

Input and Output Statements in Java

Aim:

To understand and use input and output statements in Java using the Scanner class.

PRE LAB EXERCISE

QUESTIONS

1. What is input in a program?

Input is any data or signal sent to a program from an external source, like a keyboard or sensor. It acts as the raw material that the program's logic processes.

2. What is output in a program?

Output is the final result or processed data sent from the program to a user or device, such as text on a screen. It represents the "answer" or action derived from the input.

3. Which class is used to read input from the user in Java?

The **java.util.Scanner** class is used to read input from the user through the keyboard. It can read numbers and text using methods like `nextInt()` and `nextLine()`.

IN LAB EXERCISE

Objective:

To read input from the user and display the output using Java input and output statements.

INPUT STATEMENT:

SCANNER CLASS

- ✓ The Scanner class in Java is used to read input from the user through the keyboard. It is available in the package `java.util`.
- ✓ The Scanner object reads different types of input such as integer, float, double, and string and stores them in variables.
- ✓ To use the Scanner class, it must be imported before using it in the program.

SYNTAX:

- ✓ `Scanner sc = new Scanner(System.in);`

Commonly Used Scanner Methods:

- ✓ nextInt() – reads an integer value
- ✓ nextFloat() – reads a float value
- ✓ nextDouble() – reads a double value
- ✓ next() – reads a single word
- ✓ nextLine() – reads a complete line of text

PROGRAMS:

Program 1: Read and Display Name

Source Code:

```
import java.util.Scanner;

class ReadName {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.print("Enter your name: ");

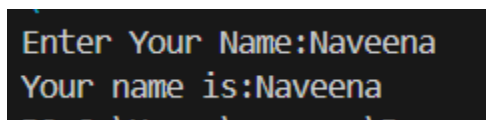
        String name = sc.nextLine();

        System.out.println("Your name is: " + name);

    }

}
```

Output:



```
Enter Your Name:Naveena
Your name is:Naveena
```

Program 2: Read Two Numbers and Print Sum

Source Code:

```
import java.util.Scanner;

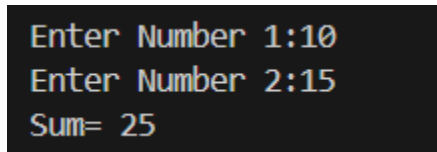
class SumInput {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);
```

```
System.out.print("Enter first number: ");
int a = sc.nextInt();
System.out.print("Enter second number: ");
int b = sc.nextInt();
int sum = a + b;
System.out.println("Sum = " + sum);
}
}
```

Output:

A screenshot of a terminal window with a dark background. It shows the output of the first program: "Enter Number 1:10", "Enter Number 2:15", and "Sum= 25".

```
Enter Number 1:10
Enter Number 2:15
Sum= 25
```

Program 3: Read Length and Breadth and Find Area of Rectangle

Source Code:

```
import java.util.Scanner;
class AreaRectangleInput {
public static void main(String[] args) {
Scanner sc = new Scanner(System.in);
System.out.print("Enter length: ");
int length = sc.nextInt();
System.out.print("Enter breadth: ");
int breadth = sc.nextInt();
int area = length * breadth;
System.out.println("Area of Rectangle = " + area);
}
}
```

Output:

```
Enter Length:20
Enter Breadth 2:6
Sum= 120
```

POST LAB EXERCISE

- ✓ What is the use of the Scanner class in Java?

The Scanner class is used to read input from the user through the keyboard. It can read different types of data such as integers, floating-point numbers, and strings.

- ✓ Which package contains the Scanner class?

The Scanner class is available in the `java.util` package, so it must be imported before use.

- ✓ What does `System.in` represent?

`System.in` represents the standard input stream in Java. It is used to take input from the keyboard.

- ✓ What does `System.out` represent?

`System.out` represents the standard output stream. It is used to display output on the screen using methods like `print()` and `println()`.

- ✓ Name any two methods used to read input using Scanner.

- `nextInt()` – used to read an integer value
- `nextLine()` – used to read a complete line of text

- ✓ What is the difference between `next()` and `nextLine()`?

The `next()` method reads only a single word and stops at space, whereas `nextLine()` reads the entire line of input including spaces.

Result:

Thus the input was successfully read from the user and the corresponding output was displayed using Java input and output statements.

ASSESSMENT

Description	Max Marks	Marks Awarded
Pre Lab Exercise	5	
In Lab Exercise	10	
Post Lab Exercise	5	
Viva	10	
Total	30	
Faculty Signature		