**Software Testing Assignment**

**Advanced Selenium**

# Assignments Module – 6 (Core Java)

* W.A.J.P to Take three numbers from the user and print the greatest number.

|  |
| --- |
| **package** com.assignment1;  **import** java.util.Scanner;  **public** **class** FindGreatestNumber  {  **public** **static** **void** main(String[] args)  {  Scanner sc=**new** Scanner(System.***in***);  System.***out***.println("Enter the first number... ");  **int** a=sc.nextInt();  System.***out***.println("Enter the Second number... ");  **int** b=sc.nextInt();  System.***out***.println("Enter the Third number... ");  **int** c=sc.nextInt();    **if**(a>b && a>c)  {  System.***out***.println(a+" is the greatest number");  }  **else** **if**(b>a && b>c)  {  System.***out***.println(b+" is the greatest number");  }  **else**  {  System.***out***.println(c+" is the greatest number");  }  }  } |

* W.A.J.P in Java to display the first 10 natural numbers using while loop.

|  |
| --- |
| **package** com.assignment1;  **public** **class** FirstTenNaturalNumber  {  **public** **static** **void** main(String[] args)  {  **int** i=0;  **while** (i<10)  {  System.***out***.println(i++);  }    }  } |

* W.A.J.P to find factorial for Given Number.

|  |
| --- |
| **package** com.assignment1;  **import** java.util.Scanner;  **public** **class** FactorialNumber  {  **public** **static** **void** main(String[] args)  {  Scanner sc=**new** Scanner(System.***in***);  **int** x=sc.nextInt();  **int** fact=1;  **for** (**int** i=x;i>=1;i--)  {  fact=fact\*i;  }  System.***out***.println("Factorial of "+x+" is "+fact);  sc.close();  }  } |

* W.A.J.P to check given number is Prime or not?

|  |
| --- |
| **package** com.assignment1;  **import** java.util.Scanner;  **public** **class** PrimeOrNot  {  **public** **static** **void** main(String[] args)  {  Scanner sc = **new** Scanner(System.***in***);  **int** a = sc.nextInt();  **boolean** prime = **true**;  **for** (**int** i = a - 1; i > 1; i--)  {  **if** (a % i == 0)  {  prime = **false**;  **break**;  }  }  **if** (prime == **true**)  {  System.***out***.println("This number is Prime number");  } **else**  {  System.***out***.println("This number is not a Prime number");  }  sc.close();  }  } |

* W.A.J.P to check given number is Armstrong or not?

|  |
| --- |
| **package** com.assignment1;  **import** java.util.Scanner;  **public** **class** ArmstrongOrNot  {  **public** **static** **void** main(String[] args)  {  Scanner sc=**new** Scanner(System.***in***);  **int** x=sc.nextInt();  **int** sum=0;  **int** copy= x;  **int** size = Integer.*toString*(copy).length();  **while**(copy!=0)  {  **int** rem = copy % 10;  **int** mul = 1;  **for**(**int** i=0;i<size;i++)  {  mul=mul\*rem;  }  sum = sum + mul;  copy=copy/10;  }  **if**(sum==x)  {  System.***out***.println("It is an Armstrong number");  }  **else**  {  System.***out***.println("It is not an armstrong number");  }      }  } |

* W.A.J.P for create Fibonacci Series.

|  |
| --- |
| **package** com.assignment1;  **import** java.util.Scanner;  **public** **class** FibonacciSeries  {  **public** **static** **void** main(String[] args)  {  Scanner sc=**new** Scanner(System.***in***);  **int** n, a=0,b=0,c=1;  System.***out***.println("Enter value of n: ");  n=sc.nextInt();  System.***out***.println("Fibonacci Series: ");  **for**(**int** i=1;i<=n;i++)  {  a=b;  b=c;  c=a+b;  System.***out***.println(a+" ");  }  }  } |

* W.A.J.P to Print pattern Given Below.

1)

1

12

123

1234

12345

|  |
| --- |
| **public** **class** Pyramid1Demo  {  **public** **static** **void** main(String[] args)  {  **for**(**int** i=1;i<=5;i++)//row  {  **for**(**int** j=1;j<=i;j++)//column  {  System.***out***.print(j);  }  System.***out***.println();  }  }  } |

2)

1

12

123

1234

12345

3)

1

01

101

01010

101010

|  |
| --- |
| **public** **class** Pyramid6Demo  {  **public** **static** **void** main(String[] args)  {  **for**(**int** i=5;i>=1;i--)  {  **for**(**int** j=i;j<=5;j++)  {  System.***out***.print(j%2);  }  System.***out***.println();  }  }  } |

4)

1

1. 2
2. 3 3
3. 4 4 4

|  |
| --- |
| **package** com.assignment1;  **public** **class** Pyramid0  {  **public** **static** **void** main(String[] args)  {    **for**(**int** i=1;i<=4;i++)  {  **for**(**int** j=1;j<=i;j++)  {  System.***out***.print(i+" ");  }  System.***out***.println();  }  }  } |

5)

\*

* + \* \*
  + \* \* \* \*
  + \* \*

\*

|  |
| --- |
| **package** com.assignment1;  **public** **class** Pyramid1  {  **public** **static** **void** main(String[] args)  {  **for**(**int** i=1;i<=3;i++)  {  **for** (**int** j=1;j<=3-i;j++)  {  System.***out***.print(" ");  }  **for**(**int** k=1;k<=2\*i-1;k++)  {  System.***out***.print("\*");  }  System.***out***.println();  }  **for** (**int** i=3;i>=1;i--)  {  **for** (**int** j=1;j<=3-i;j++)  {  System.***out***.print(" ");  }  **for** (**int** k=1;k<=2\*i-1;k++)  {  System.***out***.print("\*");  }  System.***out***.println();    }  }    } |

* WAP to compute the sum of the first 100 prime numbers.

A)

|  |
| --- |
| **package** com.assignment1;  **public** **class** SumOfPrimeNumbers  {  **public** **static** **void** main(String args[])  {  **int** number = 1, count, sum = 0;  **while**(number <= 100)  {  count = 0;  **int** i = 2;  **while**(i <= number/2 )  {  **if**(number % i == 0)  {  count++;  **break**;  }  i++;  }  **if**(count == 0 && number != 1 )  {  sum = sum + number;  }  number++;  }  System.***out***.println("The Sum of Prime Numbers from 1 to 100 is: " + sum);  }  } |

* WAP to sum values of an array.

A)

|  |
| --- |
| **public** **class** SumOfArray  {  **public** **static** **void** main(String[] args)  {    **int** [] arr = **new** **int** [] {1, 2, 3, 4, 5};  **int** sum = 0;  **for** (**int** i = 0; i < arr.length; i++)  {  sum = sum + arr[i];  }  System.***out***.println("Sum of all the elements of an array: " + sum);  }  } |

* WAP to calculate the average value of array elements.

A)

|  |
| --- |
| **public** **class** AverageOfArray  {  **public** **static** **void** main(String[] args)  {  **int**[] numbers = **new** **int**[]{20, 30, 25, 35, -16, 60, -100};  **int** sum = 0;  **for** (**int** i = 0; i < numbers.length; i++)  {  sum = sum + numbers[i];  }  **double** average = sum / numbers.length;  System.***out***.println("Average value of the array elements is : " + average);  }  } |

* WAP to find the index of an array element.

A)

|  |
| --- |
| **import** java.util.\*;  **public** **class** index  {  **public** **static** **int** findIndex(**int** arr[], **int** t)  {  **if** (arr == **null**)  {  **return** -1;  }  **int** len = arr.length;  **int** i = 0;  **while** (i < len)  {  **if** (arr[i] == t)  {  **return** i;  }  **else**  {  i = i + 1;  }  }  **return** -1;  }  **public** **static** **void** main(String[] args)  {  **int**[] my\_array = { 5, 4, 6, 1, 3, 2, 7, 8, 9 };  System.***out***.println("Index position of 5 is: "  + *findIndex*(my\_array, 5));  System.***out***.println("Index position of 7 is: "  + *findIndex*(my\_array, 7));  }  } |

* WAP to find the maximum and minimum value of an array.

A)

|  |
| --- |
| **public** **class** FindMaxMin  {  **public** **static** **void** main(String[] args)  {  **int**[] array = {5, 12, 9, 18, 3, 21};  **int** max = array[0];  **int** min = array[0];  **for** (**int** i = 1; i < array.length; i++)  {  **if** (array[i] > max)  {  max = array[i];  }  **if** (array[i] < min)  {  min = array[i];  }  }  System.***out***.println("Maximum value: " + max);  System.***out***.println("Minimum value: " + min);  }  } |

* WAP to Compare Two String.

A)

|  |
| --- |
| **public** **class** StringComparison  {  **public** **static** **void** main(String args[])  {  String s1="Hetal";  String s2="Hetal";  String s3=**new** String("Hetal");  String s4="Het";  System.***out***.println(s1.equals(s2));//true  System.***out***.println(s1.equals(s3));//true  System.***out***.println(s1.equals(s4));//false  }  } |

* WAP to concatenate a given string to the end of another string.

A)

|  |
| --- |
| **public** **class** Concate  {  **public** **static** **void** main(String[] args)  {  String str1 = "JAVA Exercises and ";  String str2 = "Python Exercises";  System.***out***.println("String 1: " + str1);  System.***out***.println("String 2: " + str2);  String str3 = str1.concat(str2);  System.***out***.println("The concatenated string: " + str3);  }  } |

* WAP to demonstrate try catch block.

A)

|  |
| --- |
|  |

* WAP to demonstrate multiple catch blocks
* WAP to create one thread by implementing Runnable interface in Class.
* WAP to create one thread by extending Thread class in another

Class.

* WAP to iterate through all elements in an array list.
* WAP to update specific array element by given element.
* WAP to remove the third element from a array list.  WAP to Copy one array into another
* WAP to reverse an array of integer values.
* WAP to find the second largest element in an array.
* W.A.J.P. Create an abstract class 'Parent' with a method 'message'. It has two subclasses each having a method with the same name 'message' that prints "This is first subclass" and "This is second subclass" respectively.

Call the methods 'message' by creating an object for each subclass.

* W.A.J.P. which will ask the user to enter his/her marks (out of 100). Define a method that will display grades according to the marks entered as below:

|  |  |
| --- | --- |
| Marks | Grade |
| 91-100 | AA |
| 81-90 | AB |
| 71-80 | BB |
| 61-70 | BC |
| 51-60 | CD |
| 41-50 | DD |
| <=40 | Fail |

* W.A.J.P. to create a custom exception if Customer withdraw amount which is greater than account balance then program will show custom exception otherwise amount will deduct from account balance.

Account balance is:2000 Enter withdraw amount:2500

Sorry, insufficient balance, you need more 500 Rs. To perform this transaction.

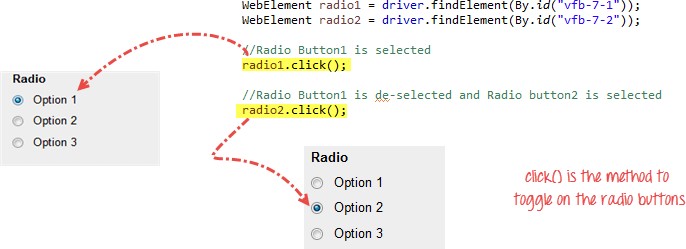
# Module – 7 (Selenium Webdriver)

**Demo Websites To Practice selenium Webdriver**

|  |  |
| --- | --- |
|  | **Website**  **List** |
|  | [1 List of Websites to Practice Selenium](https://www.techbeamers.com/websites-to-practice-selenium-webdriver-online/#list-ofwebsites-to-practice-selenium)    o 1.1 1. https://phptravels.com/demo/ o [1.2 2. http://thedemosite.co.uk/](https://www.techbeamers.com/websites-to-practice-selenium-webdriver-online/#2httpthedemositecouk) o [1.3 3. http://newtours.demoaut.com/](https://www.techbeamers.com/websites-to-practice-selenium-webdriver-online/#3httpnewtoursdemoautcom) o [1.4 4. http://the-internet.herokuapp.com/](https://www.techbeamers.com/websites-to-practice-selenium-webdriver-online/#4httpthe-internetherokuappcom) o [1.5 5. http://automationpractice.com/index.php](https://www.techbeamers.com/websites-to-practice-selenium-webdriver-online/#5httpautomationpracticecomindexphp) o [1.6 6. http://book.theautomatedtester.co.uk/](https://www.techbeamers.com/websites-to-practice-selenium-webdriver-online/#6httpbooktheautomatedtestercouk) o [1.7 7. https://s1.demo.opensourcecms.com/wordpress/](https://www.techbeamers.com/websites-to-practice-selenium-webdriver-online/#7httpss1demoopensourcecmscomwordpress) |

* W.A.J.Script for Locating links by linkText() and partialLinkText()

* W.A.J.Script for Selecting multiple items in a drop dropdown
* W.A.J. script to use different methods to manage the windows-alerts and pop ups.
* W.A.J.script to register your self in Gmail.
* W.A.J. Script To perform the radio button to select one by one in loop <http://demo.guru99.com/test/radio.html>

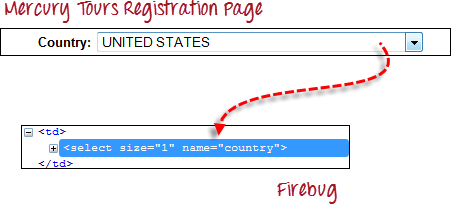


* W.A.J. script To write the script for image of logo facebook using xpath.

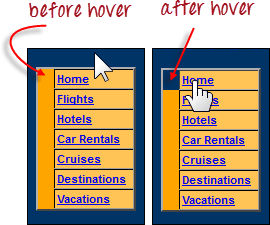


* W.A.J.Script To write a script for drop down.

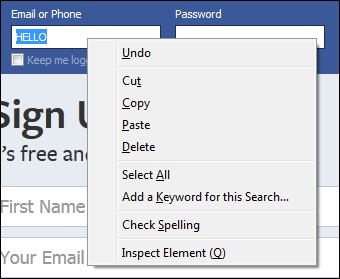
<http://demo.guru99.com/test/newtours/register.php>



* W.A.J.Script To use Mouse and Keyboard event using Action class 1.Mouse Hover Event

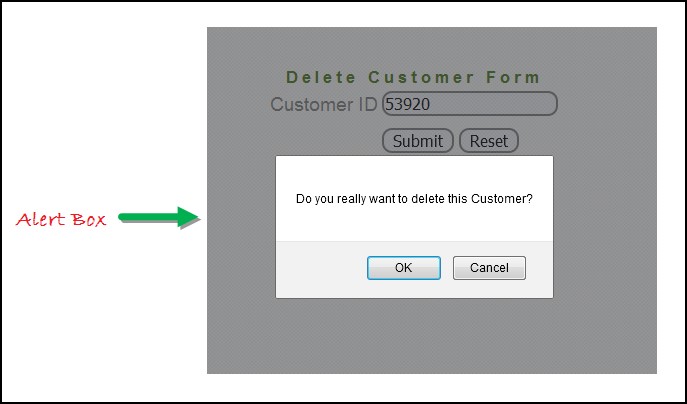


2. Keyboard event

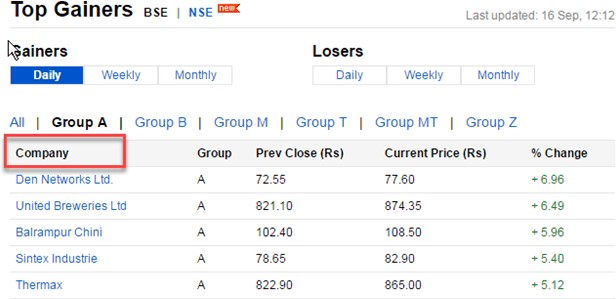


* W.A.J. Script How to handled Alert in selenium

["http://demo.guru99.com/test/delete\_customer.php "](http://demo.guru99.com/test/delete_customer.php)



* W.A.J. Script To find the total hyperlink from this web page <http://demo.guru99.com/test/web-table-element.php>



* W.a.junit program to handled Assert class with all method to check its pass or fail
* W.a. junit program to perform test with webdriver to login process of facebook
* W.a. junit program to check gmail login using with

@before,@after,@Test

* W.a. junit program to use parameterized demo with multiple parameter of Facebook login in junit.
* W.a. TestNG program to perform test with webdriver to login process of facebook
* W.a. TestNG program to check gmail login using with

@beforetest,@aftertest,@Test

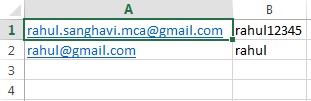
* W.a. TestNG program to use parameterized demo with multiple parameter of Facebook login with TestNG.
* W.a. TestNG program to create group with testing.xml file  w.a. TestNG program to create dataprovider.

# Module – 8 (Automation Testing Framework)

* W.a.framework program for data driven framework

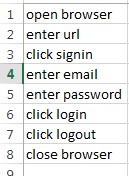
-to get value from the excel and check into your website

[(http://automationpractice.com/index.php?controller=authentication&b ack=myaccount)](http://automationpractice.com/index.php?controller=authentication&back=my-account) Direct check with login which emaild through login successful or not)



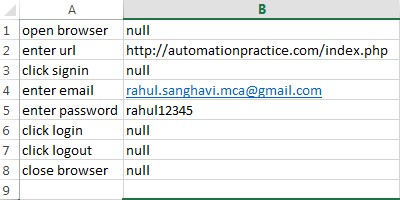
* W.a.framework program for keyword driven framework

-given below keyword which you get from the Excel and check into you website [(http://automationpractice.com/index.php)](http://automationpractice.com/index.php)



* W.a.framework program for hybrid driven framework

-given below keyword and Data both which you get from the Excel and check into you website [(http://automationpractice.com/index.php)](http://automationpractice.com/index.php)



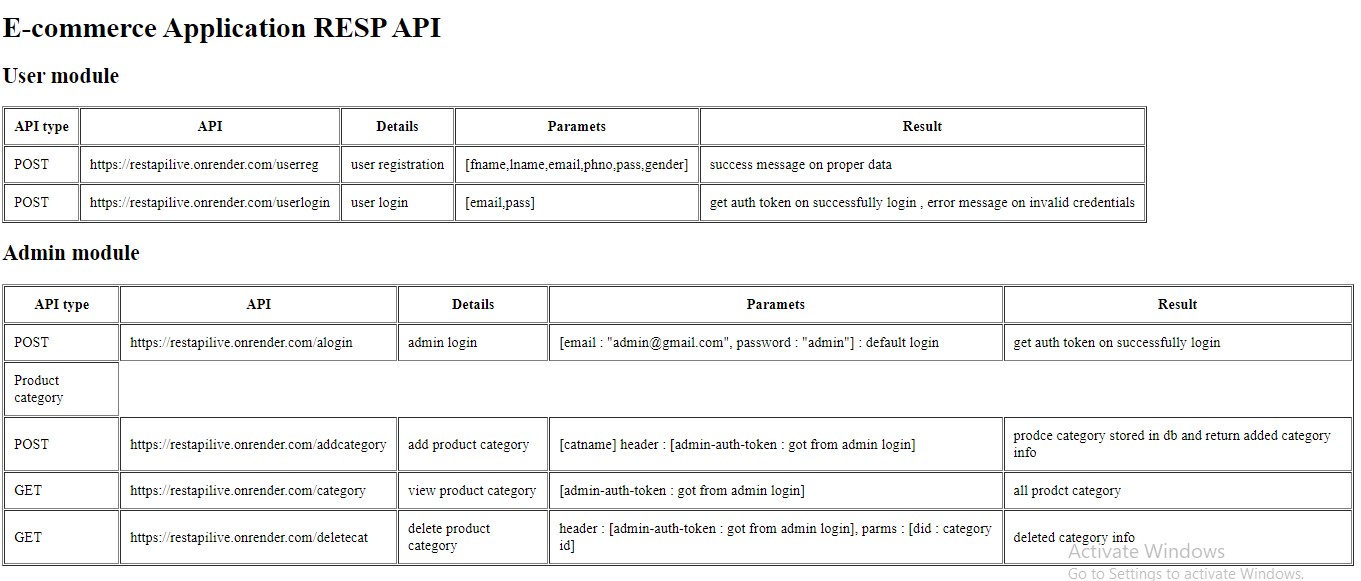
* W.a.maven program to create simple webdriver Program
* **W.a.maven program to create Junit with webdriver Program**
* W.a.maven program to create TestNG with Webdriver Program.

## Module – 9 (APPIUM Testing)

1. Write an Appium Program to connect with emulator and open APIdemo.app application on your emulator
2. Write an Appium Program to connect with Realdevice and open APIdemo.app application on your realdevice.
3. Write an Appium Program to connect with emulator wit ecommerce based application using Generalstore.app to perform locators like name, dropdown etc.
4. Write an Appium Program to connect with emulator wit ecommerce based application using Generalstore.app to perform swipe demo to swipe the menu.
5. Write an Appium Program to connect with realdevice using APIDemo.app to perform longpress to open to side menu.
6. Write an Appium Program to connect with realdevice using APIDemo.app to perform scrolling the all option.
7. Write an Appium Program to connect with realdevice using APIDemo.app to open the calculator and calculate all the operation like (addition, substraction, multiplication, division).

## Module – 10 (API Testing)

Api testing link : <https://restapilive.onrender.com/>



1. to test in above API which is passed in parameter as a column in user module to check the registration performed or not and get the result is coming or not (Method=POST)
2. to test in above API which is performed login wit using parameter in user module and check generate token or not (Method=POST)
3. to test in above API which is performed login wit using parameter in admin module and check generate token or not (Method=POST)
4. to test in above API which is performed login wit using parameter in admin module and check generate token header : [admin-auth-token : got from admin login] and add category with the help of token (Method= POST)
5. to test in above API which is performed login wit using parameter in admin module and check generate token [admin-auth-token : got from admin login] to display all the category (Method=GET)
6. to test in above API which is performed login wit using parameter in admin module and check generate token[admin-auth-token : got from admin login] to delete the category (Method=GET)