

30. JUNIT TESTING TO CHECK WHETHER THE GIVEN NUMBER IS PALINDROME OR NOT

J ASHOK REDDY

192011022

AIM

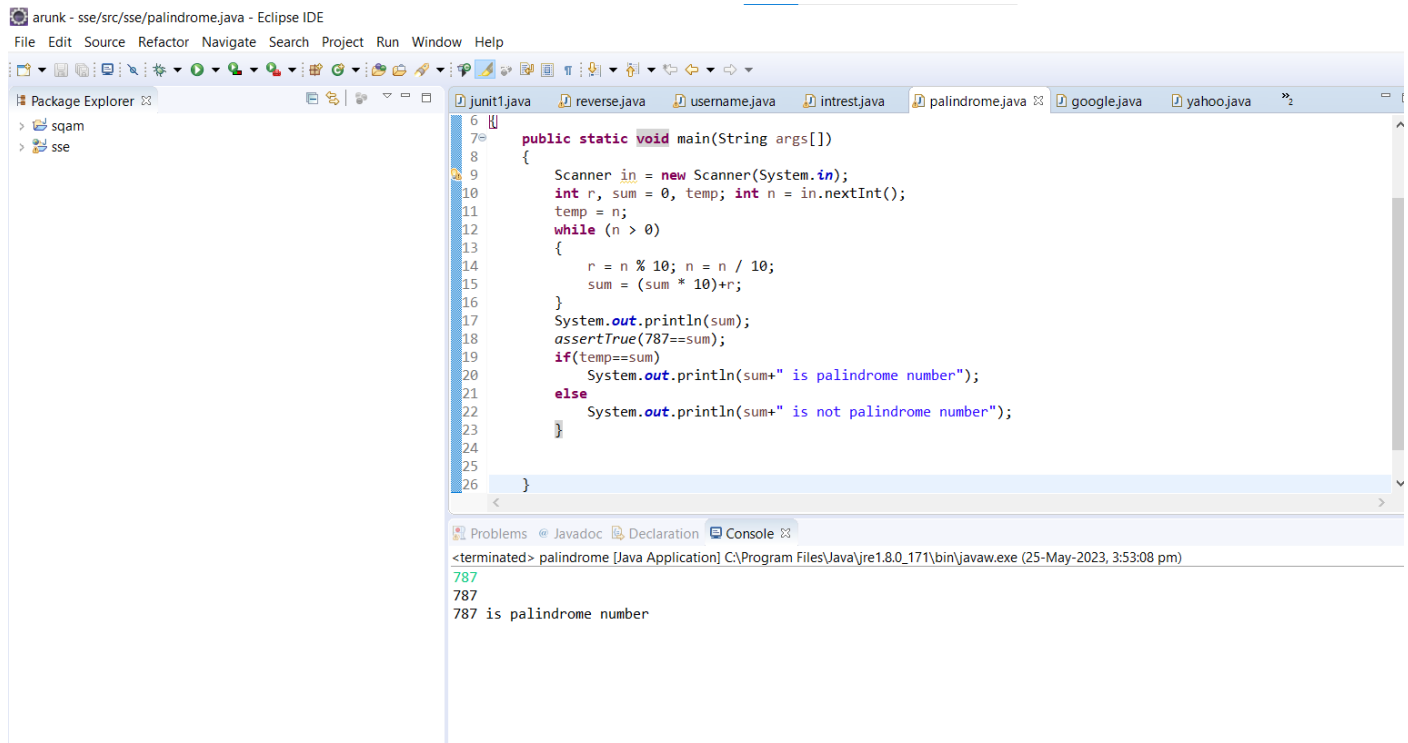
To Perform junit Testing to Check Whether the given number is palindrome or not.

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

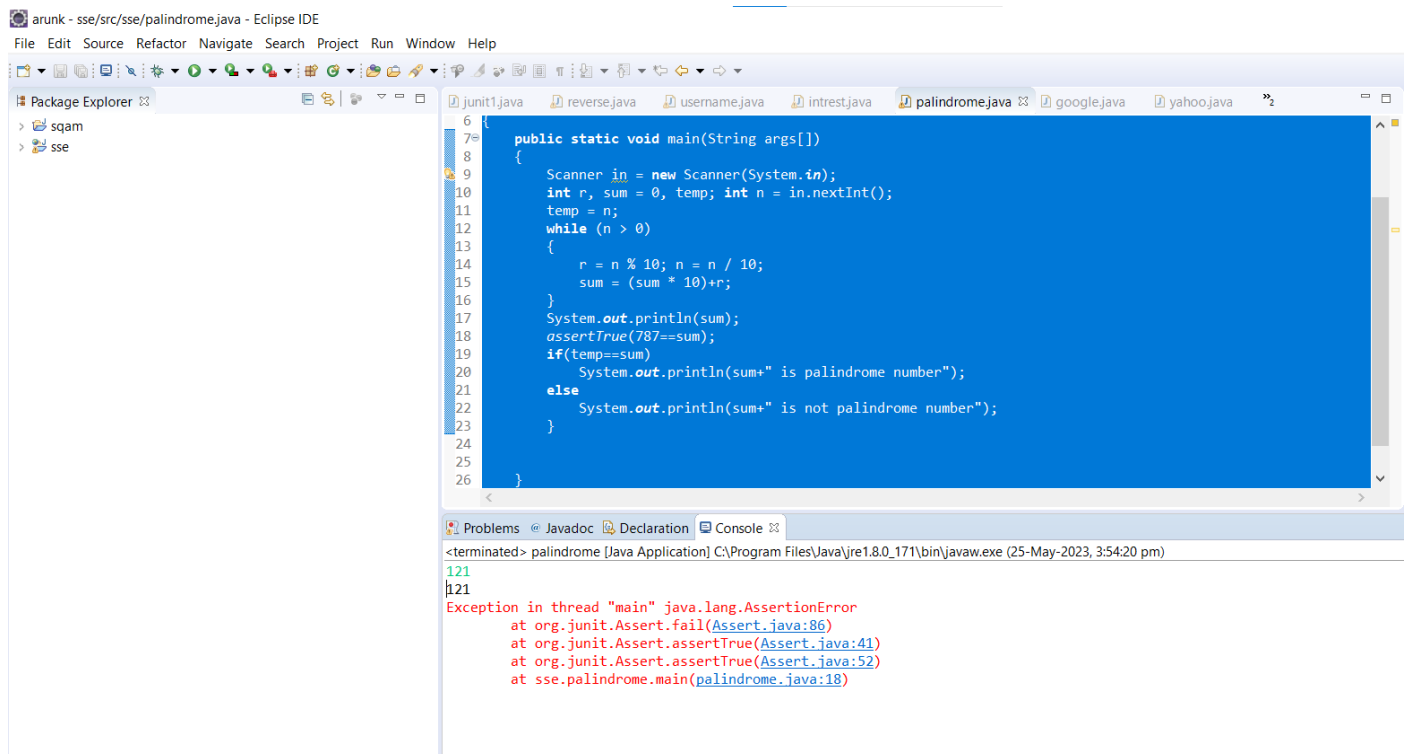


The screenshot shows the Eclipse IDE with a Java project named 'sse'. The 'Package Explorer' on the left shows the project structure. The main editor displays the source code for 'palindrome.java'. The code defines a 'main' method that uses a 'Scanner' to read an integer 'n'. It then enters a 'while' loop that calculates the reverse of the number by repeatedly taking the last digit and adding it to a 'sum' while dividing the number by 10. After the loop, it prints the 'sum' and checks if it equals the original number 'n' using 'assertTrue'. If the assertion passes, it prints 'is palindrome number'; otherwise, it prints 'is not palindrome number'.

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

The 'Console' view at the bottom shows the output of the program:

```
<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 with the same 'sse' project. The 'palindrome.java' file is open in the editor. The code is identical to the previous screenshot. However, the 'Console' view shows a different outcome: an 'AssertionError'.

```
<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.