* **Automation Testing :-**

Testing an applications feature with the help of automation tool & executing Test scripts is called AT.

* **Disadvantages of manual Testing**

1. Compatibility testing is difficult
2. Test cycle duration will be increased
3. more efforts are required
4. Regretion testing is time Consuming

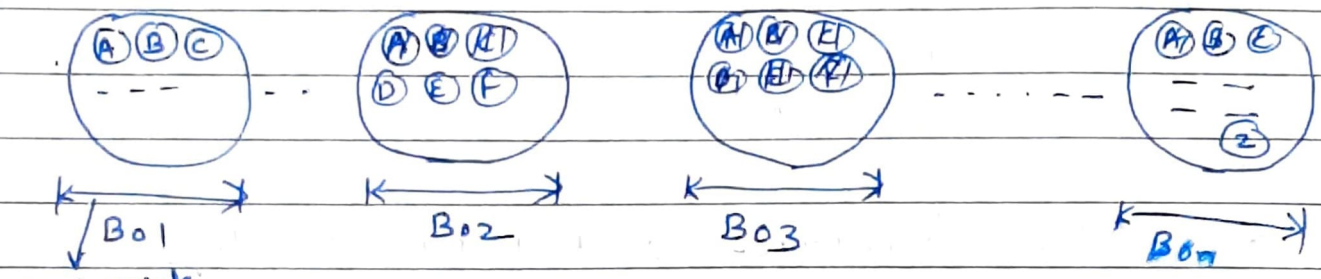
* **Advantages of Automation testing**

1. Reusibility of Test script
2. Compatibility testing is easy / possible
3. ~~cross browser /~~ Co less human efforts are require
4. project duration will be reduced
5. To overcome drawbacks of Regression Testing.
6. cost of project will be reduced
7. It is Reliable & Efficient.

* **Some of the automation tools:**

1. Selenium
2. QTP
3. Sahi / Sahi pro
4. Selendroid
5. Appium

* When we should do automation Testing ?



Automation testing tool will able to perform testing an application but to perform any action as a TE we need to give commands these commands are called Scripting.

* Advantages of Selenium :-

1. open Source
2. multi language Supportable
3. cross browser / Compatibility testing is possible
4. cross platform is also platform

* Disadvantages of Selenium :-

1. we can't automate web-based Application
2. we can't automate stand alone Appln.
3. Can't automate Captcha
4. Selenium will not support file uploading
5. Adhoc Test-Cases can't be automated.
6. Can't Read barcode

Java Concepts used in automation :-

1. Inheritence
2. Interface
3. polymorphism
4. casting (up casting)
5. Encapsulation
6. Abstraction
7. Arrays
8. Collections
9. for loops, for each, while loop & Iterator
10. Control statements
11. String class

Date
Page

Selenium Flavoues :-

① Selenium IDE (integrated develop. Enviornment)
- We can run script in only Firefox browse.
- Record & playback option
- We cant do Compatibility Testing

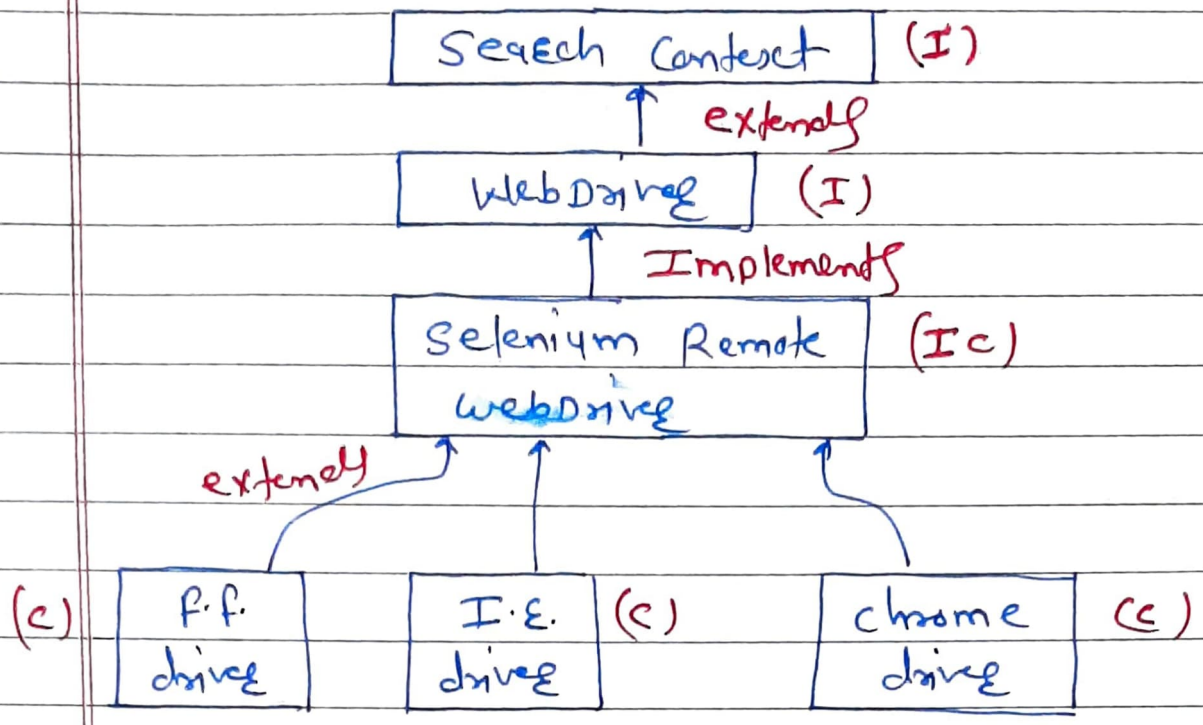
② Selenium RC (Remote Control)
- Suppoet C.T. (cross browse)
- We can run scripts in Java only

③ Selenium webDrive (Selenium tool)
- Supports C.T. (cross browse)
- we can run scripts in multiple language.

④ Selenium grid

⑤ Selendroid

* Selenium Architecture



① • Search Context is a supermost interface which contains abstract methods & inherited to webdriver.

② webdriver is an interface which containing abstract method of Search Context & its own abstract method.

③ All the Abstract methods are overriden in or implemented in Selenium Remote webdriver class.

④ Selenium Remote WebDriver :-
It is a class which implements all the abstract methods of both interface.

⑤ Selenium Remote webdriver class is extended to browsers such as ff driver, Chrome driver -- etc.

⑥ To run appln in multiple browsers (C.T.)
i.e. writing Test script by using single browser but run the same script in multiple browsers we need to use runtime polymorphism by using UpCasting in Selenium

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

- create an object of ChromeDriver class with reference of webDriver interface.

Q. How to open a browser in Selenium

```
System.setProperty("webdriver.chrome.driver",
                   "path of .exe file");

WebDriver driver = new ChromeDriver();
```

- create an object of chromedriver class & store it in 1 ref. variable with reference of webDriver interface.
- before that we need to set path of chromedriver .exe file

* **WebDriver :-**

It is an Interface use to perform action on browser.

* **Methods of webDriver :-**

① **get() method :**

This method is use to open an Appln or to Enter Url in a webPage or browser.

eg.

```
WebDriver driver = new chromeDriver();
driver.get("url");
```

① **webDriver exception :-**

We will get webDriver exception when url is not well formed

② **close() method :-**

This method is use to close current tab of the browser.

eg. `driver.close();`

③ **quit() method :-**

quit method is the alternate method to the close method,

but the diff bet^n close & quit is that close() will close current tab only & quit() will close all the tab of the browser.

Q. what if we interupt the browser while running Test script

→ We will get "unreachable browser exception"

④ getTitle() method :-

This method is use to get title of webpage as an output.

Return type of gettitle is String.

```
WebDriver driver = new ChromeDriver();
        driver.get("url");
String title = driver.getTitle();
        Sop(title);

String exptitle = "          ";

if( title.equalIgnoreCase(exptitle))
  {
      Sop ("Navigated to Correct webpage");
  }
else
  {
      Sop("navigated to wrong webpage");
  }
```

⑤ getCurrentUrl() method :-

This method is use to get url of the current webpage as an output.

Return type of getCurrentUrl is String

```
WebDriver driver = new ChromeDriver();
driver.get("url");
String url = driver.getCurrentUrl()
sop(url);
```

⑥ **maximize()** :-

This method is use to maximize the browser.

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

Note :- We can't minimize the browser using Selenium.

- But we can change size the of the browser.

⑦ **navigate()** :-

This method is use to open an application, move forward, backward & Refresh the browser.

- navigate method can be used for alternate method for get method.

Ⓐ Enter url in browser :-

```
WebDriver driver = new ChromeDriver();
driver.navigate().To("url");
```

(A) ~~Change the Size of browser~~
(B) move forward, move back & Refresh :-

```
WebDriver driver = new ChromeDriver();
    driver.navigate().To("http://google.com");
driver.navigate().To("http://facebook.com");
driver.navigate().back();
driver.navigate().forward();
driver.navigate().Refresh();
```

(8) SetSize();
This method is use to change size of the browser, which accepts Dimension arguments.

```
WebDriver driver = new ChromeDriver();
    driver.get("___");
Dimension D = new Dimension(100, 200);   // height, width
driver.manage().window().setSize(D);
```

(9) Setposition():
This method is use to change position of the browser. which accept point arg.

```
Dimension D = new Dimension(100, 200);
driver.manage().window().setSize(D);
Point P = new Point(300, 400);   // (x, y) cordinates
driver.manage().window().setPosition(P);
```

* Assignment on webDriver :-

Assignment (1)

step (1) : open browser, open google webpage
step (2) : verify google webpage opened
step (3) : wait for 2 sec. & maximize the browser
step (4) : change size of the browser
step (5) : change position
step (6) : close the browser

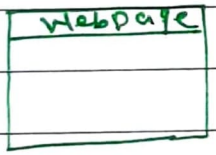
Assignment (2)

step (1) : open google webpage
step (2) : maximize the browser
step (3) : display title & url of webpage as an output
step (4) : open facebook webpage
step (5) : wait for 2 sec & navigate back to google webpage
step (6) : verify google webpage is opened or not
step (7) : wait for 3 sec & change size of browser (200, 400)
step (8) : wait for 2 sec change position of browser (100, 500) & wait for 2 sec.
step (9) : maximize browser
step (10) : wait for 2 sec & again navigate to fb page.
step (11) : verify facebook webpage opened
step (12) : Refresh the webpage
step (13) : wait for 2 sec & close the browser

* Html coding *

- Hypertext markup language used for creating a webpage
- Html coding is not case sensitive
- we can write html coding in notepad/ notepad ++.
- while saving html, extension of file should be "filename.html".

Q. write a html code to create dummy webpage

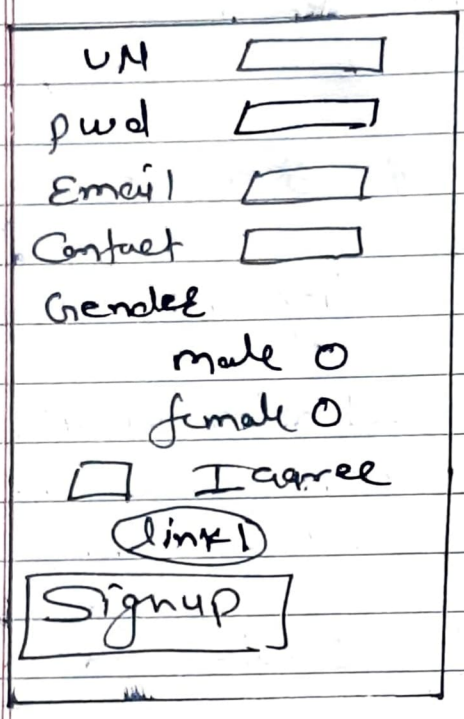



```
<html>
    <title>
        webpage
    </html>
</html>
```

Q. write a html code to create following webpage.

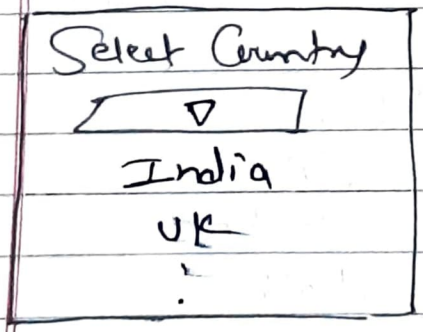

```
<html>
    <title>
        webpage
    </title>
    <body>
        Hello
        good mrng
    </body>
</html>
```

Q. write a html code ~~for~~ to create following webpage




```
<html>
  <body>
    UN <input type="text"> </br>
  pwd <input type="password"> </br>
Email <input type="text"> </br>
Contact <input type="text"> </br>
Gender
  male <input type="radio"> </br>
  female <input type="radio"> </br>
<input type="checkbox"> I agree </br>
<input type="button" value="signup">
<a href="____"> link1 </a>
</body>
</html>
```

Q. write a html code to create list-box




```
<html>
 <body>
  Select Country
   <select>
    <option> India </option>
    <option> USA </option>
    <option> UK </option>
    <option> AUS </option>
    <option> Eng <option>
   </select>
 </body>
</html>
```

Note :- To create multi selectable listbox we need to ~~create~~ use keyword multiple = 'true'

Syntax : < Select multiple = 'true' >

Q. Write a html code to create webtable :-

Sr No.	Book Type	Cost
1	Manual	100
2	SqL	200
3	Java	300

```
<html>
  <body>
    <table border='10'>
      <tbody>
        <tr>
          <th> Sr No. </th>
          <th> Book Type </th>
          <th> Cost </th>
        </tr>
        <tr>
          <td> 1 </td>
          <td> Manual </td>
          <td> 100 </td>
        </tr>
        <tr>
          <td> 2 </td>
          <td> Sql </td>
          <td> 200 </td>
        </tr>
        <tr>
          <td> 3 </td>
          <td> Java </td>
          <td> 300 </td>
        </tr>
      </tbody>
```

* Summary :-

1) To create a webpage we need to use keyword "Html".
2) Every keyword should be closed within angular brace using single forward slash (/)
3) To Creat a Componant or element we need to use a keyword "input".
4) To Create list we need to use "select" keyword.
5) To create link we need to use keyword "a" with hiper reference.
6) To Create a webTable we need to use keyword "table".
7) To Create Image we need to use "Img" keyword.

① Tagname :-

Any keyword which will be is present ~~after~~ & immidiatly after angular brace (<) left than Symbol
eg. Html, title, body, tr, table, ...

② Attribute :-

Any keyword which is present after tagname with equal to Symbol until greater than Symbol.

· Syntax: Property Name = 'property value'

Attribute Name = 'Attribute value'

eg.

id = '1234' class = 'abc' name = 'xyz'

↙ attribute name

↓ attribute value

(3) Text :-

Any keyword which is present in bet^n angular brace (>) greater than Symbol & angular brace (<) left than Symbol is known as Text.

eg > Sr No < > Link <

> Manual < > Sql <

Why Html Coding is Required in Selenium

To identify an element uniquely & to perform action with the help of Selenium, Html Coding is necessary.

Attribute Name = 'Attribute value'

eg.

id = '1234' class = 'abc' name = 'xyz'

attribute name — attribute value

(3) Text :-

Any keyword which is present in bet^n angular brace (>) greater than symbol & angular brace (<) left than symbol is known as Text.

eg > Sr No < > Link <

> Manual < > Sql. <

Q. Why Html Coding is Required in Selenium

To identify an element uniquely & to perform action with the help of Selenium, Html Coding is necessary.

Locaters

- "Locators are use to identify an element with the help of locater type".

- To identify an element present in webpage we need to use 'findElement()' method which is present in webdriver.

eg. WebDriver driver = new ChromeDriver();
driver.findElement (locater)
(By arg)

- findElement method will identify an element with the help of By class which containg static methods.

- All the static methods present in By class are known as locater type.

```
class By
{
 public static Tagname (String arg)
{
=
}
```

eg. driver.findElement (By.Tagname ("tagname"))

* Locator Types :-

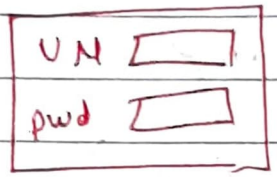
1. Tagname → Tagname
2. * id
3. name
4. Class Name
   → Attribute
5. linkText
6. partial linkText
   → Text
7. CSS Selector
8. ** Xpath
   → Expression

- All the locator types takes String arg as an input & return types is By.

```
By ele = By.Tagname(String arg);
```

- Return Type of findElement() is WebElement.

① Tagname :-




```
<html>
<body>
UN <input type='text'></br>
pwd <input type='password'></br>
</body>
</html>
```

```
WebDriver driver = new chromeDriver();
driver.get("url");
//UN
driver.findElement(By.tagname("input")).SendKeys(admin
// pwd
driver.findElement(By.tagname("input")).SendKeys("admin");
```

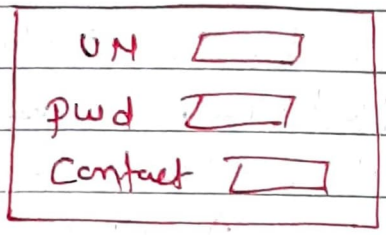
* **Disadvantage :-**

In a webpage if maybe multiple elements are presented with same tagname & we use tagname locatortype to identify an element then Selenium will perform action on 1st element present in webpage.

②

② Id :-

this locator type is use if any element html containg id attribute




```
<html>
<body>
UN <input type='text' id='1234'>
pwd <input type='password' id='abc'>
Contact <input type='text' id='abc'>
</body>
</html>
```

```
//UN
driver.findElement(By.id("1234")).sendKeys("____");
//pwd
driver.findElement(By.id('abc')).sendKeys("____");
```

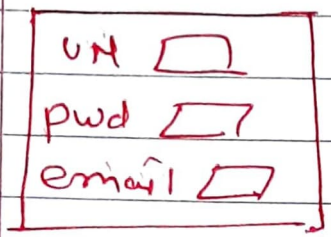
* When we can't use id of a Locater type
  1. when id attribute is not present
  2. when id attribute is duplicate

eg.

Locater for Contact :- By str = By.id("abc");

In this case selenium will perform action on pwd

③ className :-




```
<html>
  <body>
    UN <input  id='1234'  class='458'>
    pwd <input  id='1234'  class='910'>
    email <input  class='xyz'>
  </body>
</html>
```

Duplicate id

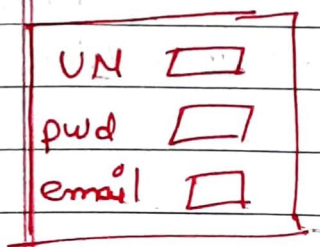
id is absent

```
//UN
    driver.FindElement(By.className("1234")).SendKeys("
//pwd
    driver.FindElement(By.className("910")).SendKeys("
//email
    driver.FindElement(By.className("xyz")).SendKeys("
```

* When we can use className of locator type
  1. if id is duplicate
  2. if id attribute is not present.

* When we can't use className
  1. className is ~~absent~~ Not present
  2. className is duplicate.

④ name :-




```
<html>
<body>
UN <input id='1234' class='xyz' >
pwd <input id='1234' class='xyz' name='45'>
email <input name='abcd'>
</body>
</html>
```

id & class are Duplicate

id & classname is Not present.

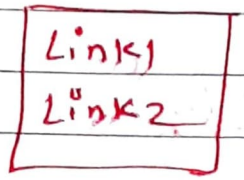
```
//UN
driver.FindElement(By.id("1234")).SendKeys("__");
//pwd
driver.findElement(By.name("45")).SendKeys("__");
// email
driver.findElement(By.name("abcd")).SendKeys("__");
```

⑤ LinkTeset :- ⑥ Partial LinkTeset

If Tagname is duplicate id, classname, name attribute are not present in html code of an element then we should use linkTeset or partial ~~Teset~~ linkTeset.

① linkTeset :- used to identify an element by taking entire teset of an input

② partial linkTeset :- used to identify element by taking few chars of a teset of an input.




```
<html>
  <body>
    <a href="___">Link1</a>
    <a href="___">Link2</a>
  </body>
</html>
```

Link1

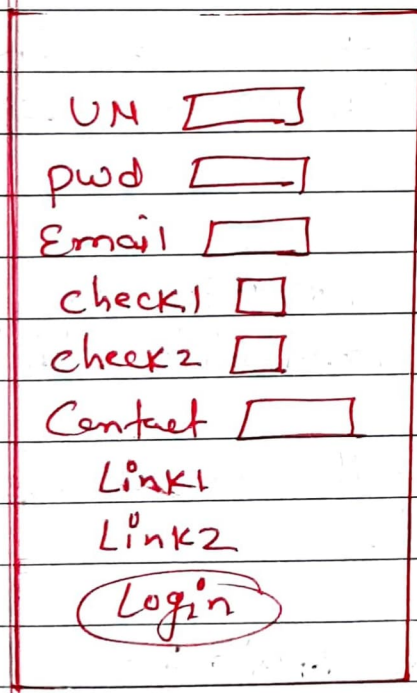
```
driver.findElement(By.LinkText("Link1")).click();
driver.findElement(By.partialLinkText("Li")).click();
```

Note:- linktext & partial LinkText locater type are use to identify elements with the help of link text present in links. but not for normal text.

⑧ Xpath :-

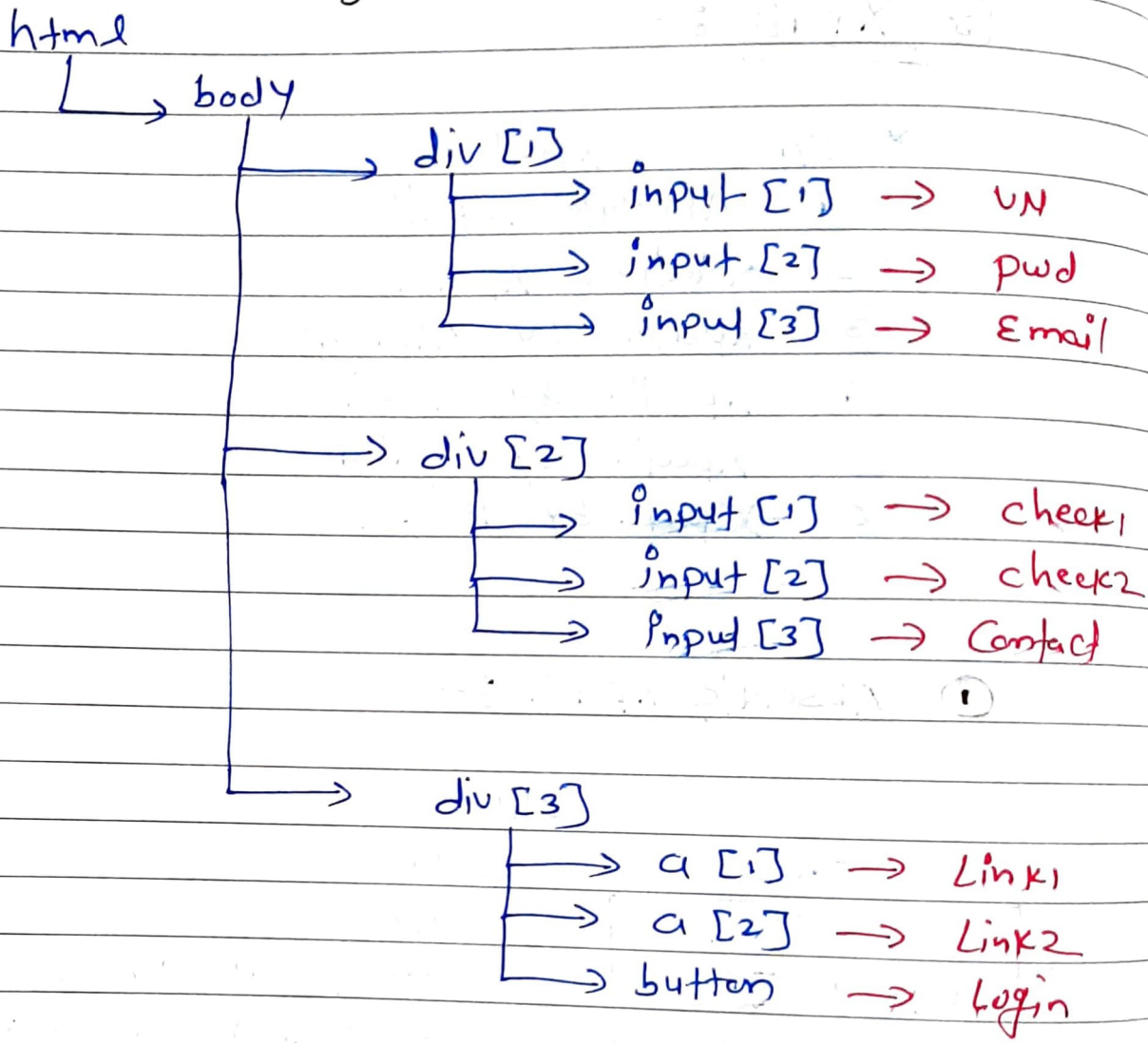
* Types of xpath
  1. Absolute x path
  2. Relative xpath
  3. xpath by Attribute
  4. xpath by Text
  5. xpath by Containg
  6. Xpath by index.

① Absolute xpath :-




```
<html>
  <body>
    <div>
      UN <input type = 'text' >
      pwd <input type = 'password' >
      Email <input type = 'text' >
    </div>
    <div>
      check1 <input type='checkbox'>
      check2 <input type='checkbox'>
      Contact <input type = 'text' >
    </div>
    <div>
      <a href = '__' > Link1 </a>
      <a href = '__' > Link2 </a>
      <button type='button' value='Log
    </div>
  </body>
</html>
```

* Html Tree Diagram :-



- Absolute Xpath is use to navigate from root of the parent to immidiate child.
- To Achieve absolute Xpath we need to use Singh forward slash (/).

* Disadvantage of Absolute Xpath :-

(1) Xpath is too lengthy & time Consuming.
(2) Identifying of an element by developing Html tree Diagram is difficult.

Element	Absolute xpath
UN →	/html/body/div[1]/input[1]
pwd →	/html/body/div[1]/input[2]
check1 →	/html/body/div[2]/input[2]
Link2 →	/html/body/div[3]/a[2]
login →	/html/body/div[3]/button

(2) Relative xpath :-

- Relative xpath is use to navigate from root of the parent to any child.

- To achieve Relative xpath we need to use Double forward slash (//).

* Disadvantages of Relative xpath :-

① Identifying of an element by developing html tree Diagram is dificult.

Element	Relative xpath
UN →	//div[1]/input[1]
pwd →	//div[1]/input[2]
check1 →	//div[2]/input[1]
Link2 →	//div[3]/a[2]
Login →	//div[3]/button

* Diff betn / & //

① / :- use to navigate from root of the parent to immidiate child.

② // :- use to navigate from root of the parent to any child.

③ Xpath by Attribute :-

Advantage :-

We can identify element by using attribute without finding html tree Diagram.

Attribute Name ↑ Attribute value ↑

```
//tagname[@propertyname = 'property value']
```

eg. //input[@id = 'abc']

```
<html>
  <body>
    UN <input id='abc' type='text'>
    pwd <input id='xyz' type='pass'>
  </body>
</html>
```

④ Xpath by Text :-

- Sometimes Developer may create an element by using tagname & text.
- To identify that text we can't use locator type ie. Xpath by attribute

```
//tagname[text() = 'text value']
```

- By using xpath by text we can identify normal text & link text also.

```
<html>
  <body>
    <a href='__'> Link1 </a>
    <a href='__'> Link2 </a>
  </body>
</html>
```

eg. //a[text() = 'Link1']
//a[text() = 'Link2']

Note :- If any element containing text with space while identifying that element using xpath by text. we need to mention space also.

eg. <Span> abc </Span>

//Span[text() = ' abc ']

- Space can be created by using 2 ways
  1. Keyboard Stroke
  2. Non-breakable space. Keyword →

- If space is created by using non-breakable spaces within the text. then we can't use xpath by text.
- So we need to go for next type ie. xpath by Contains.

List - box

Q. How to handle List box using Selenium

Step ①
Identify list box which need to be handled & store it in a reference variable.

Step ② : create an object of Select class which accept webElement argument.

Step ③ : use Select class methods to Select options.
eg. SelectByIndex (int index)
SelectBy Value (String value)
SelectByVisibleText (String Text)

Q. Write a Test script to enter DOB in facebook Signup page.

```
WebDriver driver = new ChromeDriver();
driver.get ("http://facebook.com");
```

Step ①
```
WebElement year = driver.FindElement (By.xpath(
                  "//select[@id='year']");
```

Step ②
```
Select S = new Select (year)
```

Step ③
```
S.SelectByIndex( );
   or
S.SelectBy Value ( )
   or
S.SelectByVisibleText( );
```

* List of methods present in Select class

1) SelectByIndex()
2) SelectByValue()
3) SelectByVisibleText()
4) DeselectByIndex()
5) DeselectByValue()
6) deselectByVisibleText()
7) isMultiple()
8) getAllSelectedOptions()
9) getFirstSelectedOption()
10) getOptions()

* getOptions() :-
  - this method is use to get all the options present in Listbox as an output.
  - Return type of getOptions() method is List<WebElements>.

Q. Write a Test Script to identify size of month List box in fb.com.

```
WebDriver d = new ChromeDriver();
d.get("http://facebook.com");
WebElement List = d.FindElement(By.xpath("____"));
Select S = new Select(List);

Sop(S.getOptions().Size());

or

List<WebElement> List1 = S.getOptions();
int size = List1.size();
Sop(size);
```

Q. write a Test script to display month List box options as an o/p in fb appln

```
WebDriver d = new chromeDriver();
        d.get("http://facebook.com");
WebElement month = d.findElement(By.xpath("__"));

Select s = new Select(month);
List<WebElement> list = s1.getOptions();
for( WebElement L : list )
  {
     Sop(L);
  }
```

Q. write a Test script to display month List box options in alphabetical order.

```
WebDriver d = new chromeDriver()
        d.get("http://facebook.com");
WebElement month = d.findElement(By.  )

Select S = new Select(month);
```

* for sorting operation arraylist should be created

```
ArrayList<String> rv = new ArrayList<String>
List<WebElement> list = S.getOptions();

for (WebElement L : list)
    { String s = L.getText();
      rv.add(s);
    }
```

```
Collections.sort(rv);
for(String s1 : rv)
{
    sop(s1);
}
```

* **ismultiple() :-**

this method is use to verify List-box are single selectable or multi-selectable.

Q. Write a Test Script to verify month List box is multi-selectable or not

```
WebDriver d = new ChromeDriver();
d.get("http://facebook.com");
WebElement rv = d.findElement(By.xpath("--"));
Select s = new Select(rv);

if(s.isEmpty())
  if(s.ismultiple())
  {
    sop("List is multi selectable");
  }
  else
  {
      sop("List box is not multi-
                    -selectable");
  }
```

* getAllSelectedOptions() :-
this method is use to get only selected options present in List-box as an o/p.

* To deselect any selected options in a List box we need to use below methods
(1) DeselectAll()
(2) deselectByIndex()
(3) deselectByValue()
(4) deselectByVisibleText()

Note :- these methods we can use on only multi-selectable List box.

Q. what happens if we try to deselect selected option in single selectable List-box.
→ We will get "unsuported operation Exception".

Q. what will happen if we try to perform deselect action on multi-selectable List-box, if any option is not selected
→ • Deselect action will not be performed as there is no option selected
• It will not through any Exception.

* getFirstSelectedOption() :-
this method is use to display 1st selected option in multi-selectable list box as an o/p.

Q How to Handle customized List-box

" the List-box which are created without using select tagname is known as customized List box. "

- We can't use Select class to select options present in customized List-box.
- To handle customized List-box we can use (① SendKeys & keys class)
  ② keys class

Q. write a TS. to enter destination or origin field in yatra.com

```
WebDriver d = new chromeDriver();
d.get("http://www.yatra.com");
```

Approach ①

```
WebElement origin = d.FindElement(____);
origin.click();
origin.SendKeys(Keys.Arrow-Down);
origin.SendKeys(Keys.Enter);
```

Approch ②

```
webElement dest = d.FindElement(___);
dest.click();
dest.SendKeys("banglore");
```

Q. What happen if we try to select any option present in customized list-box using select class

→ It will through " Unexpected Tagname Exception "

POP UP HANDLING

Page No.:
Date:
YOUVA

1) Pop up are small or separate window which will be displayed when we perform action on any component present in a web page.

2) These popup can be handle by Selenium Directly.

3) But sometimes we may need to use 3rd party tools to handle the popups.

4) Types of popup
   - → Hidden division popup
   - → Alert popup
   - → Child browser popup
   - → Authentication popup
   - → File upload
   - → File download popup.

5) If we are able to inspect element present in popup then we can use selenium directly to handle that popup.

6) If we are unable to inspect element present in popup then we need to use 3rd party tool to handle that popup.

Page No.:
Date:

1) Hidden Division Popup

1) These popups are colorfull.
2) We can inspect element present in popup.
3) We can not perform drag & drop action.
4) As we can inspect element then using selenium we can handle it & no need to switch.

Ex: popup on flipkart.

2) Alert popup

1) We can not inspect element present in popup.
2) We can drag & drop this popup.
3) These popup will contain ok, cancel, button & resul
4) These type of popup also contain '?' or '!' symb

→ How to handle Alert popup.

1) To handle this popup we need to switch seleniu focus from main page to alert popup by using

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

2) Alert is an interface which contain abstract metho
   - accept (use to click on ok button)
   - dismiss (use to click on cancel button)
   - getText (use to get text in alert popup)

→ To click on ~~ok~~ ok button:

```
alt.accept();
```

→ To click on cancel button:

```
alt.dismiss();
```

→ To get text present on popup.

```
String a = alt.getText();
System.out.println(a);
```

program

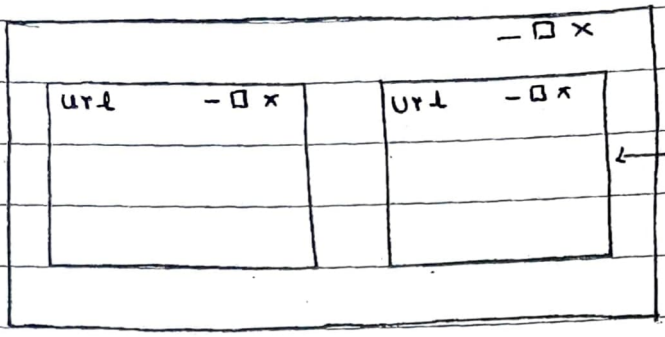
```
package demo;
public class test
{
 public static void main (String[] args)
  {
    System.setProperty("webdriver.chrome.driver", "path");
    WebDriver driver = new ChromeDriver();
    driver.get("url");

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

    alt.accept();
    alt.dismiss();
    String a = alt.getText();
    System.out.println(a);
  }
}
```

3) Child browser popup

1) These popup look like.



1) We can inspect element present in popup.
2) We can drag and drop.
3) These popup will contain address field (url), maximize, minimize, close options.

→ How to handle child browser popup?

1) To handle child browser popup we need to switch selenium focus from main page to child browser or window by using syntax:

```
driver.switchTo().window("String ID");
```

→ How to handle multiple child browser popup?

1) If multiple child browser popup will occured, then we need to identify add field or ID's which is unique

```
Set <String> ID's = driver.getWindowHandles();
String A = Ids.get("ID value");

driver.switchTo.window(A);
```

↓

Then perform actions using selenium.

* get Window Handles () :-

get address of main page as well as child popup & its return type is set<string> because it get multiple ID & set do not allow duplicate & ID are not duplicate.

* get Window Handle ()

get address of only main page so its return type is only String.

How to take Screenshot using selenium webdriver

- to take screenshot using selenium webdriver we need to type cast driver object into TakeScreenshot interface.
  eg. ((TakeScreenshot) driver)

- then we need to call the function i.e. getScreenshotAs(), here we need to pass the argument "outputType.file".

- this method will return object of type file.
  eg.
  File Src = ((TakeScreenShot) driver).
  getScreenshotAs(outputType.Fi

- when this statement will execute it will take the screenshot, but screenshot will be availab inside memory.

- To store screenshot in local drive, we need to call a method i.e. copy(_,_) which is present inside FileHandler class.

- So this function will accept 2 parameter ie. ① Source & ② destination

eg. FileHandler.copy(source, dest);

Excel Sheet

* Parameterization :-

- Fetching data from external source & using it in Selenium test script is known as parameterization.

- It can be achieve by using excel sheet, csv file, TestNG data provider.

How to fetch data from Excel Sheet?

Step① :-

To fetch data from excel sheet 1st we need to Configure Apache poi jar file into our project.

Step② :-

Create an excel sheet with some data & Save it in any drive.

Step③ :-

Create an object of "FileInputStream" class with excel sheet path as an input.

Step④ :-

To open an excel sheet we need to use a static method ie. create() which is present in "WorkBookFactory" class.

Step⑤ :-

To open specific sheet in an excel we need to use getSheet method.

Step ⑥ :-

To identify specific row in a sheet we need to use getrow() method.

Step ⑦ :-

To identify specific cell in a row we need to use getcell() method.

Step ⑧ :-

To fetch String type of information we need to call getStringCellValue() method

or

- To fetch numeric type of information we need to call getNumericCellValue() method.

write a code to fetch data from excel sheet.

```
FileInputStream file = new FileInputStream("path of
                                            Excel shu
String value = WorkBookfactory.create(file).
               getSheet("sheet name").getrow(int row
               .getcell(int cell).getStringCellValue();

S.o.p(value);
```

iframe

* **iframe** :- • Displaying webpage of a part of another webpage is known as iframe

• iframe will be created by using tagname "iframe".

Q. How to Handle iframe using selenium webdriver.

• TO handle iframe we need to switch selenium focus from main page to frame
e.g. driver.switchTo().frame()

• we can switch to iframe using 3 ways
ie. iframe ① id
② name
③ index

• once Actions are performed on Components present in iframe, Selenium will not navigate by default to main page.

• TO navigate from iframe to main page we need to use below methods like,
① parentFrame() or
② Default(Content)

• if we use driver.switchTo().parentFrame() then it will navigate from child frame to immediate parent frame.

• if we use driver.switchTo().defaultContent() then it will navigate from any child to main page

Customize Listbox

"The listbox which are created without using Select tagname is called Customize listbox".

Steps to handle customize listbox

① Identify the listbox which need to be handled & store it in a reference Variable of type webElement.

```
WebElement ele = driver.findElement(By.xPath("..."));
```

② Create an object of Actions class which accept webdriver object as an input.

```
Actions act = new Actions(driver);
```

③ click on the identified element

```
act.click(ele).perform();
```

or

```
act.moveToElement(ele).click().build().perform();
```

④ Select option

A> To move one option upward

Syntax: `act.SendKeys(Keys.Arrow-UP).perform();`

B> To move one option downward

Syntax: `act.sendkeys(Keys.Arrow_Down).perform();`

C> To Select option

Syntax: `act.sendkeys(Keys.Enter).perform();`

(5) Note :

A) To move arrow at top.

- Count how many options are present in listbox
eg. for month field 12 options are there

Syntax :-

```
for (int i=1;  i<=12;  i++)
{
  act.SendKeys (Keys.arrow_up).Perform
}
```

B) To move arrow at desire location

eg. for "may" option.

It is present at 5th position in listbox

Syntax :-

```
for (int i=1 :  i<=5 :  i++)
{
 act.SendKeys (Keys.arrow_Down).Perform()
}
```