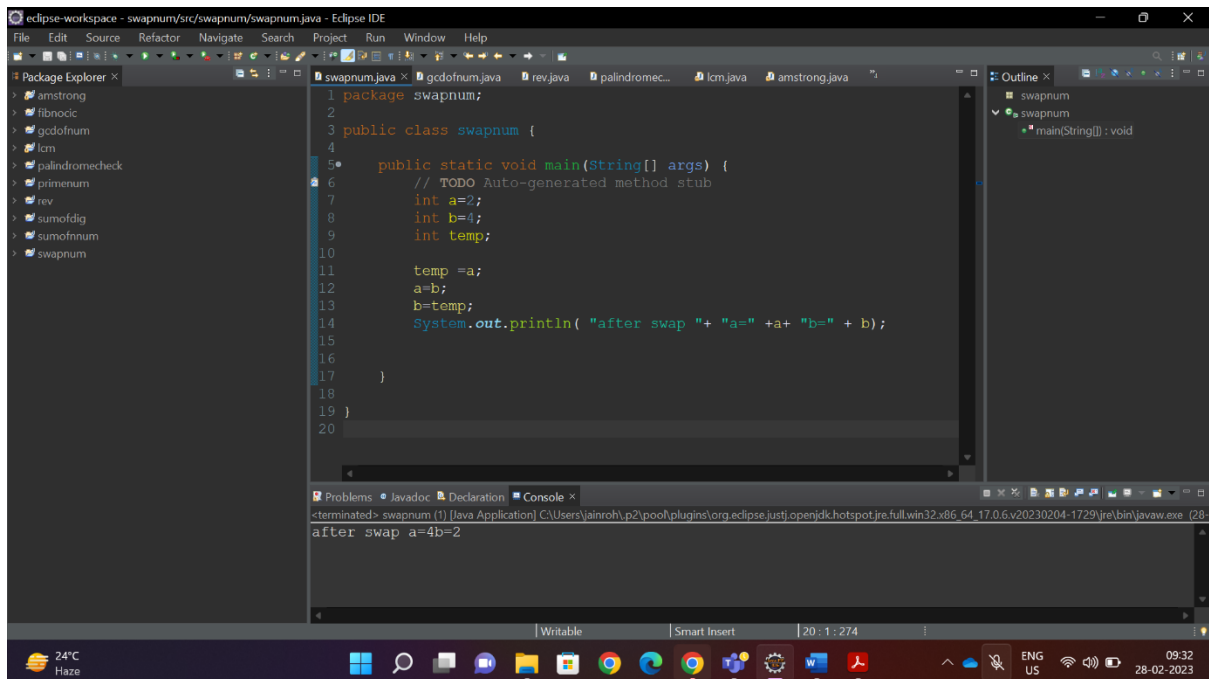


Assignment 1:

Rohan Jain T B

1) Write a program to swap two numbers in Java.

<https://codeshare.io/eV6YrK>



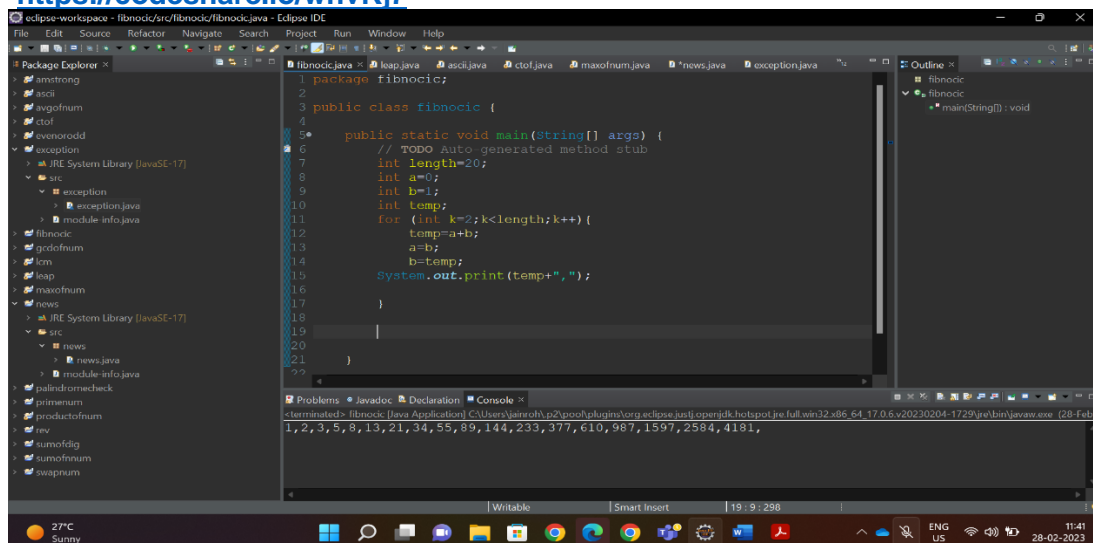
The screenshot shows the Eclipse IDE with a project named 'swapnum'. The 'Package Explorer' on the left shows the project structure. The 'Main' editor displays the following Java code:

```
1 package swapnum;
2
3 public class swapnum {
4
5     public static void main(String[] args) {
6         // TODO Auto-generated method stub
7         int a=2;
8         int b=4;
9         int temp;
10
11         temp =a;
12         a=b;
13         b=temp;
14         System.out.println( "after swap " + "a=" +a+ "b=" + b);
15
16     }
17 }
18
19 }
20
```

The 'Console' at the bottom shows the output: 'after swap a=4b=2'.

2) Write a program to print all the elements of the Fibonacci series.

<https://codeshare.io/wnvKj7>



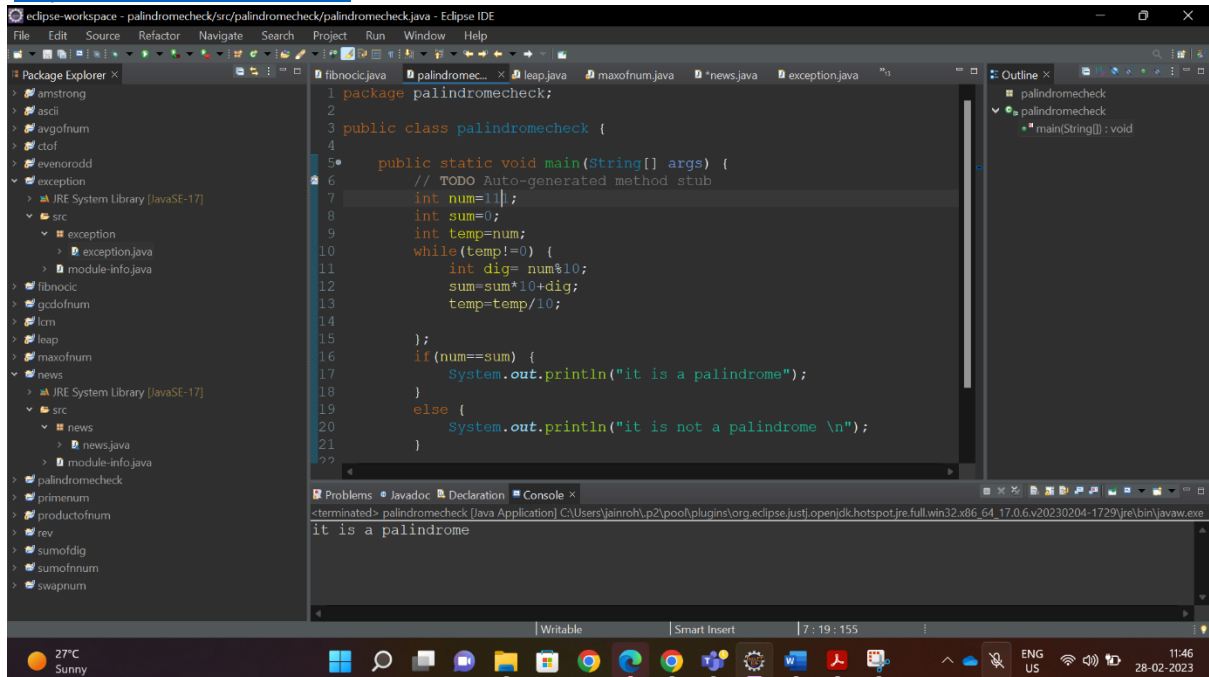
The screenshot shows the Eclipse IDE with a project named 'fibnodic'. The 'Package Explorer' on the left shows the project structure. The 'Main' editor displays the following Java code:

```
1 package fibnodic;
2
3 public class fibnodic {
4
5     public static void main(String[] args) {
6         // TODO Auto-generated method stub
7         int length=20;
8         int a=0;
9         int b=1;
10        int temp;
11        for (int k=2;k<length;k++) {
12            temp=a+b;
13            a=b;
14            b=temp;
15            System.out.print(temp+", ");
16        }
17    }
18
19 }
20
21 }
```

The 'Console' at the bottom shows the output: '1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597, 2584, 4181,'.

3) Check if a given number is palindrome or not.

<https://codeshare.io/ZJErdV>

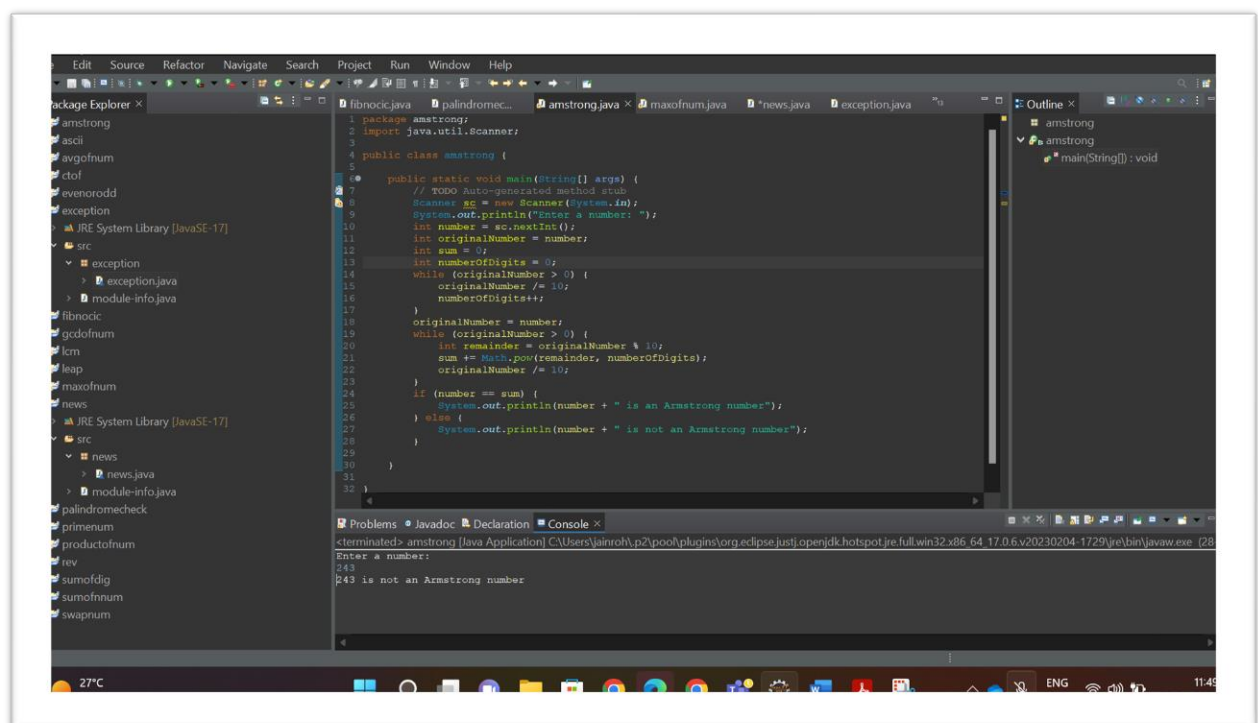


```
1 package palindromecheck;
2
3 public class palindromecheck {
4
5     public static void main(String[] args) {
6         // TODO Auto-generated method stub
7         int num=111;
8         int sum=0;
9         int temp=num;
10        while(temp!=0) {
11            int dig= num%10;
12            sum=sum*10+dig;
13            temp=temp/10;
14        };
15        if(num==sum) {
16            System.out.println("it is a palindrome");
17        }
18        else {
19            System.out.println("it is not a palindrome \n");
20        }
21    }
22 }
```

Console output: it is a palindrome

4) Write a program to find whether a number is an Armstrong number or not.

<https://codeshare.io/1Y8x7B>

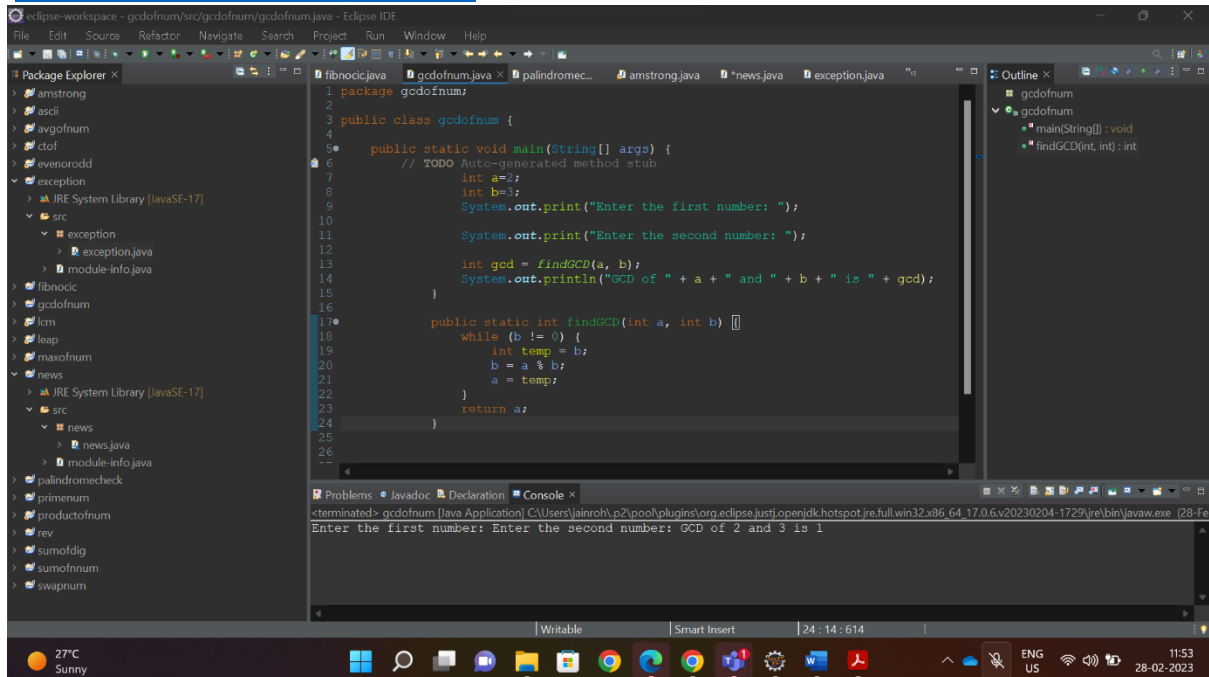


```
1 package armstrong;
2 import java.util.Scanner;
3
4 public class armstrong {
5
6     public static void main(String[] args) {
7         // TODO Auto-generated method stub
8         Scanner sc = new Scanner(System.in);
9         System.out.println("Enter a number: ");
10        int number = sc.nextInt();
11        int originalNumber = number;
12        int sum = 0;
13        int numberOfDigits = 0;
14        while (originalNumber > 0) {
15            originalNumber /= 10;
16            numberOfDigits++;
17        }
18        originalNumber = number;
19        while (originalNumber > 0) {
20            int remainder = originalNumber % 10;
21            sum += Math.pow(remainder, numberOfDigits);
22            originalNumber /= 10;
23        }
24        if (number == sum) {
25            System.out.println(number + " is an Armstrong number");
26        }
27        else {
28            System.out.println(number + " is not an Armstrong number");
29        }
30    }
31 }
32 }
```

Console output: Enter a number: 243
243 is not an Armstrong number

5) Find the GCD of two numbers.

<https://codeshare.io/pqkOWX>



The screenshot shows the Eclipse IDE with a Java project named 'gcdofnum'. The main class is 'gcdofnum.java'. The code defines a public class 'gcdofnum' with a 'main' method that prompts the user to enter two numbers and calculates their GCD using a recursive 'findGCD' method. The console output shows the user entering '2' and '3', and the program outputting 'GCD of 2 and 3 is 1'.

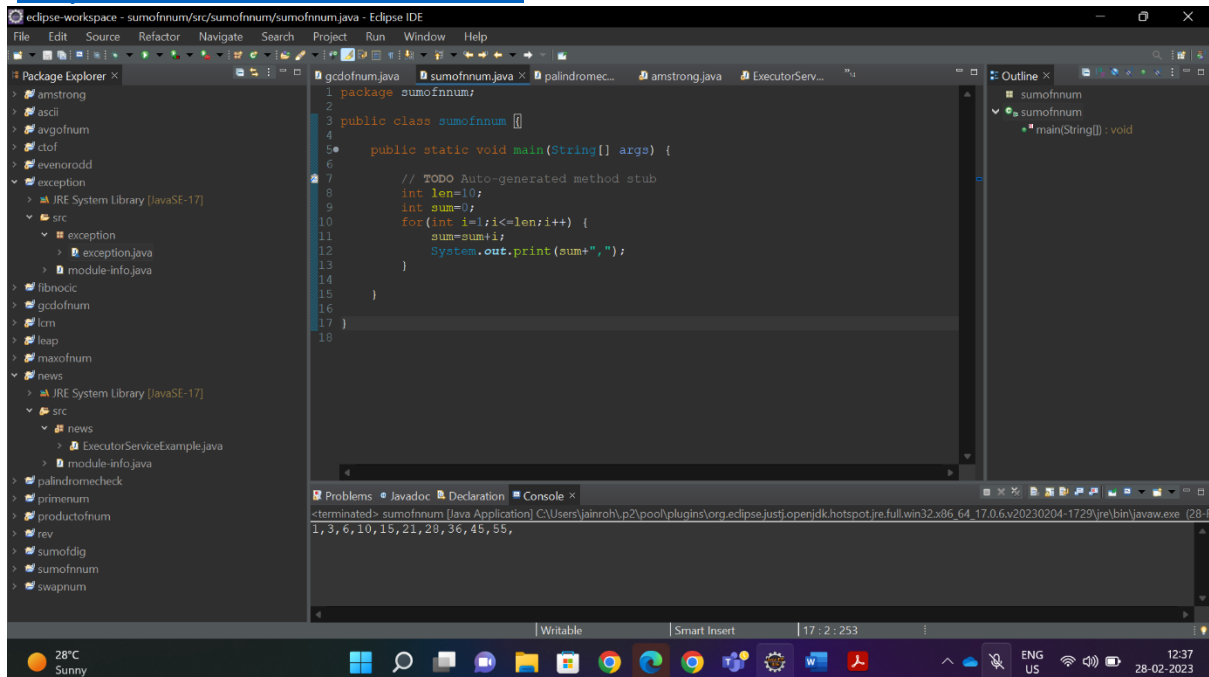
```
1 package gcdofnum;
2
3 public class gcdofnum {
4
5     public static void main(String[] args) {
6         // TODO Auto-generated method stub
7         int a=2;
8         int b=3;
9         System.out.print("Enter the first number: ");
10
11         System.out.print("Enter the second number: ");
12
13         int gcd = findGCD(a, b);
14         System.out.println("GCD of " + a + " and " + b + " is " + gcd);
15     }
16
17     public static int findGCD(int a, int b) {
18         while (b != 0) {
19             int temp = b;
20             b = a % b;
21             a = temp;
22         }
23         return a;
24     }
25 }
26
```

Console Output:

```
<terminated> gcdofnum [Java Application] C:\Users\jainroh\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64.17.0.6.v20230204-1729\jre\bin\javaw.exe (28-fe
Enter the first number: Enter the second number: GCD of 2 and 3 is 1
```

6) Write a program to find the sum of n natural numbers.

<https://codeshare.io/1Y8xQ3>



The screenshot shows the Eclipse IDE with a Java project named 'sumofnum'. The main class is 'sumofnum.java'. The code defines a public class 'sumofnum' with a 'main' method that prompts the user to enter a number 'n' and calculates the sum of the first 'n' natural numbers using a 'for' loop. The console output shows the user entering '5', and the program outputting '1, 3, 6, 10, 15, 21, 28, 36, 45, 55'.

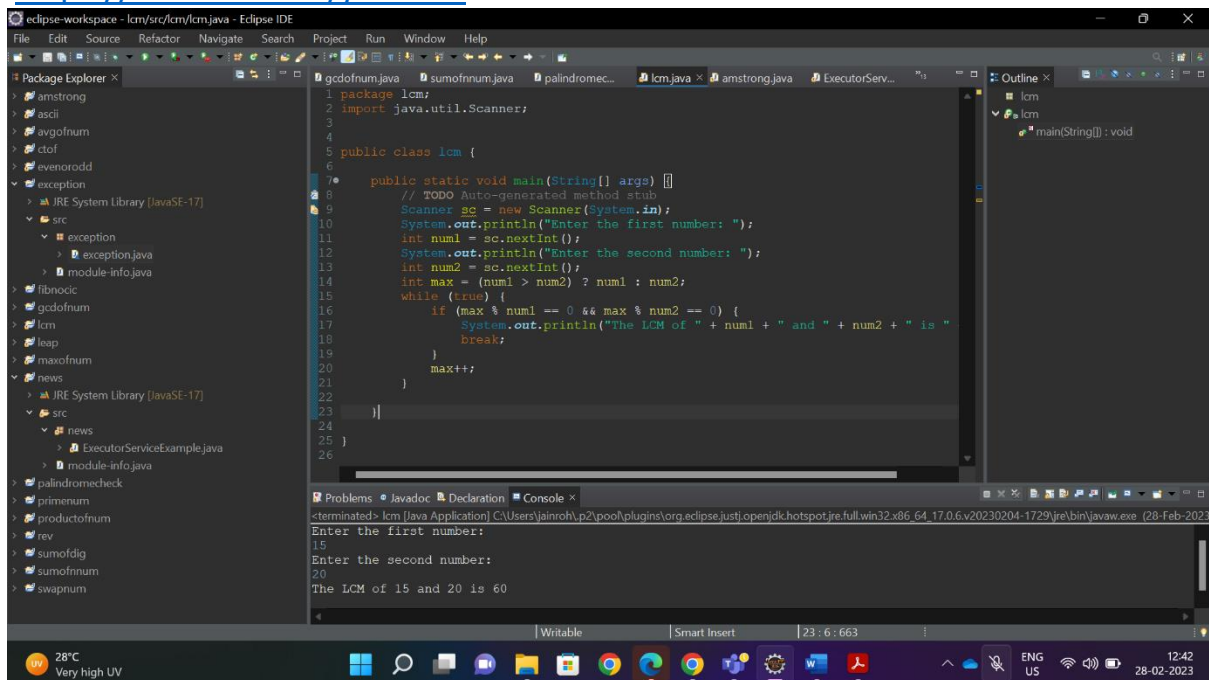
```
1 package sumofnum;
2
3 public class sumofnum {
4
5     public static void main(String[] args) {
6         // TODO Auto-generated method stub
7         int len=10;
8         int sum=0;
9         for(int i=1;i<=len;i++) {
10             sum=sum+i;
11             System.out.print(sum+",");
12         }
13     }
14 }
15
16
17 }
18
```

Console Output:

```
<terminated> sumofnum [Java Application] C:\Users\jainroh\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64.17.0.6.v20230204-1729\jre\bin\javaw.exe (28-f
1, 3, 6, 10, 15, 21, 28, 36, 45, 55,
```

7) Write a program to find the lcm of two numbers.

<https://codeshare.io/j0dDWP>



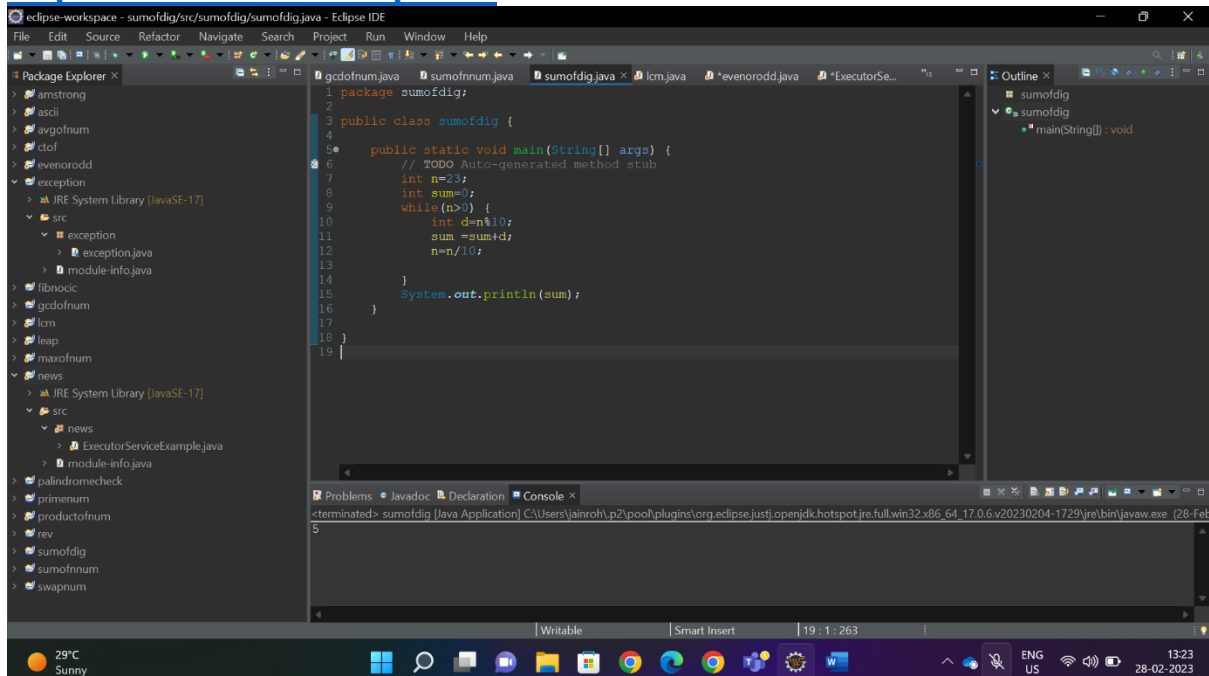
The screenshot shows the Eclipse IDE with a Java project named 'lcm'. The main class is 'lcm.java', which contains the following code:

```
1 package lcm;
2 import java.util.Scanner;
3
4 public class lcm {
5
6     public static void main(String[] args) {
7         // TODO Auto-generated method stub
8         Scanner sc = new Scanner(System.in);
9         System.out.println("Enter the first number: ");
10        int num1 = sc.nextInt();
11        System.out.println("Enter the second number: ");
12        int num2 = sc.nextInt();
13        int max = (num1 > num2) ? num1 : num2;
14        while (true) {
15            if (max % num1 == 0 && max % num2 == 0) {
16                System.out.println("The LCM of " + num1 + " and " + num2 + " is " + max);
17                break;
18            }
19            max++;
20        }
21    }
22 }
```

The console output shows the program running successfully, prompting for the first and second numbers (15 and 20) and displaying the LCM as 60.

8) Calculate the sum of digits of a given number.

<https://codeshare.io/8plenA>



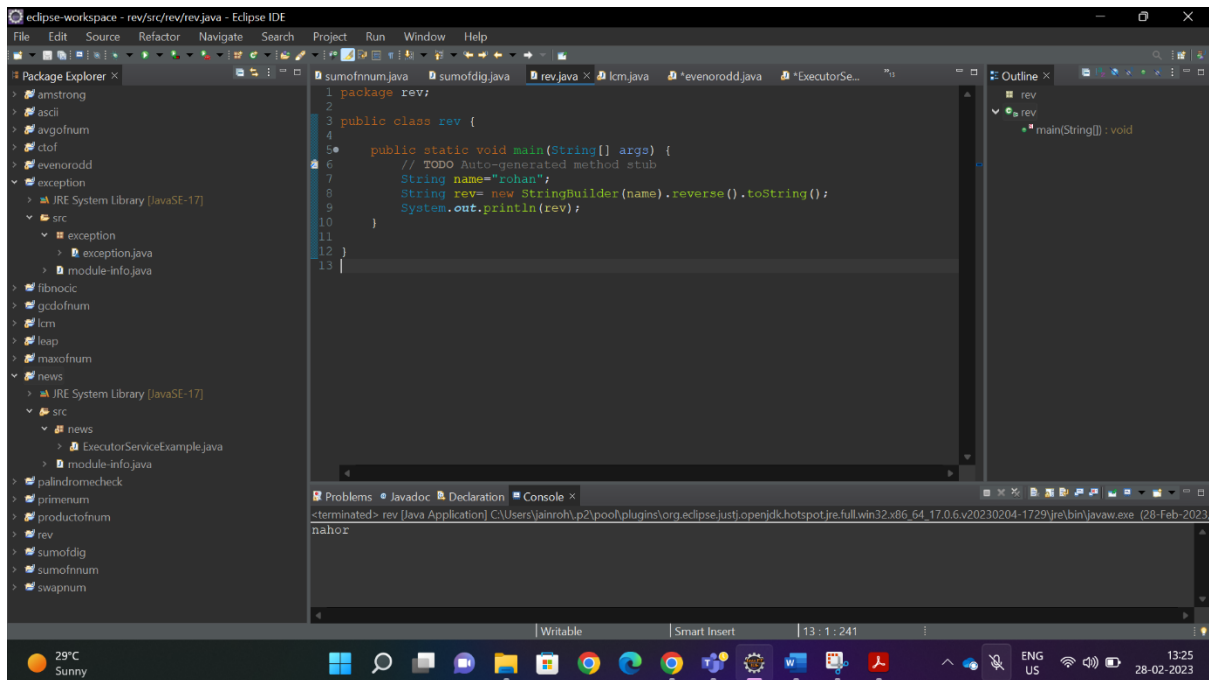
The screenshot shows the Eclipse IDE with a Java project named 'sumofdig'. The main class is 'sumofdig.java', which contains the following code:

```
1 package sumofdig;
2
3 public class sumofdig {
4
5     public static void main(String[] args) {
6         // TODO Auto-generated method stub
7         int n=23;
8         int sum=0;
9         while(n>0) {
10            int d=n%10;
11            sum=sum+d;
12            n=n/10;
13        }
14        System.out.println(sum);
15    }
16 }
```

The console output shows the program running successfully, displaying the sum of digits (5) for the input number 23.

9) Write a program to reverse a string.

<https://codeshare.io/xv4dpB>



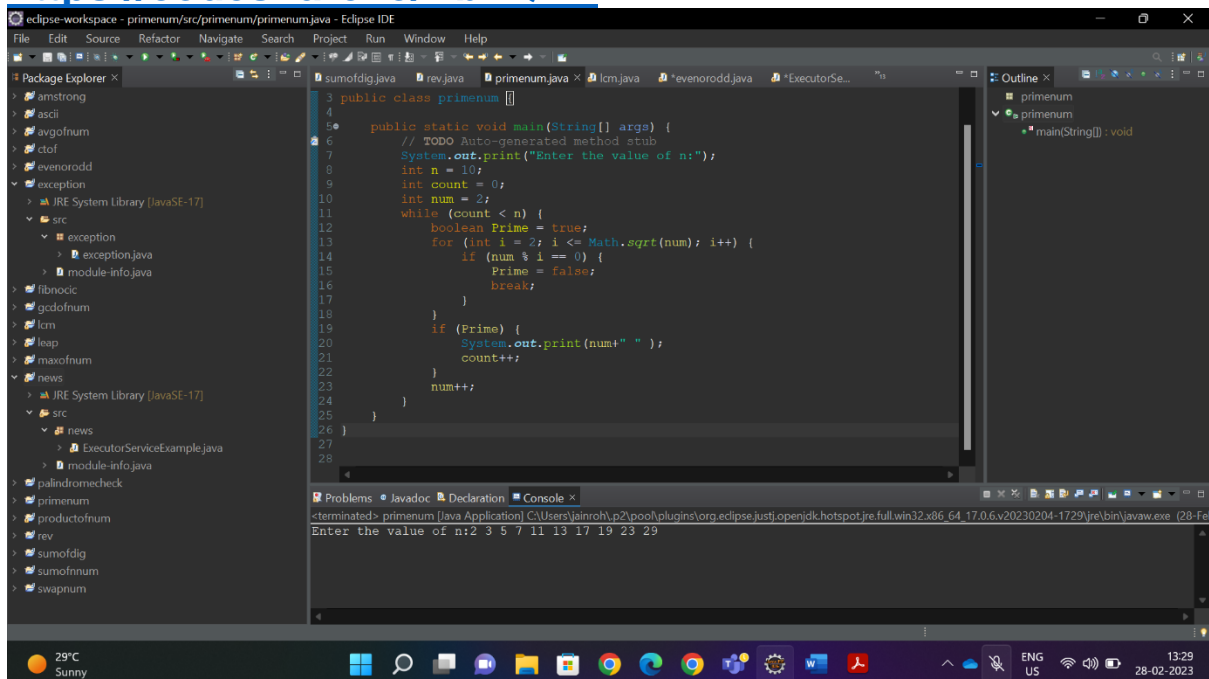
The screenshot shows the Eclipse IDE with a project named 'rev'. The Package Explorer on the left lists various packages, including 'rev'. The main editor displays the following Java code:

```
1 package rev;
2
3 public class rev {
4
5     public static void main(String[] args) {
6         // TODO Auto-generated method stub
7         String name="rohan";
8         String rev= new StringBuilder(name).reverse().toString();
9         System.out.println(rev);
10    }
11 }
12
13
```

The Console window at the bottom shows the output: 'nahor'.

10) Write a code to print all the first n prime numbers where n will be given as input.

<https://codeshare.io/RbvQAV>



The screenshot shows the Eclipse IDE with a project named 'primenum'. The Package Explorer on the left lists various packages, including 'primenum'. The main editor displays the following Java code:

```
1 public class primenum {
2
3     public static void main(String[] args) {
4         // TODO Auto-generated method stub
5         System.out.print("Enter the value of n:");
6         int n = 10;
7         int count = 0;
8         int num = 2;
9         while (count < n) {
10             boolean Prime = true;
11             for (int i = 2; i <= Math.sqrt(num); i++) {
12                 if (num % i == 0) {
13                     Prime = false;
14                     break;
15                 }
16             }
17             if (Prime) {
18                 System.out.print(num+" ");
19                 count++;
20             }
21             num++;
22         }
23     }
24 }
25
26
27
28
```

The Console window at the bottom shows the input and output: 'Enter the value of n: 3 5 7 11 13 17 19 23 29'.