

Exp No : 11

Demonstrate the working of JUnit
to Reverse a word and using assert
statement for Proof of the value

Aim : To understand the working of JUnit
assert statements by Comparing the Reversed value
with expected one

Code :

```
import static org.junit.Assert.assertEquals;
import java.util.Scanner;
class Sreeetha {
    public static void main (String [] args) {
        String str = "Hello world";
        Scanner sc = new Scanner (System.in);
        System.out.print ("Enter a string");
        str = sc.nextLine ();
        System.out.println ("Reverse of a string" + str);
        for (int j = str.length () - 1; j > 0; --j)
            System.out.print (str.charAt (j));
        assertEquals ("Hello", str);
        assertEquals ("olleH", str);
    }
}
```

Input Actual Output

Mani

inam

Test Case No: 1

Test Case name: Expected One Save or Actual One

Input mani

Expected O/p Actual O/p Remarks

From From From : SUCCESS

Exp No: 12

Write a White Box testing code (JUnit) to string comparison of word and using assert statement for proof the value

Aim: To understand the working of JUnit assert Statement by Comparing two strings

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

Public class Third {

(public class void main(String[] args)

{Scanner in = new Scanner (System.in);}

System.out.println ("Enter the user name")

Str1 = in.nextLine();

System.out.println ("Reenter the user name")

Str2 = in.nextLine();

assert assertEquals (str1, str2);

(String "Name") Input lines

3

O/P

Enter user name = Amne

Reenter user name = Amne

Exp No: 13

Write a Junit code for voting System and uses assert statement and verify the white box testing.

Aim: To understand the working of Unit Test Statement By Checking the voting age

Import static org.junit.Assert.assertEquals

```
import java.util.Scanner;
```

q

Public static void main (String[] args)

ben (Jedig) nimm bzw. schreibe schrift

int age, shot;

```
Scanner Scan = new Scanner (System.in);
```

```
System.out.println("Please enter your age");
```

age = Scan-Read Int();

if $\text{Age} \geq 18$

χ

Welcome to voting

3

~~(cool) (fast) (hot)~~ - 12 foot Can vote's

۲

else

$$S_{hot} = (18 - age)$$

~~System.out.println("Sorry, you can vote after " +
"5 years");~~

assert True (age == shot);

ج ج ج

~~Op~~ code: 70 Return

003 : Eng

input = 19

Expected off - you can vote Actual off

Remarks: Successful

Exp No: 14

Simple Interest

Aim: Write a Program that calculates the Simple interest based on the Percentage Rate Conditions and verify the result using assert True Code.

Import static org.junit.Assert.assertEquals;

import java.util.Scanner;

Class Interest

{

 // (Condition) Input Data is Integers

 Public static void main (String[] args)

{

 Scanner sc = new Scanner (System.in);

 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);

 assert.assertEquals (3600 == SI);

 }

3

of

Input: 600
 600

Expected of: 3600.0 Actual of:

P1 = 3600.0 : 36000.0

Remarks: Successful

2. Journal - ans

Ex No : 15

J1 : 06/9/17

Aim :- To check whether the given number is Palindrome or not and verify the result using assert True code.

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

public class palindrome {
    public static void main (String args[])
    {
        int n, sum=0, temp; int a = Integer.parseInt(args[0]);
        while (n>0)
        {
            temp = n % 10;
            sum = sum + temp;
            n = n / 10;
        }
        if (sum == temp)
            System.out.println("Sum is a Palindrome number");
        else
            System.out.println("Sum is not a Palindrome number");
    }
}
```

O/P

Input: 887

Expected O/P: 887 is a palindrome

Remarks: Successful