KASPER
ANALYTICS

kasperanalytics.com

+918130877931

Automation Testing Real Time Interview Questions

hr@kasperanalytics.com

kasper-analytics

1. Can you brief me about yourself?

Hi, my name is Pankaj. I started my career as a Testing Executive 4.5 years back with Infosys currently I am working as Test Engineer.

My responsibility is to understand Business Requirement Specification and High-Level scenarios and to convert them into test cases & Automation scripts if required. Execution of test cases and reporting of defect to the developer if there any and get them fixed. I have experience on Functional, Automation, Regression, Smoke, Sanity, Web accessibility, Web Analytics, Mobile Testing.

In my previous project I have worked on Automation testing where we have used Selenium with java and TestNG Cucumber framework for BDD approach. We have used Page object model where we have separated our test cases with page objects, and we performed testing on the same. For build management tool we are using Maven for version controlling we are using Git and for automating our jobs for nightly run or any schedule we are using Jenkins,.

For defect management & test case management we have used JIRA, TEST RAIL & HP ALM. I have worked on tools like BrowseStack, DeviceAnywhere, Toadsql,

I am working on Agile environment we have daily standup call and we have 2-week sprint cycle. I am part of 8-member team out of which we are 3-Tester, 2-dev, 1- manager, 1-scrum master

2. Tell me your Day to Day activities as QA?

First thing I do after login in my system. I check the active sprint in Jira for our project code. There I can see my assigned open tasks. After that I will check my mail if there is any important mail I need to take action on. Then we have our daily scrum meeting where we used to tell our previous day actions what we did, what we are planning for today and if we have any blocker to discuss. Product owner and scrum master help us to resolve that blocker. After that I need to take the pending task and do needed action whether creating test case, Execution, Defect retesting if any.

3. Do you have created framework from scratch, or you have maintained that?

I have not created Framework from scratch by myself but yes, I was part of framework creation and created some part of it.

4. How much you rate yourself in Java out of 10?

Out of 10 I will rate myself 6 in java as QA Automation engineer.

5. Can you tell me Oops concepts and relate it with your Framework?

We have Polymorphism, Inheritance, Encapsulation and Abstraction in Oops. So, we will start with

1) DATA ABSTRACTION : Data Abstraction means to handle complexity by hiding unnecessary details from the user. In java, abstraction is achieved by interfaces and abstract classes. We can achieve 100% abstraction using interfaces.

In Selenium, WebDriver itself acts as an interface. Consider the below statement:

WebDriver driver = new ChromeDriver();

We initialize the Chrome Browser using Selenium Webdriver. It means we are creating a reference variable (driver) of the interface (WebDriver) and creating an Object. Here WebDriver is an Interface and ChromeDriver is a class.

We can apply Data Abstraction in a Selenium framework by using the Page Object Model design pattern. We define all our locators and their methods in the page class. We can use these locators in our tests but we cannot see the implementation of their underlying methods. So we only show the locators in the tests but hide the implementation. This is a simple example of how we can use Data Abstraction in our Automation Framework.

2) ENCAPSULATION : Encapsulation is defined as the wrapping up of data under a single unit. It is the mechanism that binds together code and the data it manipulates. Encapsulation can be achieved by: Declaring all the variables in the class as private and writing public methods in the class to set and get the values

of variables. All the classes in an Automation Framework are an example of Encapsulation. In Page Object Model classes, we declare the data members using @FindBy and initialization of data members will be done using Constructor to utilize those in methods.

3) INHERITANCE Inheritance is the mechanism in java by which one class is allowed to inherit the features (fields and methods) of another class. We can apply Inheritance in our Automation Framework by creating a Base Class to initialize the WebDriver interface, browsers, waits, reports, logging, etc. and then we can extend this Base Class and its methods in other classes like Tests or Utilities. This is a simple example of how we can apply Inheritance in our framework.

4) POLYMORPHISM Polymorphism allows us to perform a single action in different ways. In Java polymorphism can be achieved by two ways: –

Method Overloading: When there are multiple methods with same name but different parameters then these methods are said to be overloaded. Methods can be overloaded by change in number of arguments or/and change in type of arguments. In Selenium Automation, Implicit wait is an example of Method Overloading. In Implicit wait we use different time stamps such as SECONDS, MINUTES, HOURS etc. –

Method Overriding: It occurs when a derived class has a definition for one of the member functions of the base class. That base function is said to be overridden. In Selenium Automation, Method Overriding can be achieved by overriding any WebDriver method. For example, we can override the findElement method In assertion we have used overload because in assertion we used to like asset.true(actual, expected) and second time we can use same assert.true(actual, expected, message).

6. How can you use interface and how it is different from Abstract class?

Abstract class may have Abstract and concrete methods, and there is not any compulsion in adding abstract method in abstract class. But in Interface, we do have only abstract methods and we don't need to write abstract keyword in Interface this is by default public and abstract.

7. What do you mean by Static keyword in Java?

Static means it is at class level not at instance level, we have static method, static variable & static inner class. When we have any variable as static so it will remain same for all the instance of our classes, and static/Private/Final methods can't be over-ridden like if we have initialized any method as Static so we cannot override it in any child class.

8. How to call static method and variable in java?

Direct calling, Calling by class name.

9. Can I access Static method by using object reference?

Yes we can, but we got one warning that you need to access it via Direct or By class name.

10. How to call non-static method and variable in java?

For calling non static method we need to create object first.

11. What do you mean by wrapper class and how will you do data conversion?

Wrapper class in java are used for data conversion. In data conversion if user wants to convert Int to string, String to int, Boolean, double then we use Wrapper class.

integer.parseInt(); -To convert string to Integer

Double.parseDouble(); -To convert string to Double

Boolean.parse Boolean(); -To convert string to Boolean

String.valueof(); -To convert Integer to String

12. Can you convert string a =”110a” in integer?

No we got NumberFormatException while converting the above string.

13. What do you mean by Call by Value & Call by Reference in Java?

Call by value means suppose we have created one sum method with input parameter int a, int b. So while calling the creating the object and running we provide values that is know as call by value.

14. What do you mean by Exceptions in Java?

Exception is like any interruption in our normal flow. Like if we are running anything and we got issues in our script this is we called exception,

we have 2 types of exception Run Time & Compile Time.(checked & Unchecked exceptions)

15. Can you tell me about difference between Throw and Throws keyword?

Throw is a keyword used inside a body of function. And Throws used while initializing any method. By using Throw we can throw only one exception while for Throws we can declare multiple exceptions which might occur in that particular function.

Java throw keyword is used to explicitly throw an exception. Java throws keyword is used to declare an exception..

16. How much you rate yourself in selenium out of 5?

Out of 5 I will rate myself 3.5 in selenium.

17. Which locator you are using in your framework and why?

Mostly we used ID and Xpath because Id is the fastest and unique one and after that we prefer Xpath. Anyways we have other locators as well like css, class name, tag name, Link text, Partial Link text.

18. What is the difference between findelement & findelements?

findelement will give the first appearance of that element which matches our locator, whereas findelements will give us list of all the elements which is present over the webpage and matching our locator.

And if we don't find the element findelement will give us nosuchelementexception whereas findelements will return NULL/Empty list.

19. Can you tell me how you will handle multiple window in selenium?

We have windowhandle & windowhandles function for handling Multiple windows. Windowhandle will give the string value of only the active window that is open whereas windowhandles will give set of all the windows that are open in browser.

For More :

https://www.browserstack.com/guide/handle-multiple-win dows-in-selenium

20. How you will move from one window to another?

First we will check what all windows are open by using driver.getwindowhandles, to get set of opened windows , then I use iterator to iterate over each of the pages and inside for loop will check like Current URL matches with the excepted page, if match then switch to that window by using driver.switchTo(Destination window) -> to return back to main parent window use driver.defaultwindow

21. Tell me the difference between Implicit & Explicit wait?

Implicit wait applies for all the elements and all the tests like if we give 10 sec of implicit wait it will wait for 10 sec for each element before giving nosuchelement exceptions.

While Explicit wait can be applied for any particular step for which you want extra wait time so we can use explicit wait.

We can use mix of both waits to depend on the situation of the step

Link : https://www.guru99.com/implicit-explicit-waits-selenium.html

22. Can you tell me some exceptions in selenium?

NoSuchElementException

NoSuchWindowException

NoSuchframeException

StaleElementReferenceException

TimeoutException.

Link :

https://www.guru99.com/exception-handling-seleniu m.html

23. What do you mean by User Defined Exception?

User Defined Exception or custom exception is creating your own exception class and throws that exception using 'throw' keyword.

This can be done by extending the class Exception.

... The keyword “throw” is used to create a new Exception and throw it to the catch block

24. Can you tell me what is assert in TestNG?

Assert is like verification where we check like expected thing and actual thing are same or not.

25. Which assert you have used in TestNg?

We have used Hard assert and Soft assert, while applying Hard assert if we found any glitch in expected and actual then it will through exception and move to next @test

while Soft assert it won't give exception and move to next step of that test. And to get all the exceptions in console we need to write at the end assert.all().

26. Can you tell me about the order of TestNG annotations?

@BeforeSuite

@BeforeTest

@BeforeClass

@BeforeMethod

@Test

@AfterMethod

@AfterClass

@AfterTest

@AfterSuite

27. Do you heard about Priority in TestNg can we set -ve priority?

Yes, like priority is 0, -1, TestNg will run -1 then 0

then 1.

And if we have any @test which is not having any priority set, then in that case it will search via alphabetic order whichever comes first and execute test respectively.

28. Can you explain me TestNG?

TestNG is advanced version of Junit only. It is mainly used by Dev/QA for maintain the code easily and for unit testing.

It provides lots of benefits to us like we can create a suite and we can write all the required Tc in one go only using that suite. We can group our Tc we can set priority we can run our tc in parallel mode, We can generate good reports via TestNG.

We can write functionality depends on methods, depends on group. We can run single tc multiple time with single set of data or multiple set of Data.

29. How to run single method multiple time in TestNG?

We have invocation count attribute in @test annotiation. We can write invocation count as 3 if we want to run it 3 times. Apart from that we can write threadpull.size if we want to run that case in multiple thread.

Links :

https://www.inviul.com/run-same-test-multiple-times/

30. Do you work in cucumber, can you tell me what all files required in cucumber?

In cucumber we have Feature file, Step Definition file and Test Runner file.

In feature file we used to write scenario in gherkin language which is most like in plain English language. Here we use some of the keywords like feature, scenario, scenario outline, given, when, then, and, example, background keywords for writing our test scenarios steps. In Step Definition file we write mapping code for all the scenario of feature file. In test Runner file we provide the address of the feature file, step definition file, and all-important Tags, Plugin, Listeners in that.

31. Have you used GIT in your project can you explain about it?

Yes I have used GIT, It is a version control tool. Where we can maintain our central repo. we used to manage our code via GIT only.

We use Git to maintain our project in our local system. So, if someone like to work on that project I need to send complete update copy to him and after that he can work on that.

There are chances that single project is handled by multiple teams across the globe. So, it will be difficult if we won't use GIT.

32. Can you give me some GIT commands which you used on daily basis?

Git status- which shows status of all the files,if we have some files which is not yet added to our repo so it will give us untracked file.

After that we can use GIT add command after adding it will added to particular index and we can commit this file using Git Commit -m(Message) we can commit this untracked file. Also we have Git Merge, Git Push, Git Pull, Git checkout in etc

33. How to solve Merge conflict in GIT?

As we are only 2 tester working on this project, if we have any merge conflict I used to pull all the latest file/scripts to my local system.

Then I will analyze the difference between that particular file and merge file. After that I will check with my team member whether all his imp things are covered then I will add my steps and push the script to the central repo.

34. You have worked in Jenkins can you tell me how you have created jobs in Jenkins?

We have separate Dev-Ops Team to create Jenkins jobs at broad level but we also have access to jenkins, so we have created jobs for our internal purpose.

For creating any job we have click on create new job->inside that give name of your job->select freestyle project->then add. Beside that we can provide description of our project and in source code management we can choose Git-> provide repo url ->after that provide some schedule if you want to run the job on any specific schedule time.-> select window batch command-file location-save-click on build now for running. After triggering we can check log in console

35. What is the difference between Smoke & Sanity Testing?

Smoke and Sanity we can think like a same thing because both are checking important functionality.

Smoke testing is done on first stable build from developer to check like whether it is stable enough to move further or not.

While Sanity testing is subset of regression test which we perform on stable build and here also we used to check all the imp functionality.

36. What is Agile ceremony?

We have 4 Agile ceremony(events that occur during a Scrum sprint) -

- Sprint planning
- Sprint review
- Sprint Retrospective
- Daily scrum meeting.

37. Why the main method is static?

Java main() method is always static, so that compiler can call it without the creation of an object or before the creation of an object of the class.

Static method of a class can be called by using the class name only without creating an object of a class.

38. What is Run time polymorphism?

Run-Time Polymorphism: Whenever an object is bound with the functionality at run time, this is known as runtime polymorphism.

The runtime polymorphism can be achieved by method overriding. Java virtual machine determines the proper method to call at the runtime, not at the compile time.

39. Difference between list and set?

The main difference between List and Set is that Set is unordered and contains different elements, whereas the list is ordered and can contain the same elements in it.

For more :

https://www.geeksforgeeks.org/difference-between-list-a nd-set-in-java/

40. Method overloading and overriding?

Method overriding is used to provide the specific implementation of the method that is already provided by its super class.

Method overloading is performed within class.

Method overriding occurs in two classes that have IS-A (inheritance) relationship.

In case of method overloading, parameter must be different

Link :

https://www.javatpoint.com/method-overloading-vs-method-ove rriding-in-java

41. Use of constructor?

The purpose of constructor is to initialize the object of a class while the purpose of a method is to perform a task by executing java code.

Constructors cannot be abstract, final, static and synchronised while methods can be. Constructors do not have return types while methods do.

42. Difference between static and non-static methods?

Static method uses complie time binding or early binding.

Non-static method uses run time binding or dynamic binding.

A static method cannot be overridden being compile time binding. A non-static method can be overridden being dynamic binding.

Links :

https://www.geeksforgeeks.org/difference-between-static-and- non-static-method-in-java/

43. What is a super keyword in java?

The super keyword refers to superclass (parent) objects. It is used to call superclass methods, and to access the superclass constructor.

The most common use of the super keyword is to eliminate the confusion between superclasses and subclasses that have methods with the same name.

44. Difference between break and continue statement?

Break statement resumes the control of the program to the end of loop and made executional flow outside that loop.

Continue statement resumes the control of the program to the next iteration of that loop enclosing 'continue' and made executional flow inside the loop again

45. Difference between this and super?

this keyword mainly represents the current instance of a class.

On other hand super keyword represents the current instance of a parent class.

this keyword used to call default constructor of the same class.

46. What is the difference between length and length() in Java?

array.length: length is a final variable applicable for arrays. With the help of the length variable, we can obtain the size of the array.

string.length() : length() method is a final variable which is applicable for string objects. The length() method returns the number of characters present in the string.

47. Types of the assertion in selenium?

Selenium Assertions can be of three types: “assert”, “verify”, and ” waitFor”.

When an “assert” fails, the test is aborted. When a “verify” fails, the test will continue execution, logging the failure. A “waitFor” command waits for some condition to become true.

48. Have you used the action class and where it is used?

Actions class is an ability provided by Selenium for handling keyboard and mouse events.

```
Actions action = new Actions(driver);
action.moveToElement(element).click().perform();
```

The perform() method is used to perform the series of actions that are defined.

Link : https://www.browserstack.com/guide/action-class-in-selenium

49. What is the difference between checked and unchecked exceptions?

There are two types of exceptions: checked exception and unchecked exception.

The main difference between checked and unchecked exception is that the checked exceptions are checked at compile-time while unchecked exceptions are checked at runtime.

checked exceptions –

SQLException,IOException,ClassNotFoundException,InvocationTargetException

unchecked exceptions –
NullPointerException,ArrayIndexOutOfBoundsException,ArithmeticException,IllegalArgumentException NumberFormatException

50. Apart from sendkeys, are there any different ways, to type content onto the editable field?

WebDriver driver = new FirefoxDriver();

JavascriptExecutor executor = (JavascriptExecutor)driver;

executor.executeScript("document.getElementBy Id("textbox_id").value='new value';);

51. Annotations in Cucumber?

Total 11 Annotations - Feature, Scenario, Background, given, when , then, and, but, example, scenario outline, scenario template.

52. What are hashmap and HashSet? Explain??

HashMap and HashSet both are one of the most important classes of Java Collection framework. ... HashMap Stores elements in form of key-value pair i.e each element has its corresponding key which is required for its retrieval during iteration. HashSet stores only objects no such key value pairs maintained.

Links :

https://www.geeksforgeeks.org/difference-between-hashmap-and-hashset/
https://www.javatpoint.com/difference-between-hashset-and-hashmap

53. Where do you use a hashmap??

Maps are used for when you want to associate a key with a value and Lists are an ordered collection. Map is an interface in the Java Collection Framework and a HashMap is one implementation of the Map interface.

HashMap are efficient for locating a value based on a key and inserting and deleting values based on a key. HashMap map = new HashMap<>();.

Example :

```
// Add elements to the map

map.put("vishal", 10);

map.put("sachin", 30);

map.put("vaibhav", 20);

// Print size and content

System.out.println("Size of map is:- "+ map.size()); System.out.println(map);

// Check if a key is present and if present, print value

if (map.containsKey("vishal")) { Integer a = map.get("vishal");
System.out.println("value for key"+ " \"vishal\" is:- " + a);
```

54. How do you handle if XPath is changing dynamically?

Option 1: Look for any other attribute which Is not changing every time In that div node like name, class etc. So If this div node has class attribute then we can write xpath as bellow.

```
//div[@class='post-body entry-content']/div[1]/form[1]/input[1]
```

Option 2: We can use absolute xpath (full xpath) where you do not need to give any attribute names In xpath.

```
/html/body/div[3]/div[2]/div[2]/div[2]/div[2]/div[2]/div/div[4]/div[1]/div/div/div/div[1]/div/div/div/div  [1]/div[2]/div[1]/form[1]/input[1]
```

Option 3: We can use starts-with function. In this xpath's ID attribute, "post-body-" part remains same every time. //div[starts-with(@id,'post-body-')]/div[1]/form[1]/input[1]

Option 4: We can use contains function. Same way you can use contains function as Above

```
//div[contains(@id,'post-body-')]/div[1]/form[1]/input[1]
```

55. Does Jenkins require a local system for CI??

It is a server-based application and requires a web server like Apache Tomcat

56. When finally block get executed? ?

The finally block always executes when the try block exits. This ensures that the finally block is executed even if an unexpected exception occurs.

57. How many times you can write catch block?

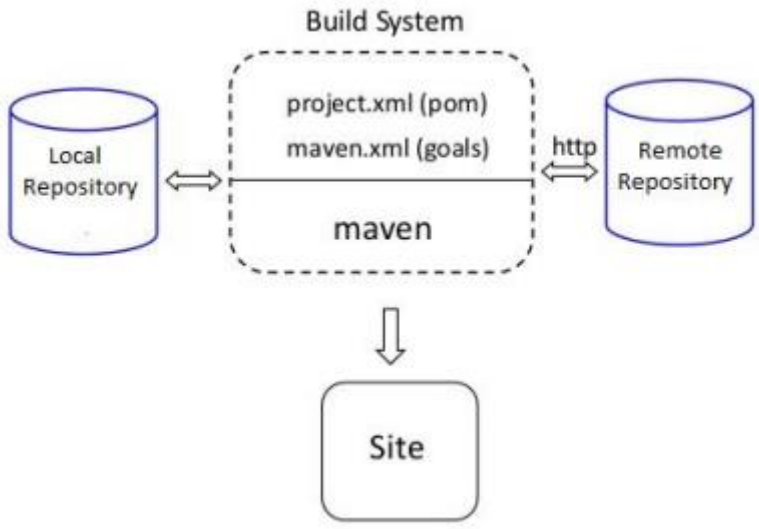
maximum one catch block will be executed.

Yes, we can write multiple catch block but only one

is executed at a time.

58. What Maven Architecture and explain pom.xml?

POM is an acronym for Project Object Model. The pom.xml file contains information of project and configuration information for the maven to build the project such as dependencies, build directory, source directory, test source directory, plugin, goals etc. Maven reads the pom.



59. How you handle alert in selenium webdriver?

Simple alert(one option), Confirm Alert(Y/N), Prompt alert(enter any value)

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

- a.getText();
- a.accept();
- a.dismiss();
- a.sendKeys("name");

60. How to handle iframes in selenium webdriver?

- driver.switchTo().frames(via index value, name, webelement);
- driver.findElement(by.id("value")).getText();
- driver.switchTo().defaultContent();-To get back from iframe

61. How many types of WebDriver API 's are available in selenium?

- Chrome
- Geko
- Chromium
- Edge
- Html
- android

62. What are the different exception you faced in selenium webdriver?

Webdriver exc, noalertpresent exc, nosuchwindow exc, nosuchelement exc, timeout exc.

63. How do you scroll down a page using javascript in selenium?

window.scrollby(,) function E.g:

js.executeScript("window.scrollBy(0,1000)"); //Scroll vertically down by 1000 pixels

Link:

https://www.guru99.com/scroll-up-down-selenium-webdriver.html

64. How do you scroll down to a particular element?

//This will scroll the page till the element is found

js.executeScript("arguments[0].scrollIntoView();", Element);

65. Which all files can be used as a data source for different frameworks?

.csv, .xml, .text etc

66. What are listeners in selenium?

Listener is defined as interface that modifies the default TestNG's behavior.

As the name suggests Listeners "listen" to the event defined in the selenium script and behave accordingly.

It is used in selenium by implementing Listeners Interface. It allows customizing TestNG reports or logs.

There are two types of Selenium Listeners:

WebDriver Listeners

TestNG Listeners

Link: https://www.guru99.com/listeners-selenium-webdriver.html

67. How do you take screenshots in selenium webdriver?

TakesScreenshot scrShot = ((TakesScreenshot)webdriver);

File SrcFile = scrShot.getScreenshotAs(OutputType.FILE); Link :

https://www.guru99.com/take-screenshot-selenium-webdriver. html

68. What do you mean by WebDriver?

Webdriver is an interface which is used to automate api of browser for testing.

69. How you handle dropdown values?

From select class,

via visible text, value, index.

70. How to handle hidden elements in selenium webdriver?

JavascriptExecuter js = (JavascriptExecutor)driver;

js.excuteScript("document.getElementById("<>").val ue='Hiddentext);

71. What does means Public static void main(variable,value) ?

Public/private/protected/default- Access specifier

Static- modifier

Void- return type

Main-class name

72. What are the open source frameworks supported by selenium webdriver?

TestNG, Junit, Cucumber, Robot Framework, Appium, Protractor

73. How to get color of webelement using selenium webdriver?

First get the locator of webElement , then get

String color = object.getCssValue("background-color");

String HexbackColor= color.fromString(color).asHex();

It will give you RGB codes , you need to convert them into back color using HEX function.

74. How to reverse an Integer?

```
int num = 12345;
int rev = 0; while(num !=0){
rev =rev *10+ num % 10;
num = num/10;
}
Sysout (rev)
}
```

75. How to reverse a string?

```
String str = "Jaikishan";
int len = str.length();
String rev = " ";
for(int i=len-1 ; i>=0 ; i--){
rev = rev + str.charAt(i);
}
Sysout(rev);
```

76. How will you print length of string without using length method.

```
String str = "Jaikishan"
1. int i = 0;
for(char c: str.toCharArray()) { i++;
```

}

System.out.println("Length of the given string ::"+i);

OR

2. Sysout(str.toCharArray().length); Sysout(str.lastIndexOf(“”));

Link: https://java2blog.com/find-length-of-string-without-using/

77. How to swap two numbers without using a third variable?

public static void swapNumbers(int a, int b) { b = b + a;

a = b - a;

b = b - a;

}

78. Write down syntax of iterator function?

Iterator<String> it = studentList.iterator(); while(it.hasNext()){
System.out.println(it.next());

}

79. How to reverse any array?

public class reverse array {

public static void main(String[] args) {

int [] Array ={7,8,9,3,4,6,11,67,98};

for(int k=Array.length-1 ; k>=0 ; k--) {

System.out.print(Array[k] + " ");

}

} }

80. How to find additional element in list while comparing 2 List?

If we have 2 list l1 & l2 , first we remove all element of l2

L1.removeAll(l2);

Sysout(L1) – you will get additional element.

STRING Post-mortem

String: String is a sequence of character in java which can be defined as below:

String str="Jaikishan";

Char ch[] = {'J','a','i','k','i','s','h','a','n'};

String.valueOf(ch);

java.lang.String class is used to create a string object.

Different String methods:

- Int compareTo() - The Java String compareTo() method is used for comparing two strings lexicographically.
- boolean equals() - The java string equals() method compares the two given strings based on the content of the string (case sensitive)
- String concat() – concat two strings
- boolean equalsIgnoreCase() - The java string equalsIgnoreCase() method compares the two given strings based on the content of the string (not case sensitive)
- char charAt() – index position - The java string charAt() method returns a char value at the given index number.
- boolean contains() - true if the sequence of char value exists, otherwise false.
- toUpperCase() – convert to upper case
- toLowerCase() – convert to lower case
- trim() – remove spaces from both sides of string
- substring() - returns part of string
- boolean endsWith() - ends with specified suffix or not
- boolean startWith() - start with specified prefix or not

- int length() - the total number of characters present in the string.
- replace() - returns a string replacing all the old char or CharSequence to new char or CharSequence.
- int num = Integer.parseInt(str); - Convert String to int using Integer.parseInt(String)
- int num = Integer.valueOf(str); - Convert String to int using Integer.valueOf(String)
- Convert int to String using String.valueOf()

  String int ivar = 123;

  ```
  String str = String.valueOf(ivar);

  System.out.println("String is: "+str);

  System.out.println(555+str);
  ```

- Convert int to String using Integer.toString()

  int ivar = 123;

  ```
  String str = Integer.toString(ivar);

  System.out.println("String is: "+str);

  System.out.println(555+str);
  ```

★ In java, string objects are immutable. Immutable simply means unmodified or unchangeable.Once string object is created its data or state can't be changed but a new string object is created.

Selenium Commands

#1) get() Methods

- driver.get("https://google.com"); -- used to open an URL and it will wait till the whole page gets loaded.
- driver.getClass(); -- The command is used to retrieve the Class object that represents the runtime class of this object
- driver.getCurrentUrl(); -- This command returns the URL of the currently active web page in the browser.
- driver.getPageSource(); -- This command helps in getting the entire HTML source code of the open web page.
- driver.getTitle(); -- This command can be used for displaying the title of the current web page.
- driver.getText(); -- delivers the innerText of a WebElement.
- driver.findElement(By.id("findID")).getAttribute("va lue"); -- used to retrieve the value of the specified attribute
- driver.getWindowHandle(); -- used to tackle with the situation when we have more than one window to deal with.

#2) Locating links by linkText() and partialLinkText()

- driver.findElement(By.linkText("jaikishan")).click();

  finds the element using link text

- driver.findElement(By.partialLinkText("jai")).click();

  find the elements based on the substring of the link

#3) Selecting multiple items in a drop dropdown

// select the multiple values from a dropdown

- Select select = new Select(driver.findElement(By.id("Id_in_select_class"

```
));
```

- select.selectByValue("greenvalue"); - By Value
- select.selectByVisibleText("Red"); - By Visible Text
- select.selectByIndex(2); - By Index

#4) Submitting a form

driver.findElement(By.id("submit")).submit();

```
driver.get("https://www.facebook.com/");
driver.findElement(By.id("email")).sendKeys("abc@gmail.com");
driver.findElement(By.id("pass")).sendKeys("123456");
driver.findElement(By.id("pass")).submit(); // submitting form with submit()
```

OR

```
driver.findElement(By.name("login")).click(); // submitting form with click()
```

#5) Handling iframes

- Select iframe by id -- driver.switchTo().frame("ID of the frame");
- Locating iframe using tagName -- driver.switchTo().frame(driver.findElements(By.tagName("iframe").get(0));
- Locating iframe using the index: -- driver.switchTo().frame(0);
- Locating by Name of iframe -- driver.switchTo().frame("name of the frame");
- Select Parent Window -- driver.switchTo().defaultContent(); •

#6) close() and quit() methods

- driver.close(); -- closes only a single window that is being accessed by the WebDriver instance currently
- driver.quit(); -- closes all the windows that were opened by the WebDriver instance

#7) Exception Handling

Exceptions are the conditions or situations that halt the program execution unexpectedly.

Reasons for such conditions can be:

- Errors introduced by the user
- Errors generated by the programmer
- Errors generated by physical resources

```
WebElement saveButton = driver.findElement(By.id("Save"));
try{ if(saveButton.isDisplayed()){
saveButton.click();} }
catch(NoSuchElementException e)
{ e.printStackTrace(); }
```

Other useful Commands :

- isEnabled() -- to Check Whether the Element is Enabled Or Disabled in the Selenium WebDriver.

  ```
  boolean textBox =
  driver.findElement(By.xpath("//input[@name='textbox1']")).isEnabled();
  ```

- pageLoadTimeout(time,unit) -- to set the time for a page to load.

  ```
  driver.manage().timeouts().pageLoadTimeout(500, SECONDS);
  ```

- implicitlyWait() -- to set a wait time before searching and locating a web element.

  driver.manage().timeouts().implicitlyWait(1000, TimeUnit.SECONDS);.

- until() and visibilityOfElementLocated() -- until() from WebdriverWait and visibilityOfElementLocated() from ExpectedConditions to wait explicitly till an element is visible in the webpage.

  WebDriverWait wait = new WebDriverWait(driver, 10);

  WebElement element = wait.until(ExpectedConditions.visibilityOfElementLocated (By.xpath("//input[@id='name']")));

- untill() and alertIsPresent() -- untill() from WebdriverWait and alertIsPresent() from ExpectedConditions to wait explicitly till an alert appears.

  WebDriverWait wait = new WebDriverWait(driver, 10);

  WebElement element = wait.until(ExpectedConditions.alertIsPresent());

#8) Select

WebElement mySelectedElement = driver.findElement(By.id("select")); Select dropdown= new Select(mySelectedElement);

- dropdown.selectByVisibleText("Jaikishan");
- dropdown.selectByValue("Fav_course");
- dropdown.selectByIndex(1);
- dropdown.deselectByVisibleText("Jaikishan");
- dropdown.deselectByValue("Fav_course");
- dropdown.deselectByIndex(1);

#9) navigate() methods

navigate() -- to navigate between the URLs.

- driver.navigate().to("https://www.mohantyacadem y.com");
- driver.navigate().back();
- driver.navigate().forward();
- driver.navigate().refresh();

#10) getScreenshotAs() methods

getScreenshotAs() -- to Capture the entire page screenshot in Selenium WebDriver.

- File shot = ((TakesScreenshot)driver).getScreenshotAs(OutputT ype.FILE);
- FileUtils.copyFile(shot, new File("D:\\criticalbug.jpg"));

Other useful Commands :

- moveToElement()

Actions actions = new Actions(driver); WebElement mouseHover =

driver.findElement(By.xpath("//div[@id='bestTutor']/div"));
actions.moveToElement(mouseHover);

actions.perform();

- dragAndDrop()

actions.dragAndDrop(sourceLocator, destinationLocator).build().perform();

- switchTo() and accept(), dismiss() and sendKeys() -- methods from Alert class to switch to popup alerts and handle them.

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

alert.sendKeys("Don't forget to give 5 star rating");

alert.accept();

★ alert.dismiss() can be used to dismiss the alert.

- assertEquals(),assertNotEquals(), assertTrue() and assertFalse() -- Assertions are used to compare the expected and actual results. Pass or fail of a test is usually decided from the result of assertions.

```
Assert.assertEquals(message, “This text”);

Assert.assertNotEquals(message, “This text”);

Assert.assertTrue(result<0);

Assert.assertFalse(result<0);
```