

Exp No 11: Working of JUnit to Reverse a Word and using Assert Statement

(20)

Aim: To understand the working of JUnit assert statements by comparing the reversed value with expected one.

Program:

```
import static org.junit.Assert.assertEquals;
import java.util.Scanner;

class SaveethaTest {

    public static void main(String[] args) {
        String str;
        char ch;
        Scanner sc = new Scanner(System.in);
        System.out.println("Reverse a string "+str+" is");
        for (int i = str.length() - 1; i >= 0; i--) {
            System.out.print(str.charAt(i));
            assertEquals("Rini", str);
        }
        assertEquals("Rini", str);
    }
}
```

Output :

Input	Actual output
Rini	inR

Testcases :

Test case no : 1

Test case name : Expected one Same as actual one

Input =	Expected output	Actual output	Remarks
Rini	inR	inR	Successes

Test case no : 2

Test case name : Expected one Same as actual one

Input =	Expected output	Actual output	Remarks
Rama	amar	a	Faliure

Expo: 12. White box Testing to String Comparison of Word (JUnit)

Aim: To understand the working of JUnit
assert statements by comparing two strings

Program:

```
import static org.junit.Assert.assertEquals;
import java.util.Scanner;

public class Word {
    public static void main (String[] args) {
        Scanner in = new Scanner(System.in);
        System.out.println ("enter user name");
        String str1 = in.nextLine();
        System.out.println ("Reenter user name");
        String str2 = in.nextLine();
        assertEquals (str1, str2);
    }
}
```

output:

Enter user name : Ame
Reenter user name : Ame
Equal

Enter user name : Ame
Reenter user name :
Amelia
Comparison failure.

Aim : To understand the working of JUnit statements by checking the Voting age

Program :

```
import static org.junit.Assert.*;
import java.util.Scanner;
class four {
    public static void main(String[] args) {
        int age, year;
        Scanner c = new Scanner(System.in);
        System.out.println("Enter age");
        age = Scan.nextIn();
        if (age >= 18) {
            System.out.println("You can Vote");
        }
        else {
            year = (18 - age);
            System.out.println("You can Vote after " + year + " years");
            assertTrue(age == year);
        }
    }
}
```

output :

Enter age : 19

You can vote

Ex No: 14 Simple Interest in JUnit

Ques : Write a program to calculate the Simple Interest based on the Percentage rate conditions and Verify assert TRUE Code

Program :

```
import static org.junit.Assert.*;
import java.util.Scanner;

class Interest {

    public static void main(String[] args) {
        Scanner Sc = new Scanner(System.in);
        float p = Sc.nextFloat();
        float R = Sc.nextFloat();
        float T = Sc.nextFloat();
        float SI = (P * T * R) / 100;
        System.out.println("Simple Interest = " + SI);
        assertEquals(3600, SI);
    }
}
```

output :

600

600

1

Simple Interest = 3600.0

Aim : To check whether the given number is Palindrome or not and verify the result using assertTrue code

Program:

```
import java.util.Scanner;
import static org.junit.Assert.assertTrue;

Public class Palindrome {
    Public static void main (String args[]) {
        Scanner in = new Scanner (System.in);
        int r, Sum=0, temp; int n = in.nextInt();
        temp=n;
        while (n>0) {
            r=n%10; n=n/10;
            Sum = (sum*10)+r; }
        System.out.println(Sum);
        assertTrue(787==Sum);
        if (temp==sum)
            System.out.println(Sum+" is Palindrome number");
        else
            System.out.println(Sum+" is not Palindrome number");
        } }
```

output :

787

787

787 is Palindrome number.

Exp 16: Decimal to Binary conversion and Octal in Junit

Aim:

To convert the decimal number to its equivalent binary number and octal number and the output values verified using Assert code

Program:

```
import java.util.Scanner;

class binary {

    public static void main (String[] args) {
        Scanner in = new Scanner (System.in);
        int decimal = in.nextInt();
        String binary = Integer.toBinaryString
                               (decimal);
        System.out.println ("Binary is" + binary);
        System.out.print ("Octal is");
        System.out.println (Integer.toOctalString
                               (decimal));
        assert True (14 == decimal);    } }
```

output:

14

Binary is 1110

Octal is 16