1. Setproperty
2. Navigate() methods
3. Driver
4. Locator
5. Id
6. Name
7. Classname
8. Linktext
9. Partiallinktext
10. tagname
11. Cssselector
12. id
13. classname
14. other attribute
15. id + other attribute
16. classname + other attribute
17. parent-child relation
18. Xpath
    1. Absolute xpath
    2. Relative xpath

Relative Xpath methods

Basic xpath

1. contains

2. contains text

3. starts-with

4. text()

5. using or & and

AXES method

1. Following
2. Ancestor
3. Child
4. Preceding
5. Parent
6. Self
7. Following-sibling
8. Preceding sibling

Types of Wait

1. Implicit wait

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

1. Explicit wait

WebElement wait=new WebElement();

Wait.until(ExpectedConditions.VisibleTheElement Located(driver.findElement(By.xpath(“Values”))).sendKeys(“value”);

1. Fluent wait

* Drop Down Select class

WebElement dropdown=driver.findElement(By.xpath(“Values”));

Select selectorbt=new Select(dropdown);

selectorbt.selectByIndex(value);

selectorbt.selectByValue(value);

selectorbt.selectByVisibleText(value);

* Getwindowhandle & getwindowhandles

String str=driver.findElement(By.xpath(“Value”));

Set<String> str1= driver.findElement(By.xpath(“Value”));

Iterator itr=new Iterator();

While(itr.hasNext())

{

String child=str.next();

If(!str.equals(str1))

{

Driver.switchTo().window(child);

System.out.println(driver.findElement(By.xpath(“value”)).getText();

Driver.close();

}

Driver.switchTo().window(str);

Driver.close();

}

* Action class

Actions action= new Actions(WebDriver var);

1 Mouse class

Element Move

actions.moveToElement(var).perform();

Click

Actions.doubleclick(var).perform();

Actions.contextClick(var).perform();

Actions.click(var).perform();

Drag & Drop

Actions.DragAndDrop(dragvar,dropvar).perform()

ToolTip

WebElement tooltips=driver.findElement(By.*xpath*("value"));

actions.moveToElement(tooltips).perform();

Scroll

WebElement scrolla=driver.findElement(By.xpath(“value”));

Actions.moveToElement(scrolla).perform();

2. Keyboard class

Actions.keyDown(keys.CONTROL);

Actions.sendKeys(“A”);

Actions.keyUp(Keys.CONTROL);

Actions.build().perform();

\* Properties class

Create a new file project level

FileInputStream file=new FileInputStream(“File Path”);

Properties prop= new Properties();

Prop.load (file);

System.setProperty(“webdriver.chrome.driver”,”D://chromedriver.exe”);

WebDriver driver= get(“url”);

\* ChromeActions class for notification

ChromeOptions option=new ChromeOptions();

Option.argument(“—disable-notifications”);

WebDriver driver=new ChromeDriver(option);

\*Alert class

Driver.switchTo().alert().accept();

Driver.switchTo().alert().dismiss();

Driver.switchTo.alert().getText();

Driver.switchTo.alert().sendKeys(Value);

\* capturing Screenshot

\*Frame

Driver.switchTo().Frame(int index);

Driver.switchTo().frame(“string name/id”);

Driver.switchTo().frame(“webelement”);

Maven

Simple @Test program

Annotation

Priority

dependsOnMethod

simple dependsOnMethods

multiple dependsOnMethods

integratedependOnMethods

dependsOnGroups

Report

Assert types

1.default Assert/Hard Assert

2. SoftAssert/ verify