

27. JUNIT TESTING TO CHECK WHETHER THE GIVEN STRING IS GETTING REVERSED OR NOT

J ASHOK REDDY

192011022

AIM

To Perform Junit Testing to Check Whether the given string is getting Reversed or not

PROGRAM

```
package sse;

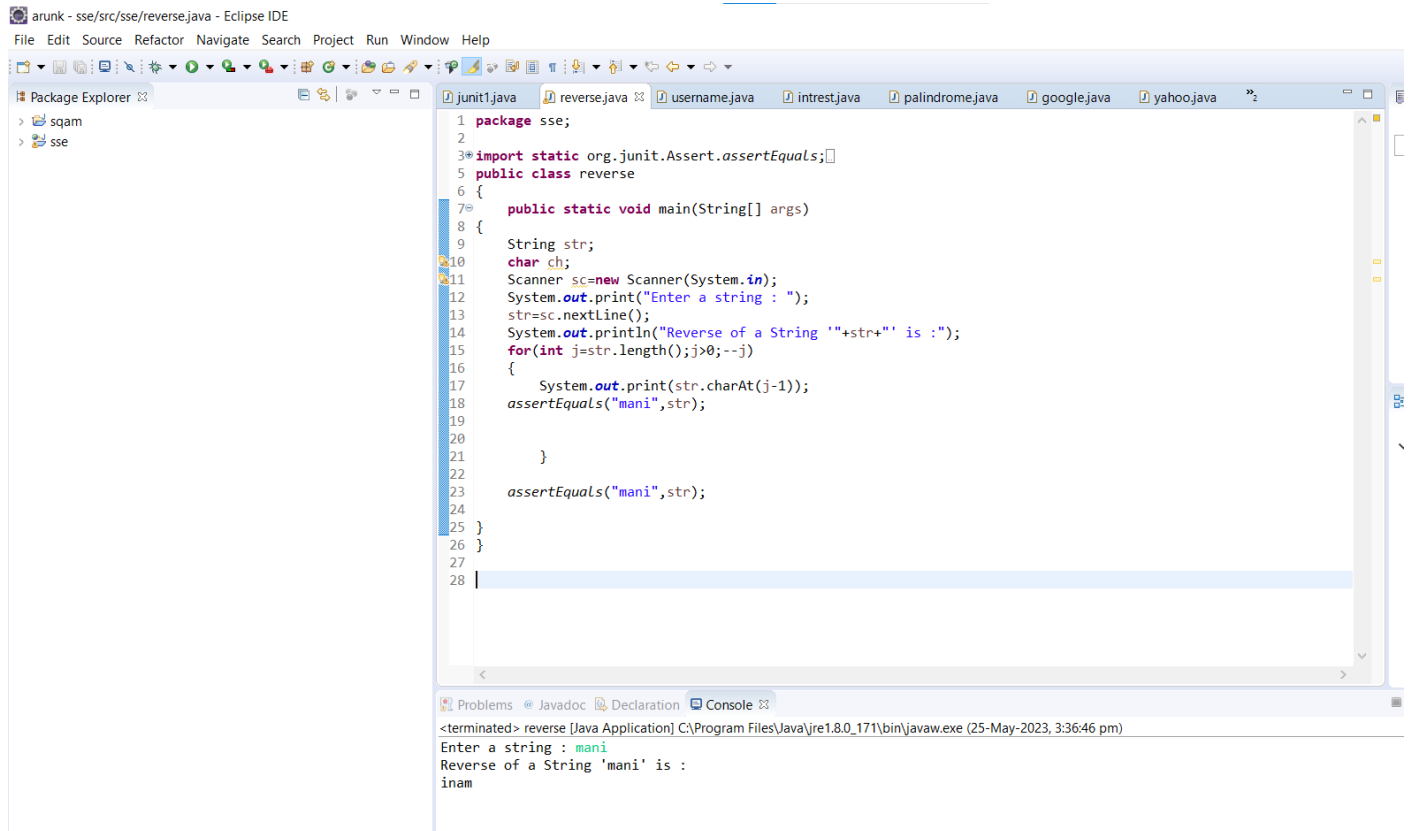
import static org.junit.Assert.assertEquals;
import java.util.Scanner;

public class reverse
{
    public static void main(String[] args)
    {
        String str;
        char ch;
        Scanner sc=new Scanner(System.in);
        System.out.print("Enter a string : ");
        str=sc.nextLine();
        System.out.println("Reverse of a String '"+str+"' is :");
        for(int j=str.length();j>0;--j)
        {
            System.out.print(str.charAt(j-1));
            assertEquals("mani",str);

        }

        assertEquals("mani",str);
    }
}
```

OUTPUT

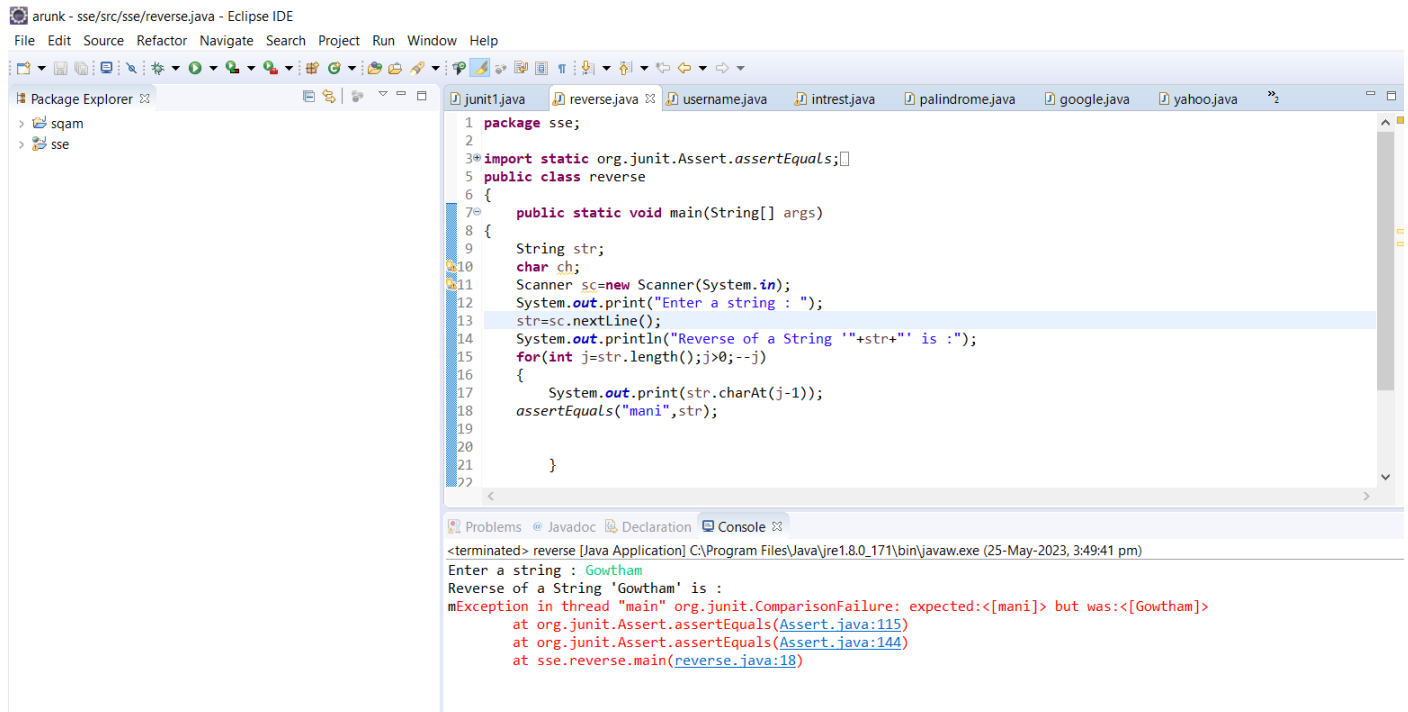


The screenshot shows the Eclipse IDE with the file `reverse.java` open. The code defines a package `sse` and a class `reverse` with a `main` method. The `main` method uses `Scanner` to read a string, prints its reverse, and then uses `assertEquals` to verify that the reverse of the string "mani" is "inam". The console output shows the program running successfully, with the input "mani" and the output "inam".

```
1 package sse;
2
3 import static org.junit.Assert.assertEquals;
4
5 public class reverse
6 {
7     public static void main(String[] args)
8     {
9         String str;
10        char ch;
11        Scanner sc=new Scanner(System.in);
12        System.out.print("Enter a string : ");
13        str=sc.nextLine();
14        System.out.println("Reverse of a String '"+str+"' is :");
15        for(int j=str.length();j>0;--j)
16        {
17            System.out.print(str.charAt(j-1));
18            assertEquals("mani",str);
19        }
20
21        assertEquals("mani",str);
22    }
23 }
24
25 }
26 }
27
28 |
```

Console Output:

```
<terminated> reverse [Java Application] C:\Program Files\Java\jre1.8.0_171\bin\javaw.exe (25-May-2023, 3:36:46 pm)
Enter a string : mani
Reverse of a String 'mani' is :
inam
```



The screenshot shows the Eclipse IDE with the file `reverse.java` open. The code is identical to the previous one, but the `assertEquals` call in the `main` method is now `assertEquals("Gowtham",str);`. The console output shows the program running, but it fails with a `ComparisonFailure` exception because the expected string "mani" does not match the actual string "Gowtham".

```
1 package sse;
2
3 import static org.junit.Assert.assertEquals;
4
5 public class reverse
6 {
7     public static void main(String[] args)
8     {
9         String str;
10        char ch;
11        Scanner sc=new Scanner(System.in);
12        System.out.print("Enter a string : ");
13        str=sc.nextLine();
14        System.out.println("Reverse of a String '"+str+"' is :");
15        for(int j=str.length();j>0;--j)
16        {
17            System.out.print(str.charAt(j-1));
18            assertEquals("Gowtham",str);
19        }
20
21        assertEquals("Gowtham",str);
22    }
23 }
24
25 }
26 }
27
28 |
```

Console Output:

```
<terminated> reverse [Java Application] C:\Program Files\Java\jre1.8.0_171\bin\javaw.exe (25-May-2023, 3:49:41 pm)
Enter a string : Gowtham
Reverse of a String 'Gowtham' is :
mException in thread "main" org.junit.ComparisonFailure: expected:<[mani]> but was:<[Gowtham]>
    at org.junit.Assert.assertEquals(Assert.java:115)
    at org.junit.Assert.assertEquals(Assert.java:144)
    at sse.reverse.main(reverse.java:18)
```

RESULT

Hence the Junit Testing to Check Whether the given string is getting Reversed or not performed successfully.