

Control Statements in Java

Aim:

To understand and implement decision-making and looping control statements in Java.

PRE LAB EXERCISE

QUESTIONS

- ✓ List different control statements in Java.

1. Decision-making statements

- if
- if-else
- else-if
- switch

2. Looping statements

- for
- while
- do-while

3. Jump statements

- break
- continue
- return

- ✓ Difference between for, while, and do-while loops.

Feature	for loop	while loop	do-while loop
Condition check	Before loop starts	Before loop starts	After loop ends
Minimum execution	May not execute	May not execute	Executes at least once
Best used when	Number of iterations is known	Iterations not known	Loop must run at least once

- ✓ What is the use of break and continue?

Break:

Used to **terminate the loop or switch statement** immediately.

Continue:

Used to **skip the current iteration** and move to the **next iteration** of the loop.

IN LAB EXERCISE

Objective:

To implement if-else and looping 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:

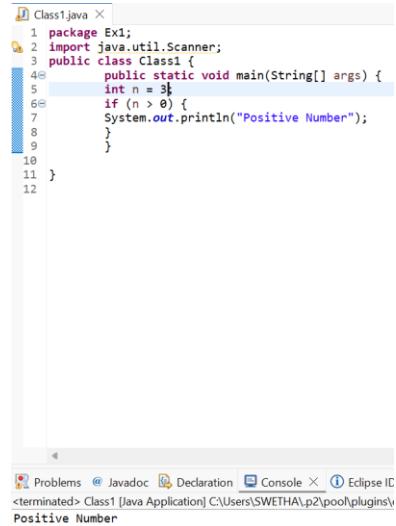
Program 1: Check Whether a Number is Positive

```
class PositiveNumber {  
    public static void main(String[] args) {  
        int n = 5;  
        if (n > 0) {
```

```
System.out.println("Positive Number");
}
}
}
```

Output:

Positive Number



The screenshot shows the Eclipse IDE interface. On the left, the code editor displays `Class1.java` with the following content:

```
1 package Ex1;
2 import java.util.Scanner;
3 public class Class1 {
4     public static void main(String[] args) {
5         int n = 3;
6         if (n > 0) {
7             System.out.println("Positive Number");
8         }
9     }
10 }
11 }
```

On the right, the `Console` tab is active, showing the output: `Positive Number`.

Program 2: Check Whether a Number is Even or Odd

```
class EvenOdd {
    public static void main(String[] args) {
        int n = 6;
        if (n % 2 == 0)
            System.out.println("Even Number");
        else
            System.out.println("Odd Number");
    }
}
```

Output:

Odd Number

The screenshot shows the Eclipse IDE interface. In the top window, titled 'Class1.java ×', there is a code editor with the following Java code:

```
1 package Ex1;
2 import java.util.Scanner;
3 public class Class1 {
4     public static void main(String[] args) {
5         int n = 5;
6         if (n % 2 == 0)
7             System.out.println("Even Number");
8         else
9             System.out.println("Odd Number");
10    }
11 }
12
```

In the bottom window, titled 'Console ×', the output of the program is displayed:

```
Odd Number
```

Program 3: Find Largest of Two Numbers

```
class LargestTwo {
    public static void main(String[] args) {
        int a = 10, b = 20;
        if (a > b)
            System.out.println("A is largest");
        else
            System.out.println("B is largest");
    }
}
```

Output:

A is largest

The screenshot shows the Eclipse IDE interface. At the top, there's a tab labeled "Class1.java X". Below it is the Java code:

```
1 package Ex1;
2 import java.util.Scanner;
3 public class Class1 {
4     public static void main(String[] args) {
5         int a = 20, b = 15;
6         if (a > b)
7             System.out.println("A is largest");
8         else
9             System.out.println("B is largest");
10    }
11 }
12
```

At the bottom of the screen, the "Console" view is open, showing the output of the program:

```
Problems @ Javadoc Declaration Console X Eclipse |<terminated> Class1 [Java Application] C:\Users\SWETHA\p2\pooh\plugins
A is largest
```

Program 4: Grade Calculation

```
class Grade {  
    public static void main(String[] args) {  
        int marks = 75;  
        if (marks >= 90)  
            System.out.println("Grade A");  
        else if (marks >= 75)  
            System.out.println("Grade B");  
        else if (marks >= 50)  
            System.out.println("Grade C");  
        else  
            System.out.println("Fail");  
    }  
}
```

Output:

Grade C

The screenshot shows the Eclipse IDE interface. The top window displays a Java file named 'Class1.java' with the following code:

```
1 package Ex1;
2 import java.util.Scanner;
3 public class Class1 {
4     public static void main(String[] args) {
5         int marks = 55;
6         if (marks >= 90)
7             System.out.println("Grade A");
8         else if (marks >= 75)
9             System.out.println("Grade B");
10        else if (marks >= 50)
11            System.out.println("Grade C");
12        else
13            System.out.println("Fail");
14    }
15}
16
```

The line 'System.out.println("Grade C");' is highlighted in blue. Below this window is the Eclipse IDE toolbar with tabs for Problems, Javadoc, Declaration, Console, and Eclipse IDE. The 'Console' tab is selected, showing the output: '<terminated> Class1 [Java Application] C:\Users\SWETHA\p2\pool\plugins\or Grade C'. The status bar at the bottom also displays 'Grade C'.

Program 5: Day of the Week

```
class DaySwitch {
    public static void main(String[] args) {
        int day = 3;
        switch (day) {
            case 1: System.out.println("Monday"); break;
            case 2: System.out.println("Tuesday"); break;
            case 3: System.out.println("Wednesday"); break;
            case 4: System.out.println("Thursday"); break;
            case 5: System.out.println("Friday"); break;
            default: System.out.println("Invalid Day");
        }
    }
}
```

Output:

Thursday

The screenshot shows the Eclipse IDE interface. The top part displays a code editor for 'Class1.java' with the following content:

```
1 package Ex1;
2 import java.util.Scanner;
3 public class Class1 {
4     public static void main(String[] args) {
5         int day = 4;
6         switch (day) {
7             case 1: System.out.println("Monday"); break;
8             case 2: System.out.println("Tuesday"); break;
9             case 3: System.out.println("Wednesday"); break;
10            case 4: System.out.println("Thursday"); break;
11            case 5: System.out.println("Friday"); break;
12            default: System.out.println("Invalid Day");
13        }
14    }
15 }
```

The bottom part shows the 'Console' tab with the output: 'Thursday'. The status bar at the bottom indicates the application is terminated.

Program 6: Print Numbers from 1 to 5

```
class ForLoop {

public static void main(String[] args) {

for (int i = 1; i <= 5; i++) {

System.out.println(i);

}

}

}
```

Output:

```
1
2
3
4
5
```

```
Class1.java ×
1 package Ex1;
2 import java.util.Scanner;
3 public class Class1 {
4     public static void main(String[] args) {
5         for (int i = 1; i <= 5; i++) {
6             System.out.println(i);
7         }
8     }
9 }
10
```

```
Problems @ Javadoc Declaration Console × Eclipse IDE
<terminated> Class1 [Java Application] C:\Users\SWETHA\p2\pool\plugins\
1
2
3
4
5
```

Program 7: Print Numbers from 1 to 5

```
class WhileLoop {
    public static void main(String[] args) {
        int i = 1;
        while (i <= 5) {
            System.out.println(i);
            i++;
        }
    }
}
```

Output:

```
1
2
3
4
5
```

The screenshot shows the Eclipse IDE interface. The top window displays the Java code for 'Class1.java'. The code defines a class 'Class1' with a main method that prints integers from 1 to 5. The bottom window shows the 'Console' tab with the output of the program, which is the numbers 1 through 5.

```
1 package Ex1;
2 import java.util.Scanner;
3 public class Class1 {
4     public static void main(String[] args) {
5         int i = 1;
6         while (i <= 5) {
7             System.out.println(i);
8             i++;
9         }
10    }
11 }
```

```
1
2
3
4
5
```

Program 8: Print Numbers from 1 to 5

```
class DoWhileLoop {
    public static void main(String[] args) {
        int i = 1;
        do {
            System.out.println(i);
            i++;
        } while (i <= 5);
    }
}
```

Output:

```
1
2
3
4
5
```

```
1 package Ex1;
2 import java.util.Scanner;
3 public class Class1 {
4     public static void main(String[] args) {
5         int i = 1;
6         do {
7             System.out.println(i);
8             i++;
9         } while (i <= 5);
10    }
11 }
```

Problems Javadoc Declaration Console Eclipse IDE for Java Developers

Class1 [Java Application] C:\Users\SWETHA\p2\pool\plugins\org.eclipse.jdt.jdt

1
2
3
4
5

Program 9: Sum of First 5 Natural Numbers

```
class SumNumbers {
    public static void main(String[] args) {
        int sum = 0;
        for (int i = 1; i <= 5; i++) {
            sum = sum + i;
        }
        System.out.println("Sum = " + sum);
    }
}
```

Output:

Sum = 15

```
1 package Ex1;
2 import java.util.Scanner;
3 public class Class1 {
4     public static void main(String[] args) {
5         int sum = 0;
6         for (int i = 1; i <= 5; i++) {
7             sum = sum + i;
8         }
9         System.out.println("Sum = " + sum);
10    }
11 }
```

Problems Javadoc Declaration Console Eclipse IDE for Java Developers

Class1 [Java Application] C:\Users\SWETHA\p2\pool\plugins\org.e

Sum = 15

Program 10: Multiplication Table of a Number

```
class MultiplicationTable {  
    public static void main(String[] args) {  
        int n = 5;  
        for (int i = 1; i <= 10; i++) {  
            System.out.println(n + " x " + i + " = " + (n * i));  
        }  
    }  
}
```

Output:

5 x 1 = 5

5 x 2 = 10

5 x 3 = 15

5 x 4 = 20

5 x 5 = 25

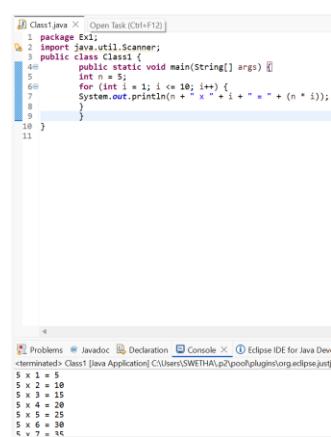
5 x 6 = 30

5 x 7 = 35

5 x 8 = 40

5 x 9 = 45

5 x 10 = 50



The screenshot shows the Eclipse IDE interface. The top part displays the Java code for generating a multiplication table. The bottom part shows the 'Console' tab where the program's output is displayed, listing the multiplication results from 5x1 to 5x10.

```
1 package Ex1;  
2 import java.util.Scanner;  
3 public class Class1 {  
4     public static void main(String[] args) {  
5         int n = 5;  
6         for (int i = 1; i <= 10; i++) {  
7             System.out.println(n + " x " + i + " = " + (n * i));  
8         }  
9     }  
10 }
```

5 x 1 = 5
5 x 2 = 10
5 x 3 = 15
5 x 4 = 20
5 x 5 = 25
5 x 6 = 30
5 x 7 = 35
5 x 8 = 40
5 x 9 = 45
5 x 10 = 50

POST LAB EXERCISE

- ✓ What is the use of if statement?

The **if statement** is used to **test a condition** and execute a block of code **only if the condition is true**.

- ✓ Difference between if-else and else-if ladder.

if-else

Checks **one condition**

Has only two blocks

Used for simple decisions

Example: pass or fail

else-if ladder

Checks **multiple conditions**

Has multiple conditions

Used for multiple choices

Example: grade system

- ✓ Why is switch statement used?

The **switch statement** is used to **select one block of code from many options** based on the value of a variable or expression.

- ✓ Difference between for, while, and do-while loops.

Feature	for	while	do-while
Condition check	Before loop	Before loop	After loop
Minimum execution	Zero times	Zero times	At least once
Best used when	Iterations known	Iterations unknown	Must run once

- ✓ Which loop executes at least once?

The **do-while loop** executes **at least once** because the condition is checked **after** executing the loop body.

Result:

Thus the different control statements were executed successfully with expected output.

ASSESSMENT

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