Automation Introduction:

Definition of automation

Testing an application features with the help of automation tool known as automation testing.

Tools

1)Selenium

2)QTP

3)Sahi

4)Sahi pro

5)Protector

6)Appium

7)Selendroid

Disadvantages of manual 1)Require

more resources.

2)It is time consuming.

3)compatibility testing (cross browser testing) is very difficult in manual testing.

because if we have to check build on different browser then if one browser takes 10 min

then 6 browser take 60 min.

4)Test cycle duration will be increased.

5)More human efforts are required.

6)Less accuracy.

Advantages of automation:

1)Require less resources

QA

2)It required less time consuming to run the automation script.

3)compatibility testing (cross browser testing) is easy in automation testing.

4)Test cycle duration will be decreased.

5)More accuracy.

6)Regression testing is easy.

Why we choose selenium

1)It is open source.

2)Supported by multiple languages.

1)Java 2) Python,3) C#,4) Pearl.

Disadvantages of automation

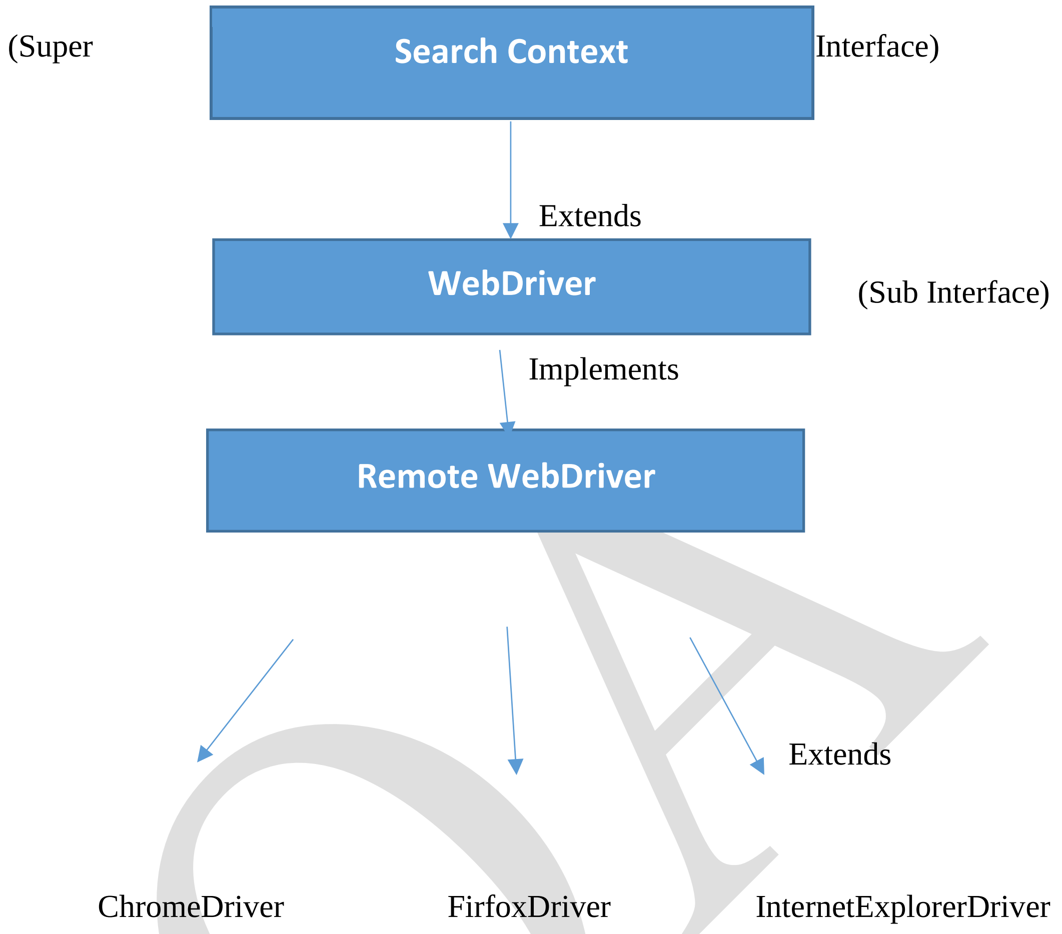
1)we can automate only web based application.

2)we cannot test captcha or barcode.

3)We can perform regression testing but cannot perform ad-hoc testing.

Selenium Architecture

QA



1) Search context is super interface which is extended by WebDriver which is sub interface of Search Context
2) Search context contains incomplete methods (i.e. abstract methods) these incomplete methods are extended by WebDriver so WebDriver contain its own incomplete method as well as incomplete method of search context.
3) Remote WebDriver is implement class which provide definition to the incomplete methods of search context and WebDriver.
4) Remote WebDriver:

   i)Remote WebDriver class is extended by different browsers like: ChromeDriver, InternetExplorerDriver, FirefoxDriver.

   ii)We write a script for a browser but we can run that script for multiple browser.

iii)But to run the test scripts we need functions of WebDriver so we do up casting. Ex: We have to run Script for google chrome, then we have to create object of ChromeDriver with reference to WebDriver.

```
WebDriver driver = new ChromeDriver ();
```

5)WebDriver:

WebDriver is an interface which perform action on browser like: open , close ,maximize , get , navigate.

WebDriver vs Web Elements

WebDriver	Web Elements
1.It is an interface which perform action on browser.	1.It is an interface which perform action on Elements of browser
2.Ex:open,close,get,navigate,maximize.	2.Ex:dropdown,radiobutton,checkbox,table.

QA

Selenium WebDriver methods:

Methods	Meaning
System.*setProperty* ("webdriver.chrome.driver","E:/Soft/chrome_driver2/chromedriver.exe");	System is class , setProperty is static method,” webdriver.chrome.driver”-this is name of chrome driver,” E:/Soft/chrome_driver2/chromedriver.exe”-path of chrome driver
WebDriver driver=**new** ChromeDriver();	WebDriver-is a Interface, driverobject, ChromeDriver-Interface ,we just up casted ChromeDriver in WebDriver
get("https://www.google.com/");	Open the specific link
driver. manage().window().maximize();	Maximize the browser window
Thread.*sleep*(1000);	Thread-Class available in lang package,*sleep-Static method of Thread class*,(value)-time in millisecond;
navigate().to	Navigate selenium control on another link.
driver.navigate().back();	Back to one time
driver.navigate().refresh();	Refresh the web page
driver.navigate().forward();	For click on forward arrow
getTitle	For getting the title of page
driver.getCurrentUrl();	For getting the url of current page
getPageSource()	Syntax: getPageSource() Example: driver.getPageSource();

QA

	Purpose: Get the source of the currently loaded page. If the page has been modified after loading (for example, by JavaScript) there is no guarantee that the returned text is that of the modified page. Returns: The source of the current page
close()	Syntax: void close() Example: driver.close(); Purpose: Close the current window, if there are multiple windows, it will close the current window which is active and quits the browser if it's the last window opened currently.

QA

quit()	syntax: void quit() Example: driver.quit(); Purpose: Quits this driver instance, closing every associated window which is opened.

QA

Change position of Browser:

```
package methods_of_WebDriver;

import org.openqa.selenium.Point; import
org.openqa.selenium.WebDriver; import
org.openqa.selenium.chrome.ChromeDriver;

public class ChangePositionOfBrowser {
public static void main(String[] args) throws InterruptedException {
System.setProperty("webdriver.chrome.driver","E:/Soft/chrome_driver2/chromedriver.exe");
WebDriver driver=new ChromeDriver(); driver.get("https://www.google.com/");
Thread.sleep(1000);
Point p = new Point(100,100); driver.manage().window().setPosition(p);
}

}
```

Change Size of Browser:

```
package methods_of_WebDriver; import
org.openqa.selenium.Dimension; import
org.openqa.selenium.WebDriver; import
org.openqa.selenium.chrome.ChromeDriver;

public class ChangeSizeOfBrowser {
public static void main(String[] args) throws InterruptedException {
System.setProperty("webdriver.chrome.driver","E:/Soft/chrome_driver2/chromedriver.exe");
WebDriver driver=new ChromeDriver(); driver.get("https://www.google.com/");
Thread.sleep(1000);
Dimension d = new Dimension(300,700); driver.manage().window().setSize(d);

}

}
```

Locators

Types of Locator

1)tagName()

2)id()

3)name()

4)className()

5)linkText()

6)partialLinkText

7)xpath

----

1)tagName():

tagName is method of By class,in this method we need to put tagName of perticular element.

Program:

```
package selenium; import org.openqa.selenium.By;

import org.openqa.selenium.Dimension; import

org.openqa.selenium.Point; import

org.openqa.selenium.WebDriver; import

org.openqa.selenium.chrome.ChromeDriver;
```

QA

```
public class Demo {


public static void main(String[] args) throws InterruptedException {
System.setProperty("webdriver.chrome.driver","E:/Soft/chromedriver.exe"); WebDriver
driver=new ChromeDriver(); driver.get("https://www.google.com/"); Thread.sleep(2000);
driver.manage().window().maximize(); Thread.sleep(2000);
driver.findElement(By.tagName("a")).click();
}

}
```

----

2)id() driver.findElement(By.id("email")).sendKeys("836543");

----

3)name()

driver.findElement(By.name("email")).sendKeys("836543");

----

4)className() driver.findElement(By.className("gb_f")).click();

----

5)linkText() driver.findElement(By.linkText("Images")).click();

----

6)partialLinkText()

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

----

7)xpath

1)X-path by attribute

2)X-path by text

3)x-path by contains

4)x-path by index

5)Absolute x path

6)Relative x path

1)X-path by attribute

Syntax:

```
driver.findElement(By.xpath("//tagname[@attributename='attributevalue']"));
```

Ex:

```
driver.findElement(By.xpath("//a[@class='gb_g']"));
```

--- **2)X-path**

by text

Syntax:

```
driver.findElement(By.xpath("//tagname[text()='textname']"));
```

Ex;

Syntax:

```
driver.findElement(By.xpath("//a[text()='Gmail']"));
```

--- **3)x-path**

by contains

Syntax:

```
driver.findElement(By.xpath("//tagname[contains(text(),'textname')]"));
```

Ex: driver.findElement(By.xpath("//a[contains(text(),'Gm')]"));

QA

--- **4)x-path**

by index

Syntax:

driver.findElement(By.xpath("//tagname[@attributename='attributevalue'][index]"));

Ex: driver.findElement(By.xpath("//a[@class='gb_g'][2]"));

-

5)Absolute x path :

1)In this focus is on the html tag

/html/head/body/div[4]/input[3]

2)Each tag are separated by / (slash)

3)It is use to navigate from root of parent to immediate child

Disadvantages

1)X path is too long and time consuming

2)Developing html tree diagram is difficult

----

6)Relative x path :

1)In this focus is on the html tag

//body//div[4]//input[3]

2)Each tag are seperated by // (slash)

3)It is use to navigate from any parent to immediate child

Disadvantages

1)X path is too long and time consuming

2)Developing html tree diagram is difficult

QA

QA

Function of Web Element

1)getText()

1)To get the text on browser we use getText() function.

```
package seleniumPrograms; import

org.openqa.selenium.By; import

org.openqa.selenium.Dimension; import

org.openqa.selenium.Point; import

org.openqa.selenium.WebDriver; import

org.openqa.selenium.chrome.ChromeDriver;


public class Demo {
```

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

System.setProperty("webdriver.chrome.driver", "D:/aaa/Chrome driver/chromedriver.exe");

WebDriver driver=new ChromeDriver(); driver.get("https://www.google.co.in/");

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

Thread.sleep(2000);

String abc = driver.findElement(By.linkText("Gmail")).getText();

System.out.println(abc);

}


}
```

----

2)isEnabled()

1)If the Elements are Enabled then return true otherwise false

2)we give return type of the isEnabled() is Boolean.

3)we use "isEnabled" function for check The web element avilable on web page is enable or disable.

Example:

```
driver.get("https://www.google.co.in/"); driver.manage().window().maximize();

Thread.sleep(2000);


Boolean abc = driver.findElement(By.linkText("Gmail")).isEnabled();

System.out.println(abc);
```

3)isSelected()

QA

Use to check whether the radio button or checkbox is selected or not.

```
package seleniumPrograms; import

org.openqa.selenium.By; import

org.openqa.selenium.Dimension; import

org.openqa.selenium.Point; import

org.openqa.selenium.WebDriver; import

org.openqa.selenium.chrome.ChromeDriver;


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

InterruptedException  {

System.setProperty("webdriver.chrome.driver", "D:/aaa/Chrome

driver/chromedriver.exe"); WebDriver driver=new ChromeDriver();

driver.get("https://www.seleniumeasy.com/test/basic-checkbox-

demo.html"); driver.manage().window().maximize();

Thread.sleep(2000);

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

Thread.sleep(2000);

Boolean abc = driver.findElement(By.id("isAgeSelected")).isSelected();

System.out.println(abc);



                              }


}
```

4)isDisplayed if component /Element is actually present or not is get checked with the help of function isDisplayed.

```
driver.get("https://www.google.co.in/"); driver.manage().window().maximize();

Thread.sleep(2000);

Boolean abc = driver.findElement(By.className("gb_f")).isDisplayed();

System.out.println(abc);
```

List Box:

List box is set of options

```
1)WebElement a = driver.findElement(By.xpath("xpathexp"));

2)Select  s = new Select(a);
```

Folllowing methods are used for select the option from listbox

1.	s.selectByIndex();

2.	s.selectByVisibleText();	3. s.selectByValue();

1.s.selectByIndex(); Example:

```
package ListBox; 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 ListBoxDemo {
```

QA

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

System.setProperty("webdriver.chrome.driver", "E:/Soft/chromedriver.exe"); WebDriver
driver =new ChromeDriver();


driver.manage().window().maximize(); driver.get("https://www.seleniumeasy.com/test/basic-

select-dropdown-demo.html");

WebElement a  = driver.findElement(By.name("States"));


Select s = new Select(a);

s.selectByIndex(1);

            }

            }
```

2.s.selectByVisibleText():

```
package ListBox; 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 ListBoxDemo { public static void main(String[] args)

throws InterruptedException {
```

QA

```
System.setProperty("webdriver.chrome.driver", "E:/Soft/chromedriver.exe"); WebDriver
driver =new ChromeDriver(); driver.manage().window().maximize();
driver.get("https://www.seleniumeasy.com/test/basic-select-dropdown-demo.html");
WebElement a  = driver.findElement(By.name("States"));

Select s = new Select(a);
s.selectByVisibleText("California");

                    }
          }
```

3. s.selectByValue():

Program:

```
package ListBox; 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 ListBoxDemo { public static void main(String[] args)
throws InterruptedException {
System.setProperty("webdriver.chrome.driver",
"E:/Soft/chromedriver.exe");
```

QA

```
WebDriver driver =new ChromeDriver();


driver.manage().window().maximize(); driver.get("https://www.seleniumeasy.com/test/basic-select-dropdown-demo.html");
WebElement a  = driver.findElement(By.name("States"));


Select s = new Select(a);
s.selectByValue("California");
                    }
          }
```

Folllowing methods are used for deselect the option from listbox

1. s.deselectByIndex();

2. s.deselectByVisibleText();

3. s.deselectByValue();

--- if we want to deselect the selected options then we use following methods of select class

```
s.deselectByIndex();
   s.deselectByVisibleText();
   s.deselectByValue();
   s.deselectAll();
```

getFirstSelectedOption()

Syntax:

QA

```
WebElement a = driver.findElement(By.xpath("xpathexp")); Select
s = new Select(a);
s.selectByIndex(0)
WebElement a = s.getFirstSelectedOption();
System.out.println(a.getText());
```

Program:

```
package ListBox; 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 ListBoxDemo { public static void main(String[] args)
throws InterruptedException {
System.setProperty("webdriver.chrome.driver", "E:/Soft/chromedriver.exe");
WebDriver driver =new ChromeDriver(); driver.manage().window().maximize();
driver.get("https://www.seleniumeasy.com/test/basic-select-dropdown-demo.html");
WebElement a  = driver.findElement(By.name("States"));

Select s = new Select(a);
s.selectByIndex(0);
WebElement b = s.getFirstSelectedOption();
String c = b.getText();
System.out.println(c);
```

QA

```
                }

        }
```

isMultiple()

Ex:

```
WebElement a  = driver.findElement(By.name("States")); Select
s = new Select(a);
s.selectByIndex(0);
s.selectByIndex(1);
s.selectByIndex(2);
Boolean sm = s.isMultiple();
System.out.println(sm);
```

Program:

```
package ListBox; 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;
```

QA

```
public class ListBoxDemo { public static void main(String[] args)

throws InterruptedException {

System.setProperty("webdriver.chrome.driver", "E:/Soft/chromedriver.exe");

WebDriver driver =new ChromeDriver(); driver.manage().window().maximize();

driver.get("https://www.seleniumeasy.com/test/basic-select-dropdown-demo.html");

WebElement a  = driver.findElement(By.name("States"));


Select s = new Select(a);

s.selectByIndex(0);

s.selectByIndex(1);

s.selectByIndex(2);


 Boolean sm = s.isMultiple();

System.out.println(sm);
                    }

}
```

getOptions()

```
Ex:package selenium; 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;
```

QA

```
public class ListBoxDemo { public static void main(String[] args)

throws InterruptedException {

System.setProperty("webdriver.chrome.driver", "E:/Soft/chrome_driver2/chromedriver.exe");

WebDriver driver =new ChromeDriver(); driver.manage().window().maximize();

driver.get("https://www.seleniumeasy.com/test/basic-select-dropdown-demo.html");

WebElement a  = driver.findElement(By.name("States"));


Select s = new Select(a);

List<WebElement> b = s.getOptions(); int

c =b.size();

System.out.println(c);


for(int i=0;i<=c-1;i++) {

        String d =b.get(i).getText();

        System.out.println(d);

        Thread.sleep(2000);

                }

}


}
```

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

Output:

```
Starting          ChromeDriver          90.0.4430.24
(4c6d850f087da467d926e8eddb76550aed655991refs/branch-heads/4430@{#429}) on port
20015

Only local connections are allowed.
```

QA

```
Please see https://chromedriver.chromium.org/security-considerations for suggestions on
keeping ChromeDriver safe.

ChromeDriver was started successfully.

[1626879039.366][WARNING]: This version of ChromeDriver has not been tested with
Chrome version 91.

Jul 21, 2021 8:20:39 PM org.openqa.selenium.remote.ProtocolHandshake createSession

INFO: Detected dialect: W3C

8

California

Florida
New Jersey

New York

Ohio

Texas

Pennsylvania

Washington


------------------------------------------------------------------------------------------------
```

QA

QA

Screenshot Code:

```
package selenium; import java.io.File; import
java.io.IOException; import
org.apache.commons.io.FileUtils; import
org.openqa.selenium.OutputType; import
org.openqa.selenium.TakesScreenshot; import
org.openqa.selenium.WebDriver; import
org.openqa.selenium.chrome.ChromeDriver;

public class ScreenshotDemo { public static void main(String[] args) throws
InterruptedException, IOException {
System.setProperty("webdriver.chrome.driver","E:/Soft/chrome_driver2/chromedriver.exe");
WebDriver driver=new ChromeDriver();
driver.get("https://www.seleniumeasy.com/test/basic-select-dropdown-demo.html");
driver.manage().window().maximize();
Thread.sleep(2000);
File s =((TakesScreenshot)driver).getScreenshotAs(OutputType.FILE);
File d = new File("C:/Users/Admin/Desktop/Screenshot/pqr.bmp");
FileUtils.copyFile(s, d);


        }}
```

ExcelSheet:

```
package ExcelSheet; import java.io.FileInputStream;
import java.io.IOException; import
org.apache.poi.EncryptedDocumentException; import
org.apache.poi.ss.usermodel.Sheet; import
org.apache.poi.ss.usermodel.WorkbookFactory;

public class Excel {
public static void main(String[] args) throws EncryptedDocumentException, IOException,
InterruptedException {

                //Open the Excel sheet

FileInputStream excel = new FileInputStream("C:/Users/Admin/Desktop/abc.xlsx");
Sheet a= WorkbookFactory.create(excel).getSheet("Sheet1");
String b= a.getRow(0).getCell(0).getStringCellValue();
String c=a.getRow(1).getCell(0).getStringCellValue();
String d=a.getRow(2).getCell(0).getStringCellValue();

Thread.sleep(2000);
System.out.println(b);
System.out.println(c);
System.out.println(d);

        }

}
```

iframe:

Displaying web page as part of another web page is known as iframe .

-For switching the selenium control from main window to frame the method used called switchTo().

-but we have to switch on frame so for that we use method frame().

-we can give three parameter of the method frame() i.e:-1)name,2)id,3)index

Syntax:-

```
driver.switchTo().frame(Name);
driver.switchTo().frame(id); driver.switchTo().frame(index);
```

Switch the selenium control from main window to first frame

Program:-

```
System.setProperty("webdriver.chrome.driver", "E:/Soft/chromedriver.exe");
WebDriver driver =new ChromeDriver();
driver.manage().window().maximize();
driver.get("https://www.selenium.dev/selenium/docs/api/java/index.html?org/openqa/selenium/package-summary.html");

Thread.sleep(2000);

driver.switchTo().frame(0);

Thread.sleep(2000);

driver.findElement(By.linkText("org.openqa.selenium.cli")).click();
```

Switch the selenium control from one frame to another frame

Program:

QA

```
System.setProperty("webdriver.chrome.driver", "E:/Soft/chromedriver.exe"); WebDriver
driver =new ChromeDriver(); driver.manage().window().maximize();
driver.get("https://www.selenium.dev/selenium/docs/api/java/index.html?org/openqa/seleni
u m/package-summary.html");
Thread.sleep(2000);

driver.switchTo().frame(0);

Thread.sleep(2000);

driver.findElement(By.linkText("org.openqa.selenium.cli")).click();

Thread.sleep(2000);

driver.switchTo().parentFrame(); //driver.switchTo().defaultContent();

Thread.sleep(1000);

driver.switchTo().frame(2);
driver.findElement(By.linkText("By.Remotable")).click();
```

Note:-For switch the selenium control from child frame to parent frame use the method i.e parentFrame() or defaultContent(); you have to use just one method between them.

Syntax:

```
1)driver.switchTo().parentFrame();
2)driver.switchTo().defaultContent();
```

Alert :

When we provide some input to the textbox and after that when we click on submit button so this data is not submitted at that time for submitting this, required some confirmation so this confirmation is come through alert popup.

For handling alert pop up need to use interface "Alert".

QA

Syntax:

```
Alert alt = driver.switchTo().alert();
alt.accept();//for click on "OK" button
alt.dismiss();//for click on "Cancel " button

String abc = alt.getText();//for get the text from alert pop up
System.out.println(abc);
```

Example:

```
package Selenium;
import org.openqa.selenium.Alert; import
org.openqa.selenium.By; import
org.openqa.selenium.WebDriver; import
org.openqa.selenium.chrome.ChromeDriver;

public class AlertPopup {
public static void main(String[] args) throws InterruptedException {
System.setProperty("webdriver.chrome.driver","D:/aaa/Chrome driver/chromedriver.exe");
WebDriver driver=new ChromeDriver();

driver.get("http://demo.guru99.com/test/delete_customer.php ");
driver.manage().window().maximize();
Thread.sleep(2000);

driver.findElement(By.name("cusid")).sendKeys("53920"); Thread.sleep(2000);
driver.findElement(By.name("submit")).click();
Thread.sleep(2000);

Alert alt = driver.switchTo().alert();
alt.accept(); //alt.dismiss();
//String abc = alt.getText();//for get the text from alert pop up
//System.out.println(abc);
```

QA

```
                    }

}
```

QA

Window popup/Windows Handling:

```
package Selenium; import java.util.Iterator; import
java.util.Set; import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver; import
org.openqa.selenium.chrome.ChromeDriver;

public class WindowPopup {

public static void main(String[] args) throws InterruptedException {
System.setProperty("webdriver.chrome.driver","D:/aaa/Chrome driver/chromedriver.exe");
WebDriver driver=new ChromeDriver(); driver.get("http://demo.guru99.com/popup.php ");
driver.manage().window().maximize();
Thread.sleep(2000);

driver.findElement(By.linkText("Click Here")).click();
Thread.sleep(2000);

Set<String> s= driver.getWindowHandles();
Iterator<String> it = s.iterator(); it.next();
String w2nd = it.next();
driver.switchTo().window(w2nd); Thread.sleep(2000);
driver.findElement(By.name("emailid")).sendKeys("gaurav.3n@gmail.com");
Thread.sleep(2000);
driver.findElement(By.name("btnLogin")).click(); Thread.sleep(2000);
driver.findElement(By.linkText("Click Here")).click();
} }
```

Actions Class:

Actions class Methods

1)moveToElement()

2)click()

3)doubleClick()

4)contextClick()

5)perform()

6)sendKeys()

7)dragAndDrop()

----

1)moveToElement():

If we want to move mouse pointer from one position to another position that time we use method moveToElement().

Program:

```
package Actions_class;

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.interactions.Actions;

public class ActionsMoveToElementDemo {

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

QA

```
System.setProperty("webdriver.chrome.driver", "D:/aaa/Chrome driver/chromedriver.exe");
WebDriver driver=new ChromeDriver();

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

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

Thread.sleep(2000);

WebElement a = driver.findElement(By.linkText("Gmail"));

Actions b = new Actions(driver);

b.moveToElement(a).perform();


}

}
```

----

2)click()

This method is used for click on particular web element

Program:

```
package Actions_class;

import org.openqa.selenium.By; import
org.openqa.selenium.Keys; import
org.openqa.selenium.WebDriver; import
org.openqa.selenium.WebElement; import
org.openqa.selenium.chrome.ChromeDriver; import
org.openqa.selenium.interactions.Actions;

public class ActionsClickDemo {

public static void main(String[] args) throws InterruptedException {
System.setProperty("webdriver.chrome.driver", "D:/aaa/Chrome driver/chromedriver.exe");
WebDriver driver=new ChromeDriver();

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

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

Thread.sleep(2000);

WebElement a = driver.findElement(By.linkText("Gmail"));
Actions b = new Actions(driver); b.click(a).build().perform();
}

}
```

----

3)doubleClick()

This method is used for doubleClick on particular web element

Program:

```
package Actions_class;

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.interactions.Actions;

public class ActionsDoubleClickME {

public static void main(String[] args) {
System.setProperty("webdriver.chrome.driver", "D:/aaa/Chrome driver/chromedriver.exe");
WebDriver driver=new ChromeDriver();
driver.get("http://www.uitestpractice.com/Students/Actions");
driver.manage().window().maximize();


WebElement doubleClick = driver.findElement(By.name("dblClick"));

Actions actions=new Actions(driver);
actions.doubleClick(doubleClick).build().perform();
}

}
```

----

4)contextClick()

This method is used for right click on particular web element

Program:

```
package Actions_class;

import org.openqa.selenium.By; import
org.openqa.selenium.WebDriver; import
org.openqa.selenium.WebElement; import
org.openqa.selenium.chrome.ChromeDriv
er; import
org.openqa.selenium.interactions.Actions;

public class ActionsContextClickDemo {

public static void main(String[] args) throws InterruptedException {
System.setProperty("webdriver.chrome.driver", "D:/aaa/Chrome driver/chromedriver.exe");
WebDriver driver=new ChromeDriver();

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

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

Thread.sleep(2000);

WebElement a = driver.findElement(By.linkText("Gmail"));

Actions b = new Actions(driver);
b.contextClick(a).build().perform();
}

}
```

----

5)perform()

This method is used for perform action on particular web element

Program:

```
System.setProperty("webdriver.chrome.driver", "D:/aaa/Chrome driver/chromedriver.exe");
WebDriver driver=new ChromeDriver();

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

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

Thread.sleep(2000);
```

QA

```
WebElement a = driver.findElement(By.linkText("Gmail"));

Actions b = new Actions(driver);
b.contextClick(a).build().perform();
```

----

6)sendKeys()

Syntax:

```
Actions_class_object . sendKeys(Keys.ARROW_UP).build().perform();
```

Two way for using sendKeys() method

```
1)b.sendKeys(Keys.ARROW_DOWN).build().perform();
```

Program:1:

```
package Selenium_Actions_Class;

import org.openqa.selenium.By; import
org.openqa.selenium.Keys; import
org.openqa.selenium.WebDriver; import
org.openqa.selenium.chrome.ChromeDriver; import
org.openqa.selenium.interactions.Actions;

public class KeysClassDemo2 {

public static void main(String[] args) throws InterruptedException {
System.setProperty("webdriver.chrome.driver", "E:/Soft/chrome_driver2/chromedriver.exe");

WebDriver driver =new ChromeDriver();

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

driver.get("http://www.amazon.in/"); driver.manage().window().maximize();
Thread.sleep(2000);

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

Thread.sleep(2000);


Actions a = new Actions(driver);

for(int i=7;i>=0;i--) {
```

QA

```
a.sendKeys(Keys.ARROW_DOWN).build().perform();
Thread.sleep(2000);
}

Thread.sleep(2000);


for(int i=0;i<=6;i++) {
a.sendKeys(Keys.ARROW_UP).build().perform();
Thread.sleep(2000);
}


}

}
```

Program :2:

```
package Actions_class;

import org.openqa.selenium.By; import
org.openqa.selenium.Keys; import
org.openqa.selenium.WebDriver; import
org.openqa.selenium.chrome.ChromeDriver; import
org.openqa.selenium.interactions.Actions;

public class KeysClassTabDemo {

public static void main(String[] args) {

System.setProperty("webdriver.chrome.driver", "D:/aaa/Chrome driver/chromedriver.exe");
WebDriver driver=new ChromeDriver(); driver.get("https://www.facebook.com/");
driver.manage().window().maximize();

driver.findElement(By.name("email")).sendKeys("8329080292");

Actions actions=new Actions(driver); actions.sendKeys(Keys.TAB).build().perform();

// driver.findElement(By.name("pass")).sendKeys("Sandip@123");


}

}
```

----

2)b.sendKeys(target, Keys.TAB).build().perform();

Program:

```
package Actions_class;

import org.openqa.selenium.Alert; import
org.openqa.selenium.By; import
org.openqa.selenium.Keys; import
org.openqa.selenium.WebDriver; import
org.openqa.selenium.WebElement; import
org.openqa.selenium.chrome.ChromeDriver; import
org.openqa.selenium.interactions.Actions;

public class KeysClassEnterDemo {

public static void main(String[] args) throws InterruptedException {
System.setProperty("webdriver.chrome.driver", "D:/aaa/Chrome driver/chromedriver.exe");
WebDriver driver=new ChromeDriver();

driver.get("http://www.uitestpractice.com/Students/Actions");
driver.manage().window().maximize();
Actions actions=new Actions(driver);
//  actions.sendKeys(Keys.END)
//  .perform();
//
//  Thread.sleep(2000);
//
//  actions.sendKeys(Keys.HOME)
//  .perform();

actions.sendKeys(driver.findElement(By.name("click")),Keys.ENTER).build().perform();
Thread.sleep(2000);



Alert alt = driver.switchTo().alert(); alt.accept();

//  driver.quit();
}

}
```

----

3)b.sendKeys(Keys.ENTER).build().perform();

Program:

QA

```
package actionsClass;

import org.openqa.selenium.Alert;
import org.openqa.selenium.By; import
org.openqa.selenium.Keys; import
org.openqa.selenium.WebDriver; import
org.openqa.selenium.chrome.ChromeDr
iver; import
org.openqa.selenium.interactions.Action
s;

public class KeysClassEnterDemo {
public static void main(String[] args) throws InterruptedException {
System.setProperty("webdriver.chrome.driver", "E:/Soft/a/chromedriver.exe");
WebDriver driver=new ChromeDriver();

driver.get("http://www.uitestpractice.com/Students/Actions");
driver.manage().window().maximize();

driver.findElement(By.name("click")).click();

Thread.sleep(2000);

Actions actions=new Actions(driver);
actions.sendKeys(Keys.ENTER).build().perform();
Thread.sleep(2000);



}
}
```

Methods of Keys Class

1)ARROW_UP

2)ARROW_DOWN

3)ARROW_LEFT

4)ARROW_RIGHT

5)ENTER

6)TAB

QA

7)dragAndDrop()

This method is used for drag one web element and drop to another position .

Program:

```
package Actions_class;

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.interactions.Actions;

public class ActionsDragAndDropDemo {

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

System.setProperty("webdriver.chrome.driver", "D:/aaa/Chrome driver/chromedriver.exe");
WebDriver driver=new ChromeDriver();

driver.get("http://www.uitestpractice.com/Students/Actions");

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

Thread.sleep(2000);

WebElement source = driver.findElement(By.id("draggable"));
WebElement destination = driver.findElement(By.id("droppable"));

Actions a = new Actions(driver);
a.dragAndDrop(source, destination).build().perform();
}

}
```

Total number of links in web page:

```
package Selenium;
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 LinkAvailableOnWebPage {

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

System.setProperty("webdriver.chrome.driver", "D:/aaa/Chrome driver/chromedriver.exe");
WebDriver driver=new ChromeDriver();
driver.get("https://www.google.co.in/"); driver.manage().window().maximize();
Thread.sleep(2000);

List<WebElement> abc = driver.findElements(By.tagName("a"));
int num = abc.size(); System.out.println(num);

for(int i=0;i<=num-1;i++) { String
c = abc.get(i).getText();
Thread.sleep(1000);
System.out.println(c);
}
}

}
```

Identify checkbox and radio button and select them one by one:

```
package Selenium;

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 CheckboxHandeling {

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

System.setProperty("webdriver.chrome.driver", "D:/aaa/Chrome driver/chromedriver.exe");
WebDriver driver=new ChromeDriver();
driver.get("http://demo.guru99.com/test/radio.html");
driver.manage().window().maximize();
Thread.sleep(2000);

List<WebElement> a = driver.findElements(By.name("webform")); int
num =  a.size();

for(int i=0;i<num;i++) { a.get(i).click();

Thread.sleep(2000);
}
}

}
```

QA

Auto suggestion selenium code:

```
package Selenium;
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 AutoSuggestionDemo {

public static void main(String[] args) throws InterruptedException {
System.setProperty("webdriver.chrome.driver","D:/aaa/Chrome driver/chromedriver.exe");
WebDriver driver=new ChromeDriver();

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

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

Thread.sleep(2000);

driver.findElement(By.name("q")).sendKeys("Selenium");

Thread.sleep(2000);

List<WebElement> a = driver.findElements(By.tagName("li"));
int size = a.size(); System.out.println(size); for(int i=0;i<=size-
1;i++) { String abc = a.get(i).getText();
System.out.println(abc);
Thread.sleep(1000);

}

}

}
```

Data driven:

```
package Selenium;
```

QA

```
import java.io.FileInputStream; import
java.io.FileNotFoundException; import
java.io.IOException;

import org.apache.poi.EncryptedDocumentException;
import org.apache.poi.ss.usermodel.Sheet; import
org.apache.poi.ss.usermodel.WorkbookFactory;
import org.openqa.selenium.By; import
org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;

public class DataDriven {

public static void main(String[] args) throws InterruptedException,
EncryptedDocumentException, IOException {
System.setProperty("webdriver.chrome.driver","E:/Soft/chromedriver.exe");
WebDriver driver=new ChromeDriver();

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

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

Thread.sleep(2000);

//Open the Excel sheet

FileInputStream excel = new FileInputStream("C:/Users/Admin/Desktop/datadriven.xlsx");

Sheet a= WorkbookFactory.create(excel).getSheet("Sheet1");

String b= a.getRow(2).getCell(2).getStringCellValue();
String c=a.getRow(7).getCell(5).getStringCellValue();

Thread.sleep(2000);

driver.findElement(By.name("email")).sendKeys(b);
driver.findElement(By.name("pass")).sendKeys(c);

Thread.sleep(2000);
driver.findElement(By.name("login")).click();

}

}
```

Page object Model(POM):

QA

In normal java programing constructor are mainly used to initialize data member or variable

```
    public class Test

       {          int a ;  // variable
declaration



        }
      Test()



         {
                    a=20;  //initialization

          }

      public void test1()
      {

        s.o.p(a);    //use

      }
```

----

main class-test class

Regular class-POM class

Concepts use

1)Encapsulation

2)Annotation

QA

1)Encapsulation:

Whenever in oops ,we have to make any data member of class usable for only that class ,that time we declare it as private,this is known as Encapsulation .

OR

Encapsulation is the Wrapping of the data .
----

2)Annotation

Annotation contains some code whenever we use annotation, then at time of exicution that code is get exicuted.

```
@findBy(xpath="xpathExpression")
```

POM Class

1)POM class1

Example:

```
package pom_class;

import org.openqa.selenium.WebDriver; import
org.openqa.selenium.WebElement; import
org.openqa.selenium.support.FindBy; import
org.openqa.selenium.support.PageFactory;

public class PomDemo1 {

@FindBy(xpath="//input[@name='q']")  private WebElement SEARCH;

public PomDemo1(WebDriver driver)
```

QA

```
{
PageFactory.initElements(driver,this) ;
}



public void search() {


SEARCH.sendKeys("Selenium");


}
}
```

----

2)POM class2

Example:

```
package pom_class;

import org.openqa.selenium.WebDriver; import
org.openqa.selenium.WebElement; import
org.openqa.selenium.support.FindBy; import
org.openqa.selenium.support.PageFactory;

public class PomDemo2 {

@FindBy(xpath="//a[@class='gb_f'][1]")  private WebElement GMAIL;

public PomDemo2(WebDriver driver)
{
PageFactory.initElements(driver,this) ;
}


public void gmail() {

GMAIL.click();

}

}
```

QA

3.Main Class:

```
package pom_class;

import org.openqa.selenium.WebDriver; import
org.openqa.selenium.chrome.ChromeDriver;

public class TestClass {



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

System.setProperty("webdriver.chrome.driver", "E:/Soft/chromedriver.exe");
WebDriver driver=new ChromeDriver();
driver.get("https://www.google.co.in/");
driver.manage().window().maximize();
Thread.sleep(2000);

PomDemo1  pom = new PomDemo1(driver);
pom.search(); Thread.sleep(2000);
PomDemo2  pom1 = new PomDemo2(driver); pom1.gmail();

}
}
```

TestNG :

In selenium using java there are two TestNG framework available
1)JUnit
2)TestNG

TestNG is a testing framework design to simplify a broad range of testing needs from unit testing to system testing

TestNG is an open source framework where NG stand for Next Genaration

QA

TestNG is inspired from JUnit

Main method is not used for TestNG programs.

TestNG programs contains only methods that contain @Test Annotation .

if we don't write @Test annotation then this method will not exicute

Advantages of TestNG

1)TestNG annotation are easy to create Test cases.
2)Test cases can be grouped and prioterized more easily.
3)Exicute multiple programs / classes using xml.
4)Generate HTML reports.
5)Parallel test exicution is possible
6)Grouping of the test cases

Simple Program

```
package TestNG;

import org.testng.annotations.Test;

import junit.framework.Assert;

public class ClassName1 {
@Test   public void
verifyTitle() {
            Assert.assertEquals("Yahoo", "Yahoo");
  }
```

QA

```
  @Test   public void
abcd() {
            Assert.assertEquals("Gmail", "Gmail1");
  }
  @Test   public
void abc() {
            Assert.assertEquals("Gmail", "Gmail");
  }
}
```

QA

TestNG Priority

```
package TestNG;

import org.testng.annotations.Test;

import junit.framework.Assert;

public class ClassName1 { @Test(priority=2)
public void verifyTitle() {

System.out.println("Verify Title method");
}
@Test(priority=1) public
void abcd() {
System.out.println("abcd method");  }
@Test(priority=3) public
void abc() {
System.out.println("abc method");  }
}
--------------------------------------------------------------------------------
```

dependsOnMethods :

1)If first method is depends on second method ,if second method failed then first method will get skipped.
2)Only that two(which having reference of "dependsOnMethods") method will be exicuted

Program:

```
package TestNG;

import org.testng.annotations.Test;

import junit.framework.Assert;

public class ClassName1 {
  @Test
  public void login() {

              System.out.println("Login successfully");
  }
  @Test   public void
logout() {
              System.out.println("Logout successfully");
              }
  @Test(dependsOnMethods= {"advancedSearch"})
public void search() {
              System.out.println("Search successfully");
}
@Test public void
advancedSearch() {
              Assert.assertEquals("Gmail", "Gmail1");
}
}
```

QA

alwaysRun=true :

for ignoring the dependency of methods

```
package TestNG;

import org.testng.annotations.Test;

import junit.framework.Assert;

public class ClassName1 {
  @Test   public void
login() {


        System.out.println("Login successfully");
  }
  @Test   public void
logout() {
        System.out.println("Logout successfully");
        }
  @Test(dependsOnMethods= {"advancedSearch"},alwaysRun=true)
public void search() {
        System.out.println("Search successfully");
}
@Test
public void advancedSearch() {
        Assert.assertEquals("Gmail", "Gmail1");
}
}
```

---

1)@BeforeMethod
2)@AfterMethod

1)@BeforeMethod

This method exicute before each methods

QA

2)@AfterMethod

This method exicute after each methods

```
Ex: package
TestNG;

import org.testng.annotations.AfterMethod; import
org.testng.annotations.BeforeMethod; import
org.testng.annotations.Test;

import junit.framework.Assert;

public class ClassName1 {
@BeforeMethod
  public void login() {

              System.out.println("Login successfully");
  }
  @AfterMethod
  public void logout() {
              System.out.println("Logout successfully");
              }
  @Test(priority=2)   public
void addProduct() {
              System.out.println("Add product successfully");
}
@Test(priority=1) public
void addVendor() {
            System.out.println("Add vendor successfully");
}
@Test(priority=3) public
void addCurrency() {
            System.out.println("Add currency  successfully");
}
}
```

QA

------ Output:

```
[RemoteTestNG] detected TestNG version 7.4.0
Login successfully
Add vendor successfully
Logout successfully
Login successfully
Add product successfully
Logout successfully
Login successfully
Add currency  successfully
Logout successfully
PASSED: addProduct
PASSED: addVendor
PASSED: addCurrency

===============================================
    Default test
    Tests run: 3, Failures: 0, Skips: 0
===============================================


===============================================
Default suite
Total tests run: 3, Passes: 3, Failures: 0, Skips: 0
===============================================
```

----

3)@BeforeClass

4)@AfterClass

3)@BeforeClass

This method is exicute one time before the class.

QA

4)@AfterClass

This method is exicute one time after the class.

Ex:

```
package TestNG; import
org.testng.annotations.AfterClass; import
org.testng.annotations.AfterMethod; import
org.testng.annotations.BeforeClass; import
org.testng.annotations.BeforeMethod;
import org.testng.annotations.Test;

import junit.framework.Assert;

public class ClassName1 {
@BeforeClass   public
void login() {

            System.out.println("Login successfully");
  }
  @AfterClass   public
void logout() {
            System.out.println("Logout successfully");
            }
  @Test(priority=2)   public
void addProduct() {
            System.out.println("Add product successfully");
}
@Test(priority=1) public
void addVendor() {
          System.out.println("Add vendor successfully");
}
@Test(priority=3) public
void addCurrency() {
          System.out.println("Add currency  successfully");
```

QA

```
}
} -----------
```

Output:

```
[RemoteTestNG] detected TestNG version 7.4.0
Login successfully
Add vendor successfully
Add product successfully
Add currency  successfully
Logout successfully
PASSED: addVendor
PASSED: addCurrency
PASSED: addProduct

===============================================
    Default test
    Tests run: 3, Failures: 0, Skips: 0
===============================================


===============================================
Default suite
Total tests run: 3, Passes: 3, Failures: 0, Skips: 0
===============================================
```

QA

---

5)@BeforeTest

6)@AfterTest

5)@BeforeTest

This method exicute once before all classes.

6)@AfterTest

This method exicute once after all classes.

Ex: **Program: 1:**

```
package TestNG;
import
org.testng.Assert
; import
org.testng.annota
tions.AfterClass;
import
org.testng.annota
tions.AfterMetho
d; import
org.testng.annota
tions.BeforeClas
s; import
org.testng.annota
tions.BeforeMet
hod; import
org.testng.annota
tions.Test;

public class DependsOnMethods {

            @BeforeClass
             public void login() {

                        System.out.println("Login successfully");
             }
             @AfterClass
             public void logout() {
                        System.out.println("Logout successfully");
                        }
             @Test(priority=2)
public void addProduct() {
                        System.out.println("Add product successfully");
            }
```

QA

```
        @Test(priority=1)
public void addVendor() {
                System.out.println("Add vendor successfully");
        }
        @Test(priority=3)
public void addCurrency() {
                System.out.println("Add currency  successfully");
        }
}
```

Program 2:

```
package TestNG;

import org.testng.Assert; import
org.testng.annotations.Test;

public class Abc {

        @Test
         public void verifyTitle() {
                 Assert.assertEquals("Yahoo", "Yahoo");
         }
@Test
         public void abcd() {
                 Assert.assertEquals("Gmail", "Gmail1");
         }
@Test
         public void abc() {
                 Assert.assertEquals("Gmail", "Gmail");
         }

}
```

QA

Program 3:

```
package TestNG;

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


public class Priority {
        @BeforeTest
         public void login() {

                 System.out.println("Test Login successfully");
         }
         @AfterTest
         public void logout() {
                 System.out.println("Test Logout successfully");
                 }

        @Test(priority=2)
public void verifyTitle() {

                 System.out.println("Verify Title method");
         }

         @Test(priority=1)
public void abcd() {
                 System.out.println("abcd method");  }

         @Test(priority=3)
public void abc() {
                 System.out.println("abc method");
```

QA

```
                        }
}
```

------------ Suite file is:

```
<?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="TestNG.DependsOnMethods"/>
       <class name="TestNG.Abc"/>
       <class name="TestNG.Priority"/>

    </classes>
  </test> <!-- Test -->
</suite> <!-- Suite -->
```

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

Output:

```
[RemoteTestNG] detected TestNG version 7.4.0
Test Login successfully
Login successfully
Add vendor successfully
Add product successfully
Add currency  successfully
Logout  successfully  abcd
method     Verify     Title
method abc method
Test Logout successfully

===============================================
Suite
Total tests run: 9, Passes: 8, Failures: 1, Skips: 0
===============================================
```

QA

----

7)@BeforeSuite

8)@AfterSuite

7)@BeforeSuite

This method exicute once before @BeforeTest method

8)@AfterSuite

This method exicute once after @AfterTest method

Program 1:

```
package TestNG; import org.testng.Assert;
import org.testng.annotations.AfterClass;
import org.testng.annotations.AfterMethod;
import org.testng.annotations.BeforeClass;
import org.testng.annotations.BeforeMethod;
import org.testng.annotations.Test;

public class DependsOnMethods {

            @BeforeClass
             public void login() {

                        System.out.println("Login successfully");
             }
             @AfterClass
             public void logout() {
                        System.out.println("Logout successfully");
                        }
             @Test(priority=2)
public void addProduct() {
                        System.out.println("Add product successfully");
           }
```

```
            @Test(priority=1)
    public void addVendor() {
                        System.out.println("Add vendor successfully");
            }
            @Test(priority=3)
    public void addCurrency() {
                        System.out.println("Add currency  successfully");
            }
    }
```

Program 2:

```
package TestNG;

import org.testng.Assert; import
org.testng.annotations.Test;

public class Abc {

            @Test
              public void verifyTitle() {
                          Assert.assertEquals("Yahoo", "Yahoo");
              }
@Test
              public void abcd() {
                          Assert.assertEquals("Gmail", "Gmail1");
              }
@Test
              public void abc() {
                          Assert.assertEquals("Gmail", "Gmail");
              }


}
```

Program 3:

QA

```
package TestNG; import
org.testng.annotations.AfterClass; import
org.testng.annotations.AfterSuite; import
org.testng.annotations.AfterTest; import
org.testng.annotations.BeforeClass; import
org.testng.annotations.BeforeSuite; import
org.testng.annotations.BeforeTest; import
org.testng.annotations.Test;

public class Priority {
    @BeforeSuite
    public void b_suite() {
        System.out.println("This is @BeforeSuite method");
    }
    @AfterSuite
    public void a_suite() {
        System.out.println("This is @AfterSuite method");
    }
     @BeforeTest
     public void login() {

         System.out.println("Test Login successfully");
     }
     @AfterTest
     public void logout() {
         System.out.println("Test Logout successfully");
         }

     @Test(priority=2)
public void verifyTitle() {
System.out.println("Verify Title
method");
     }
```

QA

```
        @Test(priority=1)
public void abcd() {
              System.out.println("abcd method");  }

        @Test(priority=3)
public void abc() {
              System.out.println("abc method");
              }
}
```

========

Suite file is:

```
<?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="TestNG.DependsOnMethods"/>
       <class name="TestNG.Abc"/>
       <class name="TestNG.Priority"/>

    </classes>
  </test> <!-- Test -->
</suite> <!-- Suite -->
```

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

Output:

```
[RemoteTestNG] detected TestNG version 7.4.0
This is @BeforeSuite method
Test Login successfully
Login successfully
Add vendor successfully
Add product successfully
Add currency  successfully
Logout  successfully  abcd
```

```
method      Verify      Title
method abc method
Test Logout successfully
This is @AfterSuite method

===============================================
Suite
Total tests run: 9, Passes: 8, Failures: 1, Skips: 0
===============================================
---------------------------------------------------------------------------------------------------------
-
```

QA

Sequence of annotations:

1.@BeforeSuite

2.@BeforeTest

3.@BeforeClass

4.@BeforeMethod

5.@Test(as per priority)

6.@AfterMethod

7.@AfterClass

8.@AfterTest

9.@AfterSuite

Grouping test cases:

In grouping we make the group of test cases,and access those test cases from xml file by mentioning the group name which test case we required.

XML file syntax for grouping is:

Program 1:

```
package TestNG_Grouping;

import org.testng.annotations.Test;

public class ClassName1 {
  @Test(groups= {"sanity","regression"},priority=1)
public void login() {


              System.out.println("Login successfully");
  }
  @Test(groups= {"sanity","regression"},priority=10)
  public void logout() {
              System.out.println("Logout successfully");
              }
  @Test(groups= {"sanity"},priority=4)
public void search() {
              System.out.println("Search successfully");
}
@Test(priority=2) public
void addVendor() {
            System.out.println("Add vendor successfully");
}
```

QA

```
@Test(groups= {"regression"},priority=3) public
void advancedSearch() {
            System.out.println("Advanced search successfully");
}
@Test(groups= {"sanity","regression"},priority=5) public
void prepaidRecharge() {
System.out.println("Prepaid recharge successfully");
}
@Test(groups= {"regression"},priority=6) public
void billPayments() {
System.out.println("Bill payment successfully");
}
}
```

=====

xml file:

```
<?xml version="1.0" encoding="UTF-8"?>
<suite name="Suite" parallel="false">
  <test name="Test">
  <groups>
  <run>
  <include name ="sanity"/>
  </run>
  </groups>
    <classes>
      <class name="TestNG_Grouping.ClassName1"/>
     </classes>
  </test> <!-- Test -->
</suite> <!-- Suite -->
```

=======

Output:

```
[RemoteTestNG] detected TestNG version 7.4.0
[TestNGContentHandler] [WARN] It is strongly recommended to add "<!DOCTYPE suite
SYSTEM "https://testng.org/testng-1.0.dtd" >" at the top of the
```

```
suite    file [C:\Users\Admin\eclipse-workspace\April
Batch\src\TestNG_Grouping\ClassName1.xml] otherwise TestNG may fail or not work as
expected.
Login successfully
Search successfully
Prepaid recharge successfully
Logout successfully

===============================================
Suite
Total tests run: 4, Passes: 4, Failures: 0, Skips: 0
===============================================
```

QA

*** Parallel Test Exicution :**

Thread:-

A Thread is concurrent unit of execution.

There are two types of Parallel Test Exicution:

1. Parallel Test Exicution Methods

2. Parallel Test Exicution Class

----

1. Parallel Test Exicution Methods:

Program:

```
package TestNG_Parallel_Test_Exicution; import
org.testng.annotations.Test;

public class ParallelTestExicutionMethods {
  @Test
  public void testCase1() {
              long id=Thread.currentThread().getId();
              System.out.println("Test case 1 is successful"+" Thread id :"+id);
  }
  @Test   public void
testCase2() {
              long id=Thread.currentThread().getId();
              System.out.println("Test case 2 is successful"+" Thread id :"+id);
  }
  @Test   public void
testCase3() {
              long id=Thread.currentThread().getId();
              System.out.println("Test case 3 is successful"+" Thread id :"+id);   }
  @Test   public void
testCase4() {
              long id=Thread.currentThread().getId();
              System.out.println("Test case 4 is successful"+" Thread id :"+id);
  }
}
---------
```

XML file:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE suite SYSTEM "https://testng.org/testng-1.0.dtd">
<suite name="Suite" parallel="methods" thread-count="2">
  <test name="Test">
    <classes>
```

QA

```
     <class name="TestNG_Parallel_Test_Exicution.ParallelTestExicutionMethods"/>
</classes>
  </test> <!-- Test -->
</suite> <!-- Suite -->
-----------
```

Output:

```
[RemoteTestNG] detected TestNG version 7.4.0
Test case 2 is successful Thread id :15
Test case 1 is successful Thread id :14
Test case 4 is successful Thread id :14
Test case 3 is successful Thread id :15

===============================================
Suite
Total tests run: 4, Passes: 4, Failures: 0, Skips: 0
===============================================
```

2. Parallel Test Exicution Classes:

Here as example we are taking two classes

Class 1 :

```
package TestNG_Parallel_Test_Exicution_Classes;

import org.testng.annotations.Test;

public class ParalellTestExicutionClass1 {
          @Test
            public void testCase1() {
                      long id=Thread.currentThread().getId();
                      System.out.println("Test case 1 is successful"+" Thread id :"+id);
            }
@Test
```

QA

```
            public void testCase2() {
                      long id=Thread.currentThread().getId();
                      System.out.println("Test case 2 is successful"+" Thread id :"+id);
            }
@Test
            public void testCase3() {
                      long id=Thread.currentThread().getId();
                      System.out.println("Test case 3 is successful"+" Thread id :"+id);
            }
@Test
            public void testCase4() {
                      long id=Thread.currentThread().getId();
                      System.out.println("Test case 4 is successful"+" Thread id :"+id);
            }
}
```

Class 2:

```
package TestNG_Parallel_Test_Exicution_Classes;

import org.testng.annotations.Test;

public class ParalellTestExicutionClass2 {
           @Test
            public void testCase5() {
                      long id=Thread.currentThread().getId();
                      System.out.println("Test case 5 is successful"+" Thread id :"+id);
            }
            @Test
            public void testCase6() {
                      long id=Thread.currentThread().getId();
                      System.out.println("Test case 6 is successful"+" Thread id :"+id);
            }
}
```

QA

XML file:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE suite SYSTEM "https://testng.org/testng-1.0.dtd">
<suite name="Suite" parallel="classes" thread-count="2">
  <test thread-count="5" name="Test">
    <classes>
       <class name="TestNG_Parallel_Test_Exicution_Classes.ParalellTestExicutionClass1"/>
     <class name="TestNG_Parallel_Test_Exicution_Classes.ParalellTestExicutionClass2"/>
    </classes>
  </test> <!-- Test -->
</suite> <!-- Suite -->
```

Output:

```
[RemoteTestNG] detected TestNG version 7.4.0
Test case 1 is successful Thread id :14
Test case 5 is successful Thread id :15
Test case 2 is successful Thread id :14
Test case 6 is successful Thread id :15
Test case 3 is successful Thread id :14
Test case 4 is successful Thread id :14

===============================================
Suite
Total tests run: 6, Passes: 6, Failures: 0, Skips: 0
===============================================
```

----

invocationCount in TestNG

Invocation count is used when you want to run the same tests multiple times. Below example illustrates how to use invocation count in TestNG. In below example, test1 will be executed 5 times.

```
package TestNG;

import org.testng.annotations.Test;

public class InvocationCountDemo {
@Test(invocationCount = 5)    public
void test1(){
    System.out.println("Invocation count demo");

}
}
```

---

enabled = false :

Sometimes, it happens that our code is not ready and the test case written to test that method/code fails. In such cases, annotation **@Test(enabled = false)** helps to disable this test case.

QA

If a test method is annotated with *@Test(enabled = false)*, then the test case that is not ready to test is bypassed.

```
package TestNG;

import org.testng.annotations.Test;

public class EnabledequaltoFalseDemo {
            @Test(enabled = false)
public void btest1() {
        System.out.println("B.btest1");
    }

}
```

timeOut=time in millisecond:

If a test class contains multiple test methods, if one of the test method is time consuming to execute then TestNG by default fail that test method and execute other test methods which can be possible using timeOut.

Example:

```
package TestNG;

import org.testng.annotations.Test;

public class TimeOutDemo {
            @Test
            public void ContactVerify(){
               System.out.println("Contact validation is successful");
            }
 @Test(timeOut = 1000)  public void
LandingPage(){
```

QA

```
      System.out.println("Landing page verification is successful");
    }
    @Test
    public void LoanContact(){
      System.out.println("Loan contact details verification is successful");
    }

}
```

QA

Hard and Soft Assertions in Selenium

1)Hard Assert

2)Soft Assert

1)Hard Assert:

When assertion get fail then selenium stop the remaining execution .

Program:

```
package HardAssertSoftAssert;

import org.testng.Assert; import
org.testng.annotations.Test;

public class Test1 {

        @Test
        public void verifyPageTitle() {
                String expected_Title = "Google";
                String actual_Title = "Google1";
                System.out.println("Test case exicution started");
                Assert.assertEquals(actual_Title, expected_Title);
                System.out.println("Test case exicution finished");



        }

}
```

QA

Output:

```
[RemoteTestNG] detected TestNG version 7.4.0
Test case exicution started FAILED: verifyPageTitle java.lang.AssertionError:
expected [Google] but found [Google1]        at
org.testng.Assert.fail(Assert.java:99)            at
org.testng.Assert.failNotEquals(Assert.java:1037)    at
org.testng.Assert.assertEqualsImpl(Assert.java:140)            at
org.testng.Assert.assertEquals(Assert.java:122)        at
org.testng.Assert.assertEquals(Assert.java:629)        at
org.testng.Assert.assertEquals(Assert.java:639)        at
HardAssertSoftAssert.HardAssert.verifyPageTitle(HardAssert.java:13)      at
sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)     at
sun.reflect.NativeMethodAccessorImpl.invoke(Unknown Source)   at
sun.reflect.DelegatingMethodAccessorImpl.invoke(Unknown Source)        at
java.lang.reflect.Method.invoke(Unknown Source)
            at
org.testng.internal.MethodInvocationHelper.invokeMethod(MethodInvocationHelper.java:13
3)          at org.testng.internal.TestInvoker.invokeMethod(TestInvoker.java:598)
at org.testng.internal.TestInvoker.invokeTestMethod(TestInvoker.java:173)
at org.testng.internal.MethodRunner.runInSequence(MethodRunner.java:46)
            at
org.testng.internal.TestInvoker$MethodInvocationAgent.invoke(TestInvoker.java:824)
at org.testng.internal.TestInvoker.invokeTestMethods(TestInvoker.java:146)
            at
org.testng.internal.TestMethodWorker.invokeTestMethods(TestMethodWorker.java:146)
at org.testng.internal.TestMethodWorker.run(TestMethodWorker.java:128)           at
java.util.ArrayList.forEach(Unknown Source)          at
org.testng.TestRunner.privateRun(TestRunner.java:794)      at
org.testng.TestRunner.run(TestRunner.java:596)        at
org.testng.SuiteRunner.runTest(SuiteRunner.java:377)        at
org.testng.SuiteRunner.runSequentially(SuiteRunner.java:371)

            at org.testng.SuiteRunner.privateRun(SuiteRunner.java:332)           at
org.testng.SuiteRunner.run(SuiteRunner.java:276)    at
org.testng.SuiteRunnerWorker.runSuite(SuiteRunnerWorker.java:53)        at
```

QA

```
org.testng.SuiteRunnerWorker.run(SuiteRunnerWorker.java:96)     at
org.testng.TestNG.runSuitesSequentially(TestNG.java:1212)          at
org.testng.TestNG.runSuitesLocally(TestNG.java:1134)      at
org.testng.TestNG.runSuites(TestNG.java:1063)       at
org.testng.TestNG.run(TestNG.java:1031)     at
org.testng.remote.AbstractRemoteTestNG.run(AbstractRemoteTestNG.java:115)  at
org.testng.remote.RemoteTestNG.initAndRun(RemoteTestNG.java:251)    at
org.testng.remote.RemoteTestNG.main(RemoteTestNG.java:77)


======================================================
    Default test
    Tests run: 1, Failures: 1, Skips: 0
======================================================



======================================================
Default suite
Total tests run: 1, Passes: 0, Failures: 1, Skips: 0
======================================================
```

2.Soft Assert:

When assertion get fail then selenium not stop the remaining execution ,remaining code line are exicuted.

Program:

```
package HardAssertSoftAssert;

import org.testng.Assert; import
org.testng.annotations.Test; import
org.testng.asserts.SoftAssert;
```

QA

```
public class SoftAssert1 {

        @Test
        public void verifyPageTitle() {
                String expected_Title = "Google";
                String actual_Title = "Google1";

                SoftAssert softassert = new SoftAssert();

                System.out.println("Test case exicution started");
softassert.assertEquals(actual_Title, expected_Title);

                String expected_PageUrl = "google.com";
                String actual_PageUrl = "google.com1";


                softassert.assertEquals(expected_PageUrl, actual_PageUrl);
                System.out.println("Test case exicution finished");


                }


}
-----------
```

OutPut:

```
[RemoteTestNG] detected TestNG version 7.4.0
Test case exicution started
Test case exicution finished
PASSED: verifyPageTitle
```

QA

```
===============================================
    Default test
    Tests run: 1, Failures: 0, Skips: 0
===============================================


===============================================
Default suite
Total tests run: 1, Passes: 1, Failures: 0, Skips: 0
===============================================

===============
```

QA

How to take screenshot of failed test case?

Class 1:

```
package failedTestCaseScreenshot;

import org.openqa.selenium.WebDriver;

public class MainTest {
```

```
            public static WebDriver driver;


            }
```

--- Class

2:

```
package failedTestCaseScreenshot;


import java.io.File; import
java.io.IOException;


import org.apache.commons.io.FileUtils; import
org.openqa.selenium.OutputType; import
org.openqa.selenium.TakesScreenshot; import
org.openqa.selenium.io.FileHandler;


public class GetScreenshot extends MainTest {


public static String capture(String screenshotName) throws IOException {
File s = ((TakesScreenshot)driver).getScreenshotAs(OutputType.FILE);
File d = new File("C:/Users/Admin/Desktop/Failedtest case screenshot/abc.bmp");

FileUtils.copyFile(s,d);
return screenshotName;



}

}
```

-- Class

3:

```
package failedTestCaseScreenshot;

import java.io.IOException;

import org.testng.ITestListener; import
org.testng.ITestResult;

public class ListenerTest implements ITestListener{
```

```
public void onTestFailure(ITestResult result) { try
{
GetScreenshot.capture(result.getName());
} catch (IOException e) {
// TODO Auto-generated catch block e.printStackTrace();
}
}

}
```

-- **Class 4:**

```
package failedTestCaseScreenshot;

import org.openqa.selenium.chrome.ChromeDriver;
import org.testng.Assert;
import org.testng.annotations.Test;



public class CaptureScreenshot extends MainTest {

@Test
public void captureScreenshot() throws InterruptedException {
System.setProperty("webdriver.chrome.driver","E:/Soft/chrome_driver2/chromedriver.exe");
driver=new ChromeDriver();

driver.get("https://www.facebook.com/"); driver.manage().window().maximize();
Thread.sleep(2000);

String title =driver.getTitle(); //Facebook
- लॉग इन क िंवा साइन अप
Assert.assertEquals("Home", title);

Thread.sleep(6000);

driver.close();
}

}
```

Then right click on same package and select testNG--converttestNG--Next---Next--Finish

xml file will be generated

run that xml file

XML file:

```
<?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="failedTestCaseScreenshot.CaptureScreenshot"/>
    </classes>
  </test> <!-- Test -->
</suite> <!-- Suite -->
```

QA