProperty("webdriver.chrome.driver", "./drivers/chromedriver.exe");
		ChromeDriver driver = new ChromeDriver();
		Options manage = driver.manage();
		Window = manage.window();
		window.maximize();
		driver.get("https://www.htmldog.com/examples/inputcheckboxes.html");
		WebElement drama = driver.findElementByName("drama");
		WebElement action = driver.findElementByName("action");
		WebElement comedy = driver.findElementByName("comedy");
		WebElement horror = driver.findElementByName("horror");
		WebElement scifi = driver.findElementByName("scifi");
		comedy.click();
		scifi.click();
		WebElement lt20 = driver.findElementById("lt20");
		WebElement gt20tolt40 = driver.findElementById("20to39");
		WebElement gt40tolt59 = driver.findElementById("40to59");
		WebElement gt59 = driver.findElementById("gt59");
```

gt40tolt59.click();

Thread.*sleep*(3000);

driver.close(); }}

OUTPUT:-



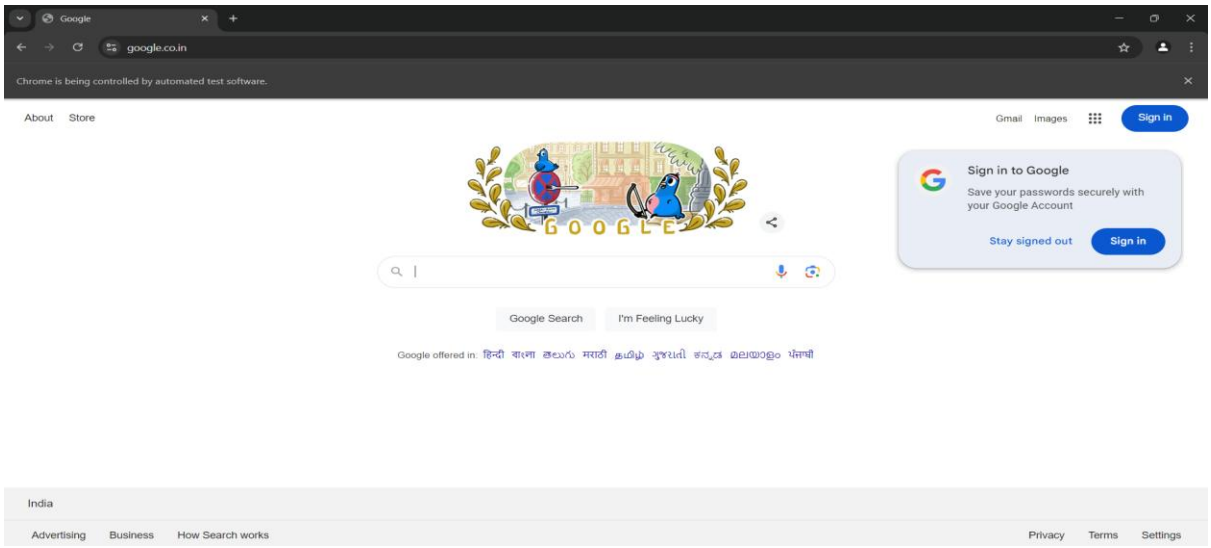
Practical 8

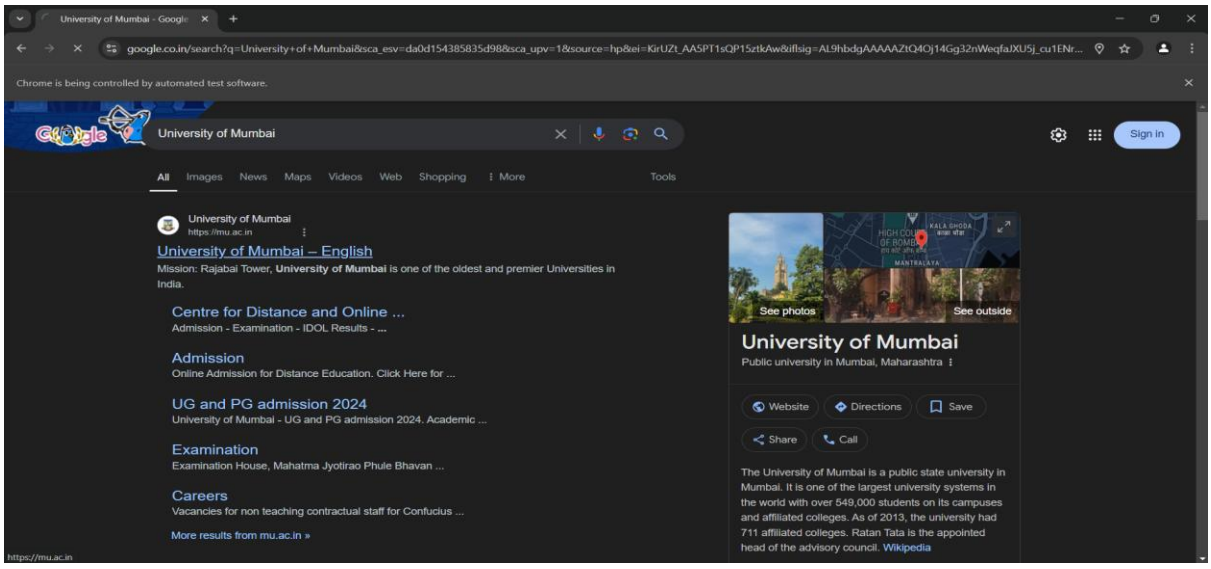
Aim: Implementing Selenium WebDriver – find element command, Locator(Class name, css selector, Xpath)

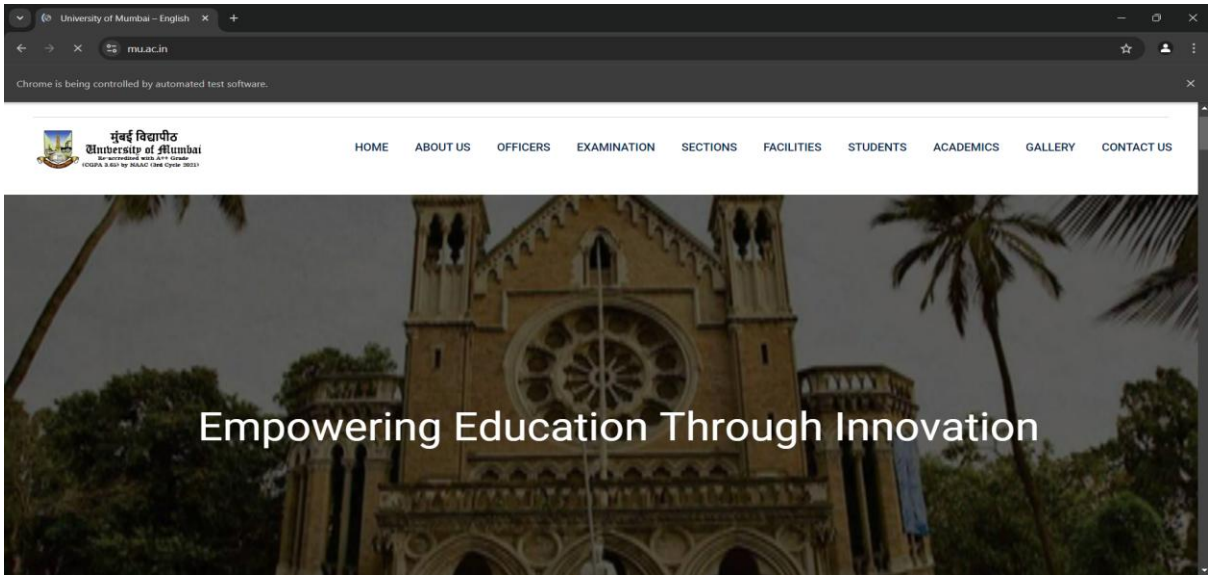
CODE:-

```
package demoselenium;

//import org.testng.annotations.Test;

//import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebDriver.Options;

import org.openqa.selenium.WebDriver.Window;

//import org.openqa.selenium.WebElement;

import org.openqa.selenium.chrome.ChromeDriver;

public class testdemo {

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

		System.setProperty("webdriver.chrome.driver", "./drivers/chromedriver.exe");

		ChromeDriver driver = new ChromeDriver();

		Options manage = driver.manage();

		Window = manage.window();

		window.maximize();

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

		//driver.findElementByClassName("gLFyf").sendKeys("Mumbai University");

		//driver.findElementByCssSelector("#APjFqb").sendKeys("Mumbai University");

		driver.findElementByXPath("//*[@id=\"APjFqb\"]").sendKeys("University of Mumbai");

driver.findElementByCssSelector("body > div.L3eUgb > div.o3j99.ikrT4e.om7nvf > form > div:nth-child(1) > div.A8SBwf > div.FPdoLc.lJ9FBc > center > input.gNO89b").click();

	driver.findElementByXPath("//*[@id=\"rso\"]/div[1]/div/div/div/div/div/div/div/div[1]/div/span/a").click();

Thread.sleep(2000);

driver.close();}

}
```

OUTPUT:-


Google
google.co.in
Chrome is being controlled by automated test software.
About
Store
Gmail
Images
Sign in
GOOGLE
Sign in to Google
Save your passwords securely with your Google Account
Stay signed out
Sign in
Google Search
I'm Feeling Lucky
Google offered in: हिन्दी বাংলা తెలుగు मराठी தமிழ் ગુજરાતી ಕನ್ನಡ മലയാളം ਪੰਜਾਬੀ
India
Advertising
Business
How Search works
Privacy
Terms
Settings


University of Mumbai - Google
google.co.in/search?q=University+of+Mumbai&sca_esv=da0d154385835d98&sca_upv=1&source=hp&ei=KirUZt_AA5PT1sQP15ztkAw&iflsig=AL9hbdgAAAAAZtQ4Oj14Gg32nWeqfaJXU5j_cu1ENr...
Chrome is being controlled by automated test software.
University of Mumbai
Sign in
All
Images
News
Maps
Videos
Web
Shopping
More
Tools
University of Mumbai
https://mu.ac.in
University of Mumbai – English
Mission: Rajabai Tower, University of Mumbai is one of the oldest and premier Universities in India.
Centre for Distance and Online ...
Admission - Examination - IDOL Results - ...
Admission
Online Admission for Distance Education. Click Here for ...
UG and PG admission 2024
University of Mumbai - UG and PG admission 2024. Academic ...
Examination
Examination House, Mahatma Jyotirao Phule Bhavan ...
Careers
Vacancies for non teaching contractual staff for Confucius ...
More results from mu.ac.in »
https://mu.ac.in
See photos
See outside
University of Mumbai
Public university in Mumbai, Maharashtra
Website
Directions
Save
Share
Call
The University of Mumbai is a public state university in Mumbai. It is one of the largest university systems in the world with over 549,000 students on its campuses and affiliated colleges. As of 2013, the university had 711 affiliated colleges. Ratan Tata is the appointed head of the advisory council. Wikipedia


University of Mumbai – English
mu.ac.in
Chrome is being controlled by automated test software.
मुंबई विद्यापीठ
University of Mumbai
HOME
ABOUT US
OFFICERS
EXAMINATION
SECTIONS
FACILITIES
STUDENTS
ACADEMICS
GALLERY
CONTACT US
Empowering Education Through Innovation

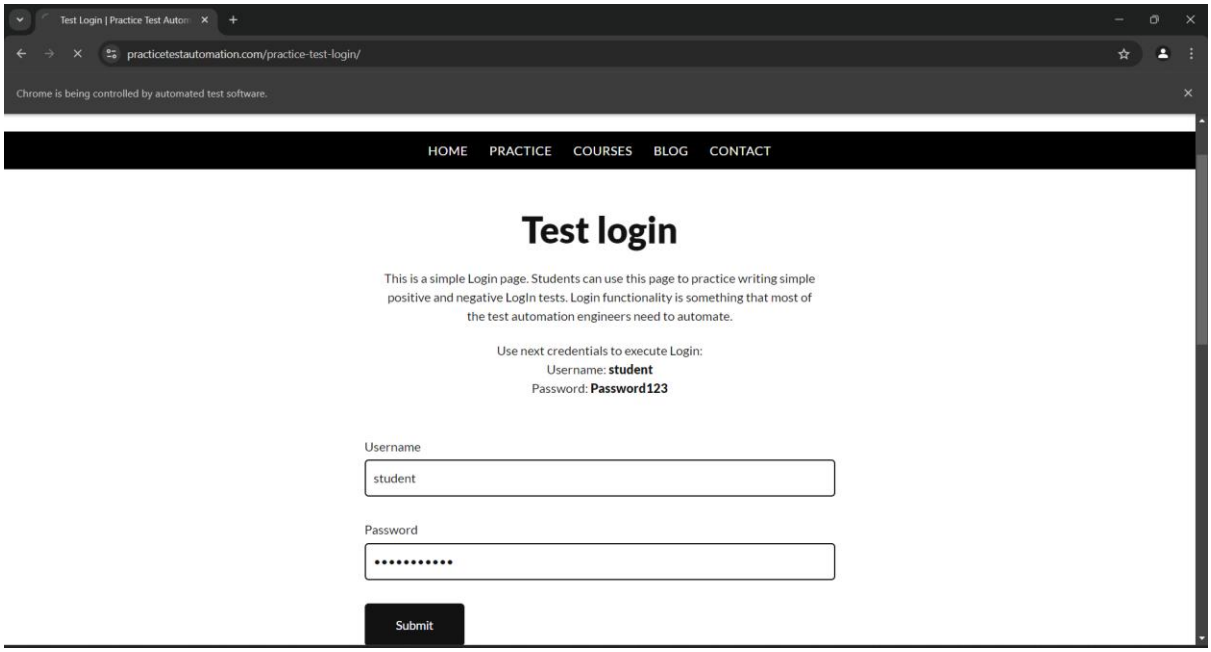
Practical 9

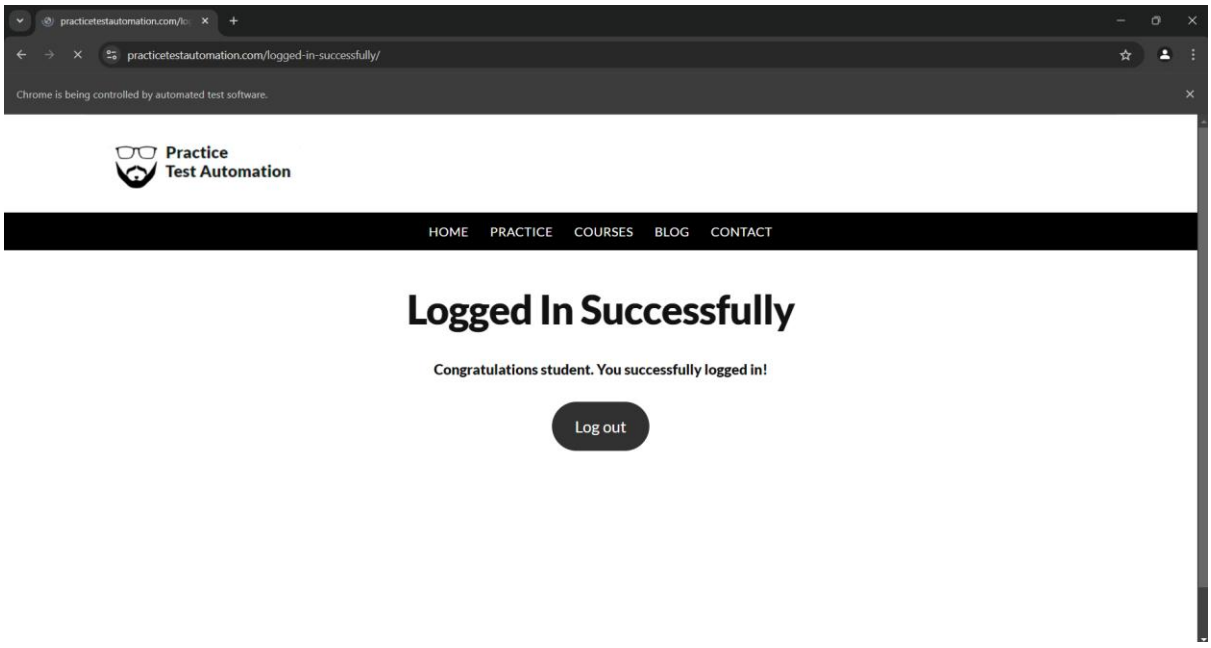
Aim: Demonstrate login form in Selenium WebDriver and Testng.

CODE:-

```
package demoselenium;
//import org.testng.annotations.Test;
//import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebDriver.Options;
import org.openqa.selenium.WebDriver.Window;
//import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;

public class testdemo {

	public static void main(String args[]) throws InterruptedException {
		System.setProperty("webdriver.chrome.driver", "./drivers/chromedriver.exe");
		ChromeDriver driver = new ChromeDriver();
		Options manage = driver.manage();
		Window = manage.window();
		window.maximize();
		driver.get("https://practicetestautomation.com/practice-test-login/");
		driver.findElementByName("username").sendKeys("student");
		driver.findElementByName("password").sendKeys("Password123");
		driver.findElementById("submit").click();
		Thread.sleep(2000);
		driver.close();
	}}
```

OUTPUT:-


Test Login | Practice Test Autom
practicetestautomation.com/practice-test-login/
Chrome is being controlled by automated test software.
HOME PRACTICE COURSES BLOG CONTACT
Test login
This is a simple Login page. Students can use this page to practice writing simple positive and negative LogIn tests. Login functionality is something that most of the test automation engineers need to automate.
Use next credentials to execute Login:
Username: student
Password: Password123
Username
student
Password
Submit


practicetestautomation.com/lo
practicetestautomation.com/logged-in-successfully/
Chrome is being controlled by automated test software.
Practice
Test Automation
HOME PRACTICE COURSES BLOG CONTACT
Logged In Successfully
Congratulations student. You successfully logged in!
Log out

Practical 10

Aim: Preparing a Report on test case for given scenario

A]

Aim: Functional Testing using Boundary Value Analysis

Scenario: A program takes the value of age from 21 to 65. Design test cases using boundary value analysis.

BOUNDARY VALUE TEST CASE		
INVALID TEST CASE (Min Value - 1)	VALID TEST CASE (Min, +Min, Max, -Max)	INVALID TEST CASE (Max Value + 1)
20	21, 22, 65, 64	66

Test Case Scenario

1. Input: - Enter the value of age as 20(21 - 1)
   Output: - Invalid

2. Input: - Enter the value of age as 21
   Output: - Valid

3. Input: - Enter the value of age as 22(21 + 1)
   Output: - Valid

4. Input: - Enter the value of age as 65
   Output: - Valid

5. Input: - Enter the value of age as 64(65 - 1)
   Output: - Valid

6. Input: - Enter the value of age as 66(65 + 1)
   Output: - Invalid

B]

Aim: Functional Testing using Equivalence Partitioning

Scenario: 3% rate of interest is given if the balance in the account is in the range of $0 to $100, 5% rate of interest is given if the balance in the account is in the range of $100 to $1000, and 7% rate of interest is given if the balance in the account is $1000 and above, we would initially identify three valid equivalence partitions and one invalid partition as shown below.

Partition 1: balance 0-100

Valid Inputs - 0-100- >=0 And <=100

Invalid Input-<0, $,#,@ A-Z

Partition 2: balance 100-1000

Valid Input - 100-1000 (>100 And <=1000)

Invalid Input- $,#,@ A-Z

Partition 3: balance >1000

Valid Input- >1000

Invalid Input- $,#,@ A-Z

Invalid Valid partitions Valid Partition

<0 0-100 100-1000 >1000

3% Interest 5% Interest 7% Interest

Test Case

Test case id Test input Expected

Account Balance Output

B001 -90 Invalid input

B002 50 3% Interest

B003 900 5% Interest

B004 2500 7% Interest

B005 A Invalid input

B006 $ Invalid input

C]

Aim: Functional Testing using Decision Table

Consider test cases based on decision table for a _Login' Page Functionality.

Business Rules:

1. On entering correct combination of ID & Password, user should be able to login successfully.

2. User is not allowed to login when any or both of the ID & Password are incorrect /blank. In such cases, it should show _Invalid

Credentials' message.

We created the following combinations of Conditions, Actions and the respective rules in the decision table.

	Rule 1	Rule 2	Rule 3	Rule 4	Rule 5	Rule 6	Rule 7	Rule 8	Rule 9
Conditions	TC01	TC02	TC03	TC04	TC05	TC06	TC07	TC08	TC09
UserID	Blank	Valid	Invalid	Blank	Blank	Invalid	Valid	Invalid	Valid
Password	Blank	Blank	Blank	Valid	Invalid	Valid	Invalid	Invalid	Valid
Actions									
Login Successfully									Execute
Error Showing 'Invalid Credentials'	Execute	Execute	Execute	Execute	Execute	Execute	Execute	Execute	

In the above table, there are

1. conditions – UserID, Password

2. Actions – Login Successfully, Error showing _Invalid Credentials‘ and

3. Options — Blank, Valid, Invalid.

So, the total number of test cases are as follows:

Options $^{\text{conditions}}$ i.e $3^2 = 9$ Test cases

All test cases are not valid and significant some we need to optimise the test cases

Rules 1, 2, 3, 4, and 5 cover the same action Item —Invalid Credentials‖ with options Blank and Invalid. Hence, we can consider any one of these test cases TC01 OR TC02 OR TC03 OR TC04 OR TC05

1. Rules 6,7, and 8 cover the same action Item —Invalid Credentials‖ with options Valid and Invalid. Hence, we can consider any of these test cases TC06 OR TC07 OR TC08

2. Rule 9 covers the action item —Login Successfully‖ with all valid options. Hence, we should consider the test case TC09.

Condensed Decision Table as shown below:

	Rule 2	Rule 8	Rule 9
Conditions	TC02	TC08	TC09
UserID	Valid	Invalid	Valid
Password	Blank	Invalid	Valid
Actions			
Login Successfully			Execute
Error Showing 'Invalid Credentials'	Execute	Execute	

D]

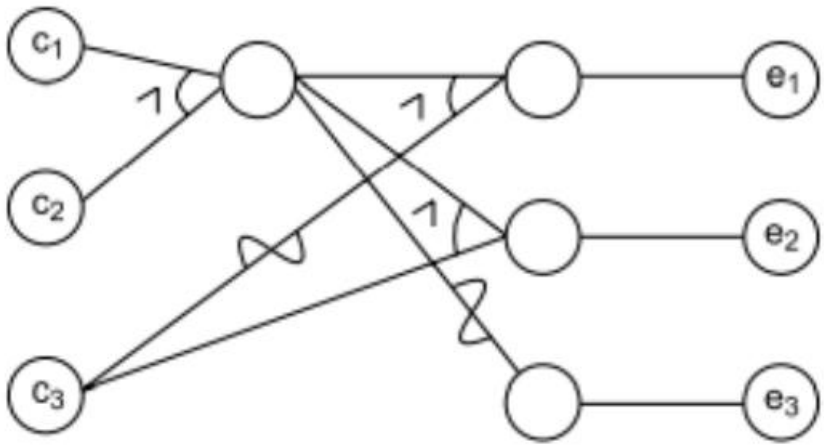
Aim: Functional Testing using Cause Effect Graph

A tourist of age greater than 21 years and having a clean driving record is supplied a rental car. A premium amount is also charged if the tourist is on business, otherwise it is not charged. If the tourist is less than 21 year old, or does not have a clean driving record, the system will display

the following message: —Car cannot be supplied‖ Draw the cause-effect graph and generate test cases

Solution: The causes are c1 : Age is over 21 c2 : Driving record is clean c3 : Tourist is on business and effects are e1 : Supply a rental car without premium charge. e2 : Supply a rental car with premium charge e3 : Car cannot be supplied .The Cause Effect Graph and test cases based on that are as follows :

Cause Effect Graph:



Decision Table on Rental Car Problem:

Conditions	1	2	3	4
c_1 : Over 21 ?	F	T	T	T
c_2 : Driving record clean ?	-	F	T	T
c_3 : On Business ?	-	-	F	T
e_1 : Supply a rental car without premium charge			X	
e_2 : Supply a rental car with premium charge				X
e_3 : Car cannot be supplied	X	X		

Test Cases:

Test Case	Age	Driving_record_clean	On_business	Expected Output
1.	20	Yes	Yes	Car cannot be supplied
2.	26	No	Yes	Car cannot be supplied
3.	62	Yes	No	Supply a rental car without premium charge
4.	62	Yes	Yes	Supply a rental car with premium charge.