Week-1

Introduction and software requirements to run the java programs:

1# What is the software that helps to run java programs.

- The software that helps to run java programs is JDK(java development kit)
  And JRE(java runtime environment).

2# What is JDK and JRE.

- Java Development Kit (JDK)

  A software development kit that includes tools for developing and compiling Java programs. JDK is a superset of JRE, and contains all the tools needed to compile, debug, and run Java programs.

- Java Runtime Environment (JRE)
  A set of software tools that runs Java programs on a system. JRE includes the Java Virtual Machine (JVM) and run-time class libraries.

To run Java applications, you only need JRE. However, if you want to develop Java applications, you need JDK, which includes both the JVM and development tools.

3# What is eclipse IDE.

Eclipse IDE (Integrated Development Environment) is a popular open-source software platform primarily used for programming. It provides a comprehensive environment where developers can write, test, and debug code in various programming languages. Initially designed for Java development, Eclipse has evolved to support a wide range of languages through plugins, including Python, C++, JavaScript, and more.

Key Features of Eclipse IDE:

1. **Code Editing**: It includes advanced features like syntax highlighting, code completion, and error checking.
2. **Debugging**: Offers powerful debugging tools, allowing developers to set breakpoints, inspect variables, and control program execution.
3. **Extensibility**: With a vast plugin ecosystem, Eclipse can be customized to meet different programming needs.
4. **Version Control Integration**: Supports integration with Git, SVN, and other version control systems.

5. **Build Automation**: It can integrate with tools like Maven and Gradle for project building and dependency management.
6. **Cross-Platform**: Available for Windows, macOS, and Linux.

4# How to run the java program in eclipse/NetBeans IDE.

Running a Java Program in Eclipse IDE/NetBeans IDE:

Step 1: Install Eclipse and Set Up Java

- Download and install **Eclipse IDE/**NetBeans IDE **for Java Developers** from the Eclipse website and Apache NetBeans website.
- Ensure that **Java Development Kit (JDK)** is installed and configured on your system. Eclipse/NetBEans typically detects the JDK automatically.

Step 2: Create a New Java Project

1. Open **Eclipse IDE**/NetBeans IDE.
2. Go to **File > New > Java Project**.
3. Name your project and click **Finish**.

Step 3: Create a Java Class

1. Right-click the **src** folder in your project in the **Project Explorer** pane.
2. Select **New > Class**.
3. Give your class a name (e.g., `HelloWorld`), and check the box for **public static void main(String[] args)**.
4. Click **Finish**.

Step 4: Run the Java Program

1. Click the green **Run** button in the toolbar (or press **F6** on your keyboard).
2. The output will appear in the **Output** window at the bottom of the IDE.

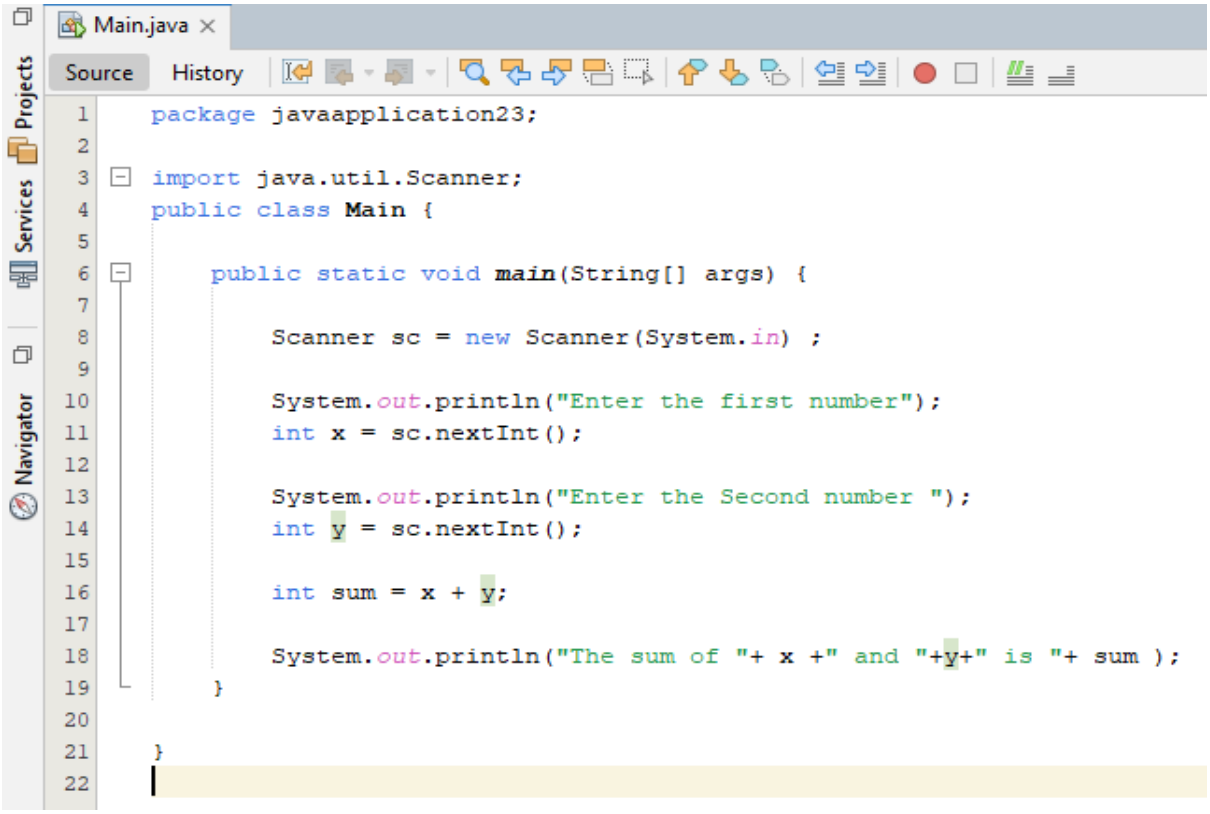
5# What is the software other than eclipse/NetBeans to run the java programs.

- **IntelliJ IDEA**
- **BlueJ**
- **JDeveloper**
- **DrJAVA**
- **JGrasp etc.**

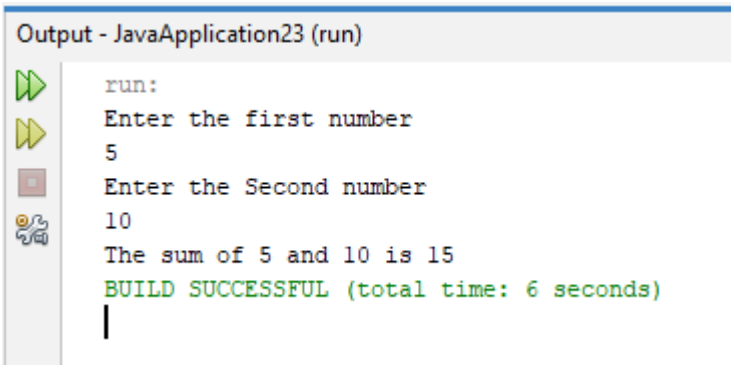
Week-2

Basics problems in java

1# Write a java program to add the two numbers.



Output:



2# Write a java program to multiply two floating numbers.

Main.java | NewClass.java

Source | History

```
1
2   package javaapplication23;
3
4   import java.util.Scanner;
5   public class NewClass {
6
7
8       public static void main(String[] args){
9
10          Scanner sc = new Scanner(System.in) ;
11
12          System.out.println("Enter the first number");
13          double x = sc.nextDouble();
14
15
16
17          System.out.println("Enter the Second number");
18          double y = sc.nextDouble();
19
20
21          double pro = x*y;
22
23          System.out.println("Product " + pro);
24      }
25  }
```

Output:

Output - JavaApplication23 (run)

```
run:
Enter the first number
6.5
Enter the Second number
7.9
Product 51.35
BUILD SUCCESSFUL (total time: 5 seconds)
```

3# Write a java program to display a cube of a number.

Main.java | NewClass.java | cube.java

Source | History

```
1  package javaapplication23;
2
3  import java.util.Scanner;
4  public class cube {
5
6      public static void main(String[] args){
7
8          Scanner sc = new Scanner(System.in) ;
9
10         System.out.println("Enter the  number");
11         int x = sc.nextInt();
12
13         int cube = (x*x*x);
14
15         System.out.println("The cube of " + x+ " is " +cube);
16 }
17 }
```

Output:

Output - JavaApplication23 (run)

```
run:
Enter the  number
6
The cube of 6 is 216
BUILD SUCCESSFUL (total time: 2 seconds)
```

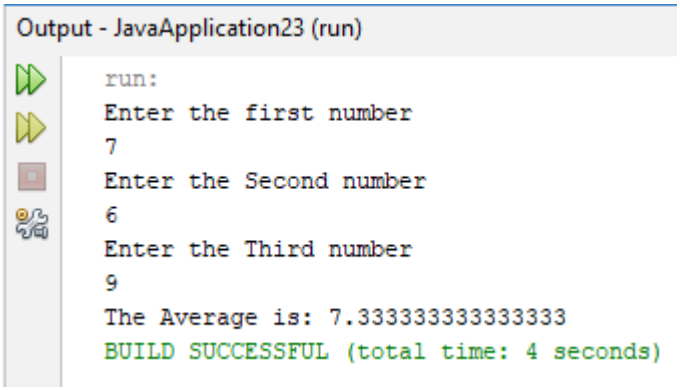
4# Write a Java program that takes three numbers as input to calculate and print the average of the numbers.

Main.java | NewClass.java | cube.java | average.java

Source | History

```
1  package javaapplication23;
2
3  import java.util.Scanner;
4  public class average {
5
6      public static void main(String[] args){
7
8          Scanner sc = new Scanner(System.in) ;
9
10         System.out.println("Enter the first number");
11         double x = sc.nextDouble();
12
13         System.out.println("Enter the Second number");
14         double y = sc.nextDouble();
15
16         System.out.println("Enter the Third number");
17         double z = sc.nextDouble();
18
19         double avg = (x+y+z)/3.0;
20
21         System.out.println("The Average is: " +avg );
22 }
23 }
```

Output:



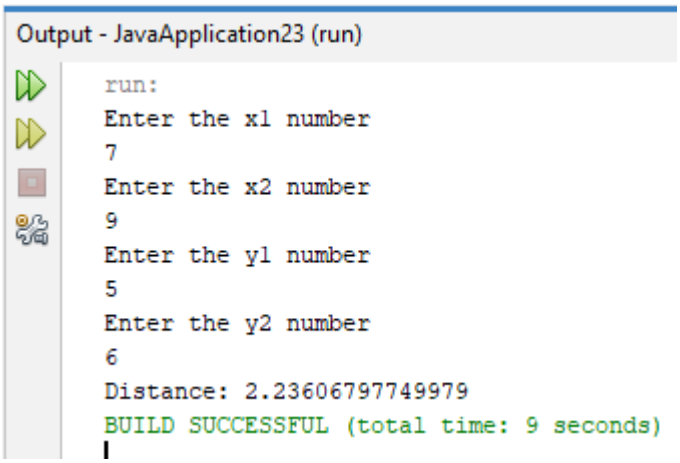
5# Write a Java program to compute the distance between two points.

Main.java | NewClass.java | cube.java | average.java | distance.java

Source | History

```
1  package javaapplication23;
2
3  import java.util.Scanner;
4  public class distance {
5
6
7      public static void main(String[] args){
8
9          Scanner sc = new Scanner(System.in) ;
10
11         System.out.println("Enter the x1 number");
12        int x1 = sc.nextInt();
13
14        System.out.println("Enter the x2 number");
15        int x2 = sc.nextInt();
16
17        System.out.println("Enter the y1 number");
18        int y1 = sc.nextInt();
19
20        System.out.println("Enter the y2 number");
21        int y2 = sc.nextInt();
22
23        double dis = Math.sqrt(Math.pow(x1-x2, 2) + Math.pow(y1 - y2, 2));
24
25        System.out.println("Distance: "+ dis);
26 }
27 }
```

Output:


Output - JavaApplication23 (run)
run:
Enter the x1 number
7
Enter the x2 number
9
Enter the y1 number
5
Enter the y2 number
6
Distance: 2.23606797749979
BUILD SUCCESSFUL (total time: 9 seconds)

Week-3

Problems Based on if statement/Looping in JAVA

1# Write a java program to check whether the given number is odd or even.

Main.java × NewClass.java × cube.java × average.java ×

Source History

```
1   package javaapplication23;
2
3   import java.util.Scanner;
4   public class oddeven {
5
6
7       public static void main(String[] args){
8
9           Scanner sc = new Scanner(System.in) ;
10
11          System.out.println("Enter the  number");
12          int x = sc.nextInt();
13
14          if(x%2 == 0){
15
16              System.out.println("number is even");
17          }
18          else {
19              System.out.println("NUMBER IS ODD");
20      }
21  }
22  }
```

Output:

Output - JavaApplication23 (run)

```
run:
Enter the  number
5
NUMBER IS ODD
BUILD SUCCESSFUL (total time: 26 seconds)
```

Output - JavaApplication23 (run)

```
run:
Enter the  number
6
number is even
BUILD SUCCESSFUL (total time: 2 seconds)
```

2# Write a java program to find the largest number among the three numbers.

Main.java | NewClass.java | cube.java | average.java | distance.java

Source | History

```
1   package javaapplication23;
2
3   import java.util.Scanner;
4   public class LARGEST {
5
6       public static void main(String[] args){
7
8           Scanner sc = new Scanner(System.in) ;
9
10          System.out.println("Enter the a number");
11          int a = sc.nextInt();
12
13          System.out.println("Enter the a number");
14          int b = sc.nextInt();
15
16          System.out.println("Enter the a number");
17          int c = sc.nextInt();
18
19          if((a>b)&&(a>c))
20              System.out.println(a +" is greater");
21          else if((b >a) && (b > c))
22              System.out.println(b+" is greater");
23          else
24              System.out.println(c +" c is greater");
25
26
27  }
28  }
```

Output:

Output - JavaApplication23 (run)

```
run:
Enter the a number
7
Enter the a number
9
Enter the a number

10
10 c is greater
BUILD SUCCESSFUL (total time: 6 seconds)
```

3# Write a Java program that takes a number as input and prints its multiplication table upto 10.

...va | cube.java | average.java | distance.java | oddeven.java | LARG

Source | History

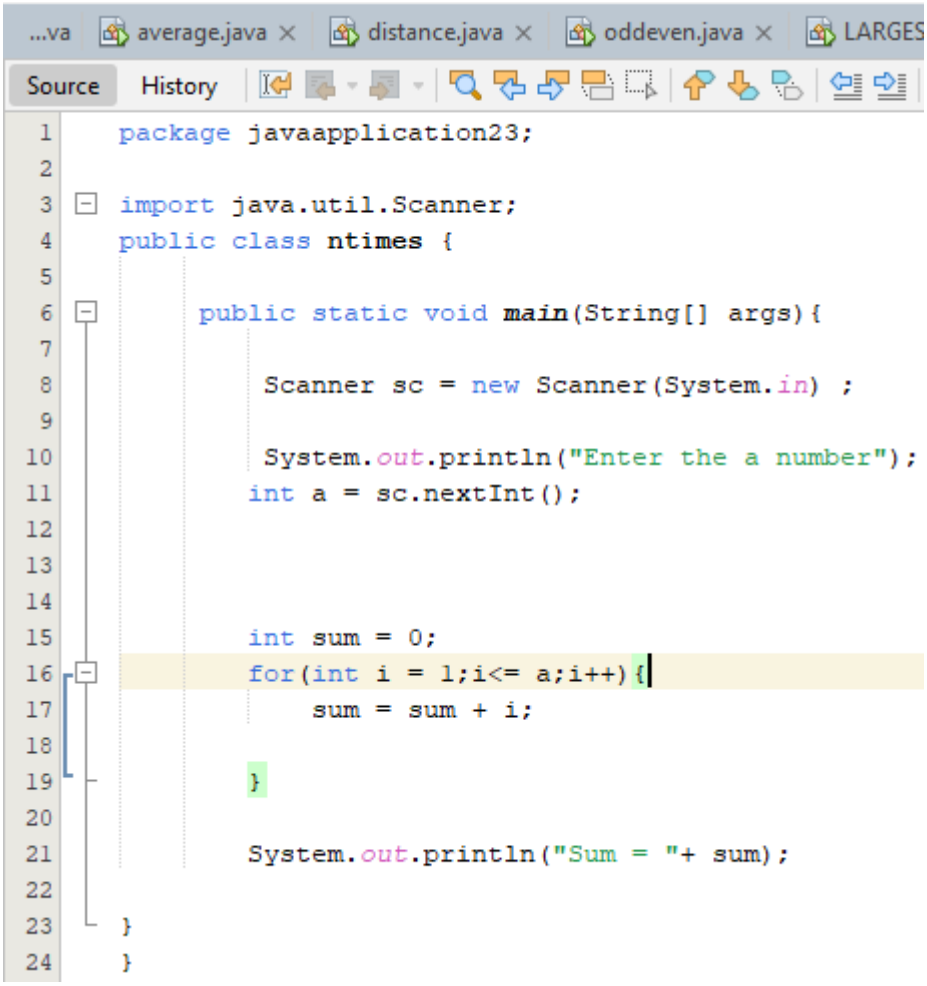
```java
/**
 *
 * @author Admin
 */
import java.util.Scanner;
public class multiplication {


    public static void main(String[] args){

        Scanner sc = new Scanner(System.in) ;

        System.out.println("Enter the a number");
       int a = sc.nextInt();

       for(int i = 1;i<=10;i++){
           int result = a *i;
           System.out.println(a+" x "+i+ " = " +result);
       }

}
}

```

Output:

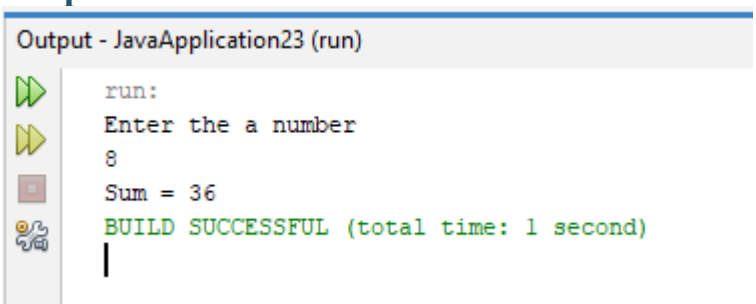
Output - JavaApplication23 (run)

```
run:
Enter the a number
3
3 x 1 = 3
3 x 2 = 6
3 x 3 = 9
3 x 4 = 12
3 x 5 = 15
3 x 6 = 18
3 x 7 = 21
3 x 8 = 24
3 x 9 = 27
3 x 10 = 30
BUILD SUCCESSFUL (total time: 2 seconds)
```

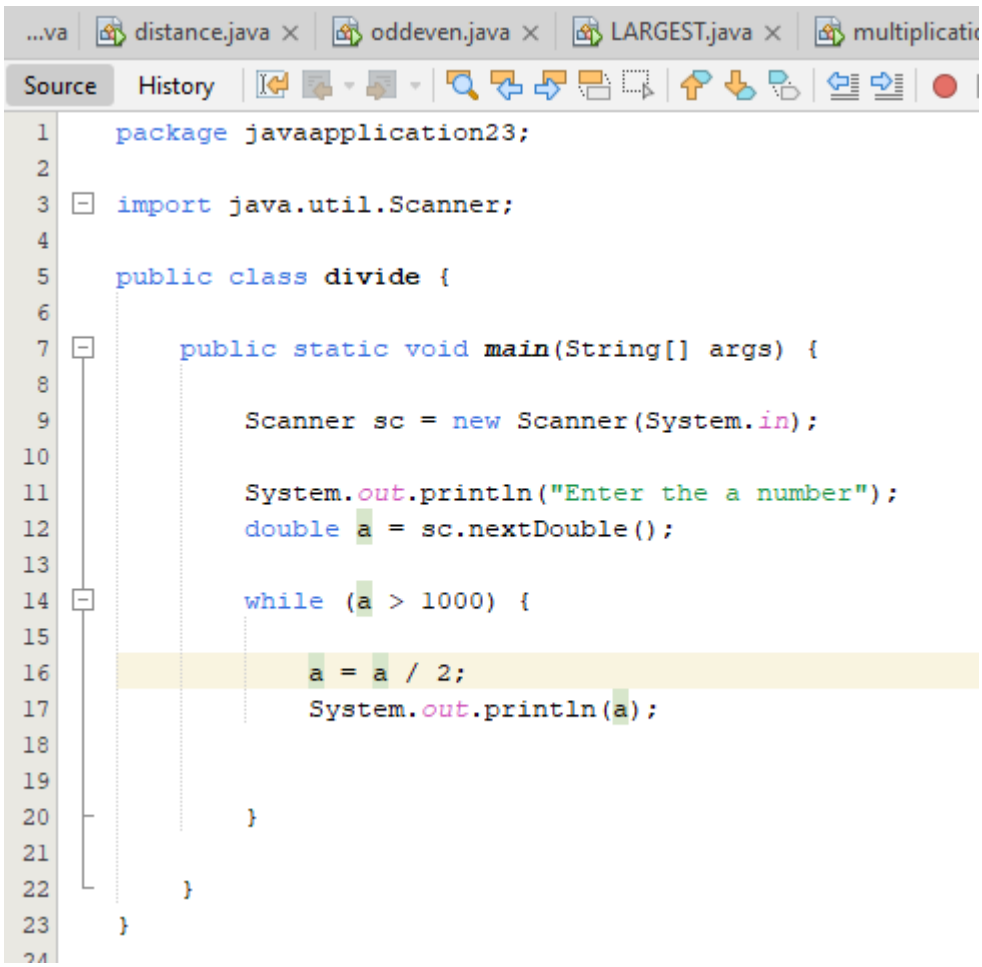
4# Write a Java program to calculate the sum of following series:
1 + 2 + 3 + 4 + ... + N



Output:

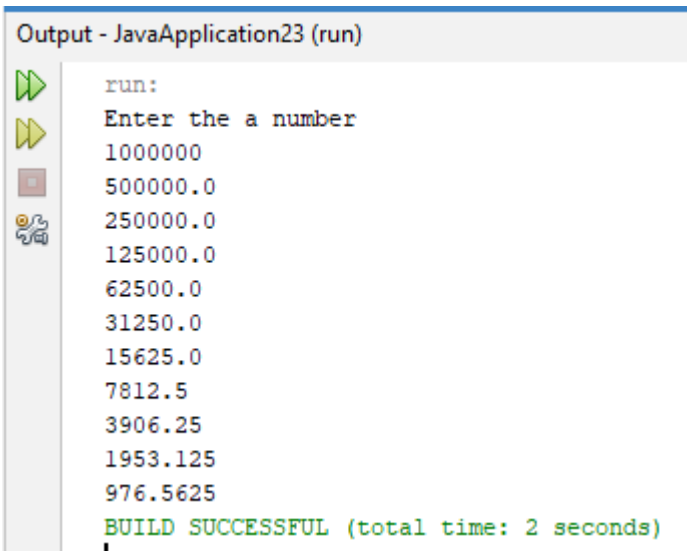


5# Write a Java program to take a number, divide it by 2 and print the result until the number becomes less than 10.



```java
package javaapplication23;

import java.util.Scanner;

public class divide {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.println("Enter the a number");
        double a = sc.nextDouble();

        while (a > 1000) {

            a = a / 2;
            System.out.println(a);


        }

    }
}
```

Output:



```
Output - JavaApplication23 (run)
run:
Enter the a number
1000000
500000.0
250000.0
125000.0
62500.0
31250.0
15625.0
7812.5
3906.25
1953.125
976.5625
BUILD SUCCESSFUL (total time: 2 seconds)
```

Week-4

Problems Based on Array in JAVA

1# Write a Java program to insert 10, 20, 30 .... in an array and display them.

```
Source  History
1   package javaapplication23;
2
3   public class arrayadd {
4
5       public static void main(String[] args) {
6
7           int [] arr = {10,20,30,40,50};
8
9           System.out.println(arr.length);
10
11
12          for(int i = 0;i<arr.length;i++){
13
14              System.out.println(arr[i]);
15          }
16
17
18      }
19
20  }
```

Output:

```
Output - JavaApplication23 (run)
run:
5
10
20
30
40
50
BUILD SUCCESSFUL (total time: 0 seconds)
```

2# Write a Java program to calculate the sum of all the array elements.

Output:

```
Output - JavaApplication23 (run)
run:
5
The Sum is 150
BUILD SUCCESSFUL (total time: 0 seconds)
```

...va | LARGEST.java | multiplication.java | ntimes.java | d

Source | History

```java
package javaapplication23;

public class arraysum {

    public static void main(String[] args) {

        int [] arr = {10,20,30,40,50};

         int sum = 0;

         System.out.println(arr.length);


         for(int i = 0;i<arr.length;i++){

             sum = sum + arr[i];


         }
         System.out.println("The Sum is "+ sum);
    }


}
```

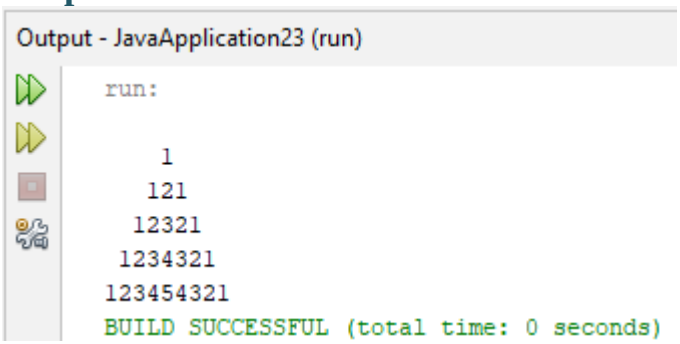
3# Write a java program to print the following pattern.

1
121
12321
1234321
123454321

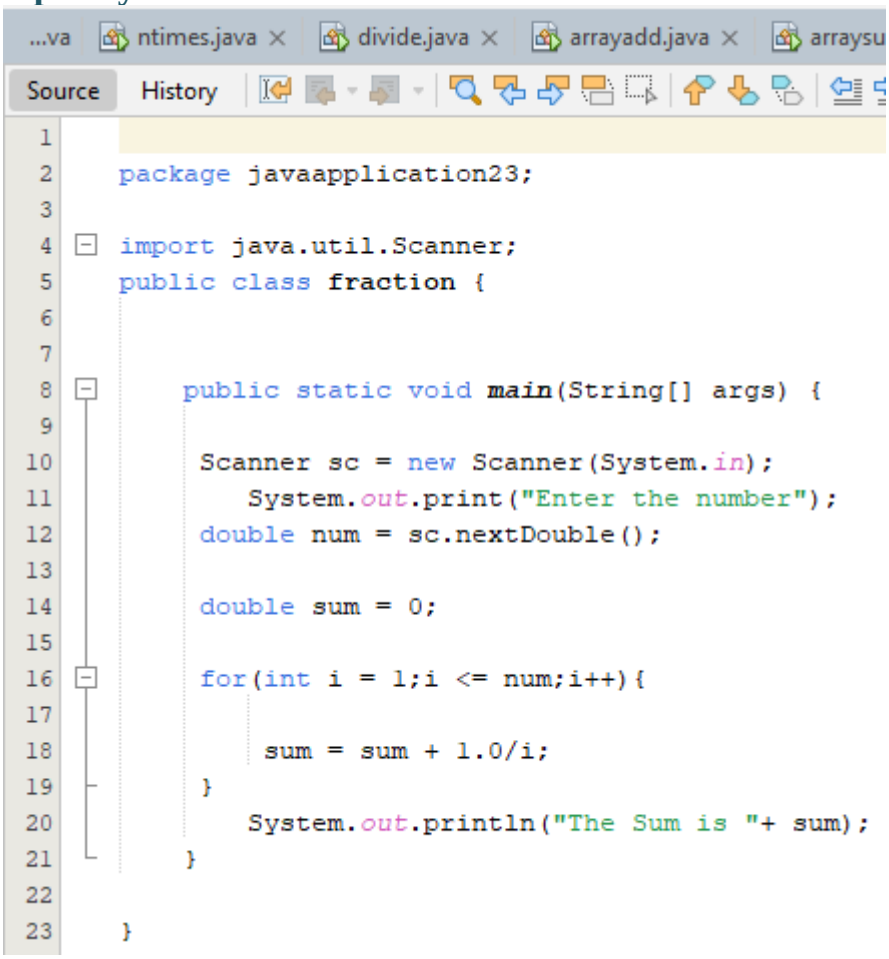
Source | History

```java
package javaapplication23;

public class pattern {
    public static void main(String[] args) {
        int n = 5;
        for(int i = 0; i<=n;i++){
            for(int j=n-i;j>0;j--){
                System.out.print(" ");
            }
            for(int k=1;k<=i;k++){
                    System.out.print(k);
            }
            for(int m=i-1 ; m>0 ; m--){
                System.out.print(m);
            }
            System.out.println();

        }
    }
}
```

Output:

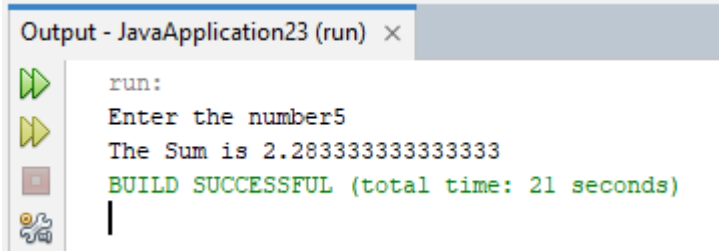


4# Write a java program to find the sum of following series where n is input by the user. 1+1/2+1/3+1/4+.....+1/n.




```java
package javaapplication23;

import java.util.Scanner;
public class fraction {


    public static void main(String[] args) {

     Scanner sc = new Scanner(System.in);
        System.out.print("Enter the number");
     double num = sc.nextDouble();

     double sum = 0;

     for(int i = 1;i <= num;i++){

         sum = sum + 1.0/i;
     }
        System.out.println("The Sum is "+ sum);
    }

}
```

Output:



5# Write a Java program and compute the sum of the digits of an integer.

```java
package javaapplication23;
import java.util.Scanner;
public class digitsum {


    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.println("Enter the number");
        int num = sc.nextInt();

        int digit = 0;
        int sum = 0;

        while(num > 0)
        {
            digit = num % 10;

            sum = sum + digit;

            num = num/ 10;
        }

        System.out.println("Thw Sum is"+ sum);
    }

}
```

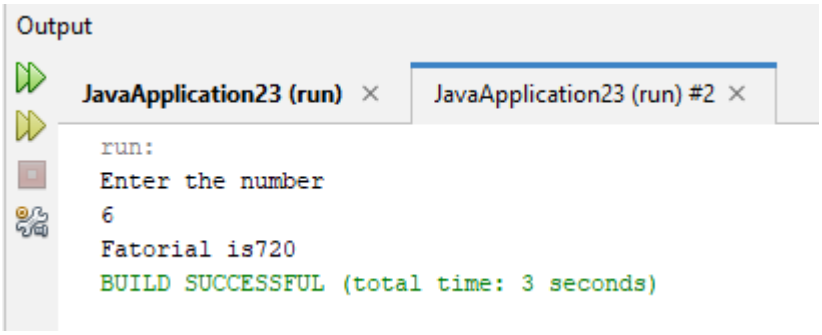
Output:

```
Output - JavaApplication23 (run)
run:
Enter the number
127
Thw Sum is10
BUILD SUCCESSFUL (total time: 4 seconds)
```

6# Write a Java program to calculate the factorial of a number.

```java

package javaapplication23;

import java.util.Scanner;
public class fatorial {

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.println("Enter the number");
        int num = sc.nextInt();

        int fact = 1;

        for(int i = 1;i <= num; i++){

            fact  = fact * i;

        }
        System.out.println("Fatorial is"+ fact);
    }

}

```

Output:



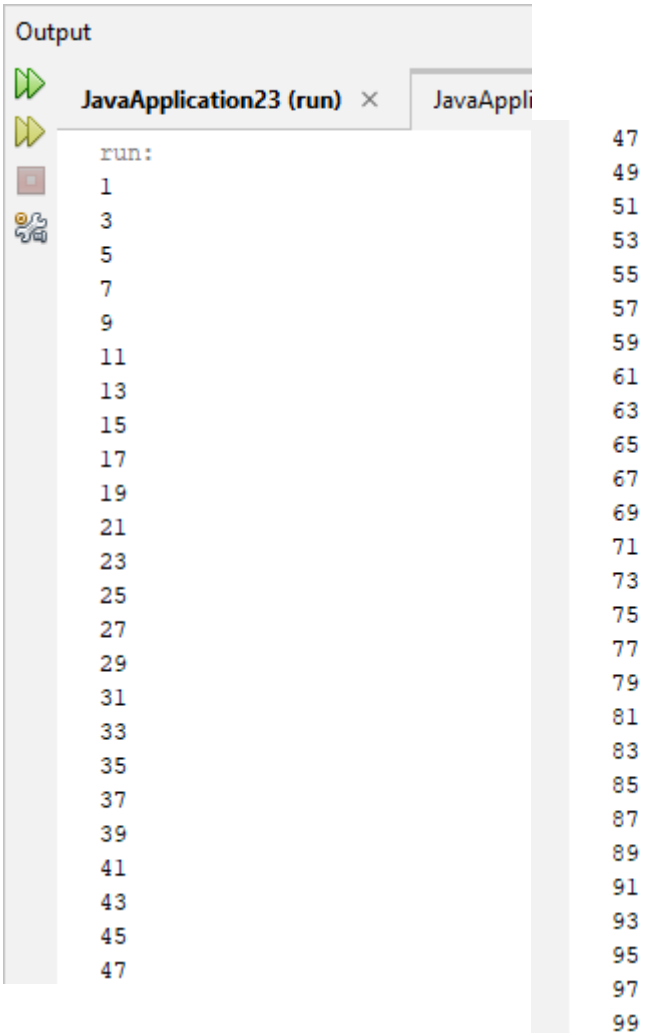
Week-5

Problems Based on If statement/Looping/Array in JAVA

1# Write a Java program to print the odd numbers from 1 to 99.

```
package javaapplication23;

public class oddnum99 {

    public static void main(String[] args) {

        for (int i = 1; i < 100; i++) {
            if (i % 2 != 0) {
                System.out.println(i);

            }
        }
    }

}
```

Output:



2# Write a Java program to check whether a number is prime or not.

```java

package javaapplication23;
import java.util.Scanner;
public class prime {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);
        int num = sc.nextInt();

        boolean flag = true;

        for (int i = 2; i <= num / 2; i++) {
            if (num % i == 0) {
                flag = false;

            }
        }
        if (flag)
            System.out.println(num +" is a prime");
        else
            System.out.println(num +" is not");
    }
}
```

Output:

```
Output
JavaApplication23 (run) ×   JavaApplication23 (run)
run:
7
7 is a prime
BUILD SUCCESSFUL (total time: 2 seconds)
```

```
Output
JavaApplication23 (run) ×   JavaApplication23 (run) #2 ×
run:
6
6 is not
BUILD SUCCESSFUL (total time: 3 seconds)
```

3# Write a Java program to swap the first and last elements of an array.

```
public class swaparray {
    public static void main(String[] args){
        Scanner sc =  new Scanner(System.in);
        System.out.println("enter the size of the arr
        int n = sc.nextInt();

        int arr[] = new int[n];
        for(int i =0 ; i<arr.length ; i++){
            System.out.println("enter the elements");
            arr[i] = sc.nextInt();
        }
        for(int element : arr){
            System.out.println(element + "\t");
        }

        int temp = arr[0];
        arr[0] = arr[arr.length - 1];
        arr[arr.length - 1] = temp;

        System.out.println("array after swapping");

        for(int element : arr){

            System.out.println(element + "\t");
        }
    }
}
```

Output:

```
Output
JavaApplication23 (run) ×   JavaApplication23 (run) #2 ×
run:
enter the size of the array
4
enter the elements
6
enter the elements
7
enter the elements
9
enter the elements
12
6
7
9
12
array after swapping
12
7
9
6
BUILD SUCCESSFUL (total time: 14 seconds)
```

4# Write a Java program to find the maximum and minimum among array elements.

```java
package javaapplication23;
import java.util.Scanner;
public class maxmin {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter the number of elements in the array: ");
        int n = scanner.nextInt();

        int[] arr = new int[n];

        System.out.println("Enter the elements of the array:");
        for (int i = 0; i < n; i++) {
            arr[i] = scanner.nextInt();
        }

        int max = arr[0];
        int min = arr[0];

        for (int i = 1; i < n; i++) {
            if (arr[i] > max) {
                max = arr[i];
            }
            if (arr[i] < min) {
                min = arr[i];
            }
        }

        System.out.println("Maximum element: " + max);
        System.out.println("Minimum element: " + min);

        scanner.close();
    }
}
```

Output

```
Output - JavaApplication23 (run) #2
run:
Enter the number of elements in the array: 4
Enter the elements of the array:
4
6
8
9
Maximum element: 9
Minimum element: 4
BUILD SUCCESSFUL (total time: 10 seconds)
```

5# Write a Java program to print all prime numbers between 0 to 100

```java
package javaapplication23;

public class prime0to100 {

    public static void main(String[] args) {
        System.out.println("Pime numbers between 0 to 100:");

        for (int num = 2; num < 100; num++) {
            boolean isprime = true;
            for(int i = 2;i<=Math.sqrt(num);i++){
                if(num%i==0){
                    isprime =false;
                    break;
                }
            }
            if (isprime) {
                System.out.println(num);
            }

        }
    }
}
```

Output:

```
Output - JavaApplication23 (run)
run:
Pime numbers between 0 t
2
3
5
7
11
13
17
19
23
29
31
37
41
43
47
53
59
61
67
71
73
79
83
89
97
```

6# Write a Java program to implement linear search.

```java
package javaapplication23;
import java.util.Scanner;
public class linearsearh {

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        int[] arr = {5, 8, 12, 15, 23, 32, 45, 56, 67, 78};

        System.out.print("Enter the element to search: ");
        int target = sc.nextInt();

        int index = -1;

        for (int i = 0; i < arr.length; i++) {
            if (arr[i] == target) {
                index = i;
                break;
            }
        }
              if (index != -1) {
            System.out.println("Element " + target + " found at index " + inc  ;
        } else {
            System.out.println("Element " + target + " not found.");
        }

    }
}
```

Output: