

### 30. Junit Testing to check whether the given number is palindrome or not

#### AIM

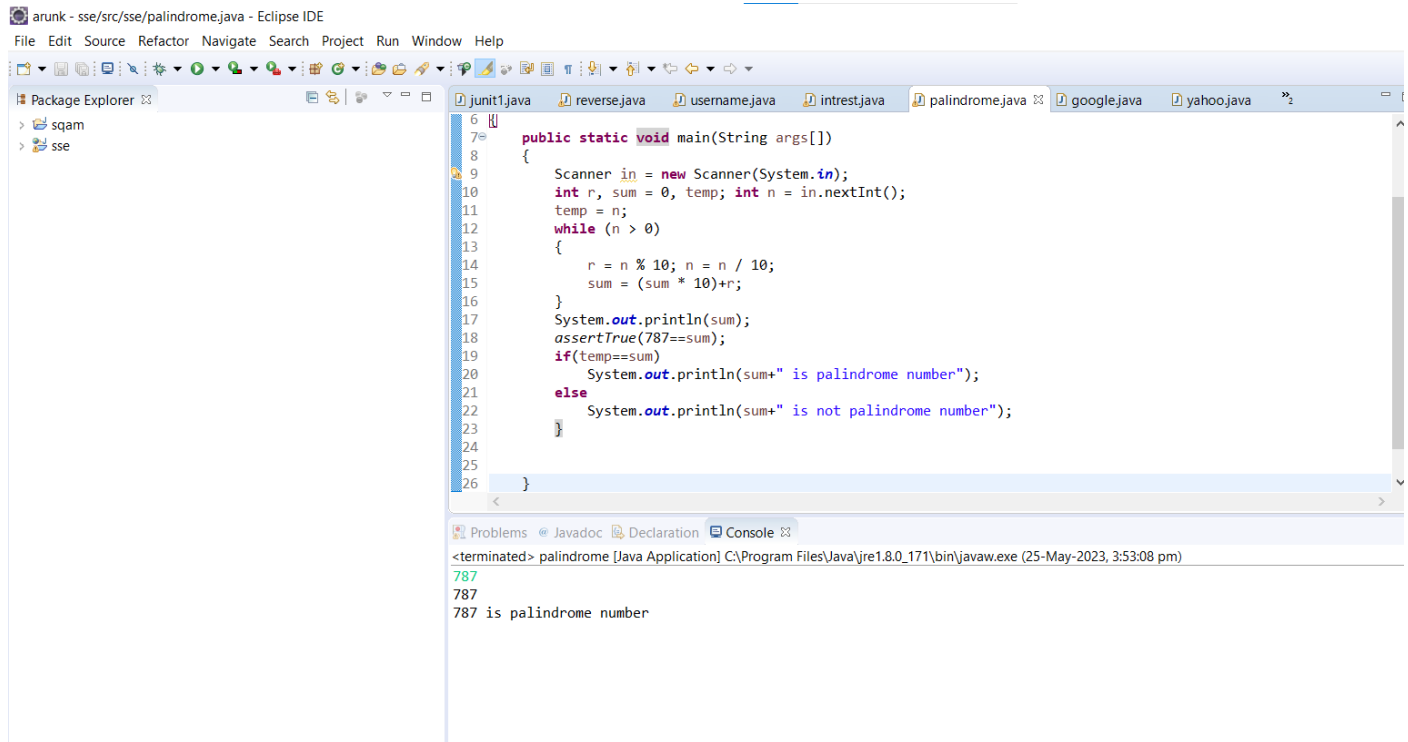
To Perform junit Testing to Check Whether the given number is palindrome or not.

#### PROGRAM

```
package sse;

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");
    }
}
```

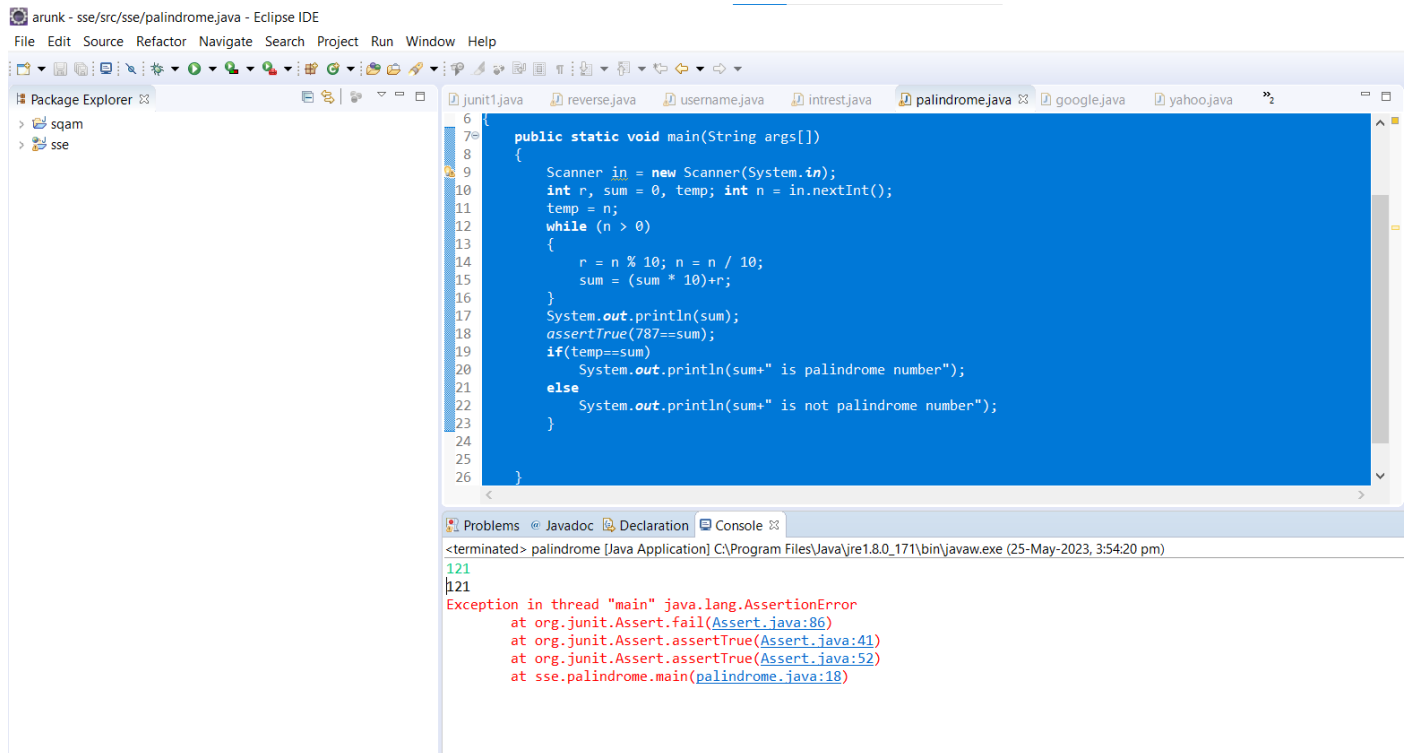
## OUTPUT



The screenshot shows the Eclipse IDE interface. The Package Explorer on the left displays a project named 'sse' with a sub-package 'sse'. The main editor window shows the source code for 'palindrome.java'. The code defines a 'main' method that takes an array of strings 'args'. It uses a 'Scanner' to read an integer 'n' from 'System.in'. A 'while' loop extracts the digits of 'n' from right to left, building a reversed number 'sum'. After the loop, it prints 'sum' and uses 'assertTrue(787==sum)' to verify the result. Since the assertion passes, it prints '787 is palindrome number'. The Console window at the bottom shows the output: '787' followed by '787 is palindrome number'.

```
6 {
7     public static void main(String args[])
8     {
9         Scanner in = new Scanner(System.in);
10        int r, sum = 0, temp; int n = in.nextInt();
11        temp = n;
12        while (n > 0)
13        {
14            r = n % 10; n = n / 10;
15            sum = (sum * 10)+r;
16        }
17        System.out.println(sum);
18        assertTrue(787==sum);
19        if(temp==sum)
20            System.out.println(sum+" is palindrome number");
21        else
22            System.out.println(sum+" is not palindrome number");
23    }
24 }
25
26 }
```

<terminated> palindrome [Java Application] C:\Program Files\Java\jre1.8.0\_171\bin\javaw.exe (25-May-2023, 3:53:08 pm)  
787  
787  
787 is palindrome number



The screenshot shows the Eclipse IDE interface. The Package Explorer on the left displays a project named 'sse' with a sub-package 'sse'. The main editor window shows the source code for 'palindrome.java'. The code is identical to the previous one, but the Console window at the bottom shows a different output. It displays an 'AssertionError' exception in the 'main' thread, indicating that the assertion '787==sum' failed. The stack trace points to the 'assertTrue' method in 'Assert.java' and the 'main' method in 'palindrome.java'.

```
6 {
7     public static void main(String args[])
8     {
9         Scanner in = new Scanner(System.in);
10        int r, sum = 0, temp; int n = in.nextInt();
11        temp = n;
12        while (n > 0)
13        {
14            r = n % 10; n = n / 10;
15            sum = (sum * 10)+r;
16        }
17        System.out.println(sum);
18        assertTrue(787==sum);
19        if(temp==sum)
20            System.out.println(sum+" is palindrome number");
21        else
22            System.out.println(sum+" is not palindrome number");
23    }
24 }
25
26 }
```

<terminated> palindrome [Java Application] C:\Program Files\Java\jre1.8.0\_171\bin\javaw.exe (25-May-2023, 3:54:20 pm)  
121  
121  
Exception in thread "main" java.lang.AssertionError  
at org.junit.Assert.fail(Assert.java:86)  
at org.junit.Assert.assertTrue(Assert.java:41)  
at org.junit.Assert.assertTrue(Assert.java:52)  
at sse.palindrome.main(palindrome.java:18)

## RESULT

Hence the junit Testing Check Whether the given number is palindrome or not performed successfully.