

# SNS COLLEGE OF TECHNOLOGY



An Autonomous Institution  
Coimbatore-35

## Department of Computer Science & Engineering

**23ITT202 – Object Oriented Programming**

I B.E CSE/ II SEMESTER

## **UNIT I :Introduction To OOPS**

**Topic 2 : Comments in Java**



# Topics for Discussion

- ✓ Introduction to Comments
- ✓ Meaning of Comments
- ✓ Importance of Comments
- ✓ Advantages of Comments
- ✓ Types of Comments in Java
- ✓ Syntax and Examples
- ✓ Mind Map & Assignment

```

import java.util.Scanner;
public class ArraySwitchExample {
public static void main(String[] args) {
// ===== ARRAY DECLARATION =====
// Array to store 5 student marks
int[] marks = {85, 72, 90, 66, 58};
Scanner sc = new Scanner(System.in);
// Display menu
System.out.println("===== MENU =====");
System.out.println("1. Display All Marks");
System.out.println("2. Display Highest Mark");
System.out.println("3. Display Lowest Mark");
System.out.println("4. Display Average Marks");
System.out.println("Enter your choice (1-4): ");
int choice = sc.nextInt();
// ===== SWITCH CASE =====
switch (choice) {

```

case 1:

// Display all elements of array

```

System.out.println("Student Marks:");
for (int i = 0; i < marks.length; i++) {
System.out.println("Mark " + (i + 1) + ": " +
marks[i]);
}
break;

```

case 2:

// Find highest mark

```

int max = marks[0];
for (int i = 1; i < marks.length; i++) {
if (marks[i] > max) {
max = marks[i];
}}
System.out.println("Highest Mark: " +
max);
break;

```

case 3:

// Find lowest mark

```

int min = marks[0];
for (int i = 1; i < marks.length; i++) {
if (marks[i] < min) {

```

- case 4:  
// Calculate average  
int sum = 0;  
for (int i = 0; i < marks.length; i++) {  
sum += marks[i];  
}  
double average = (double) sum / marks.length;  
System.out.println("Average Marks: " +  
average);  
break;  
  
default:  
System.out.println("Invalid Choice!");  
}  
  
sc.close();  
}

### Sample Output (Example)

===== MENU =====

1. Display All Marks
2. Display Highest Mark
3. Display Lowest Mark
4. Display Average Marks

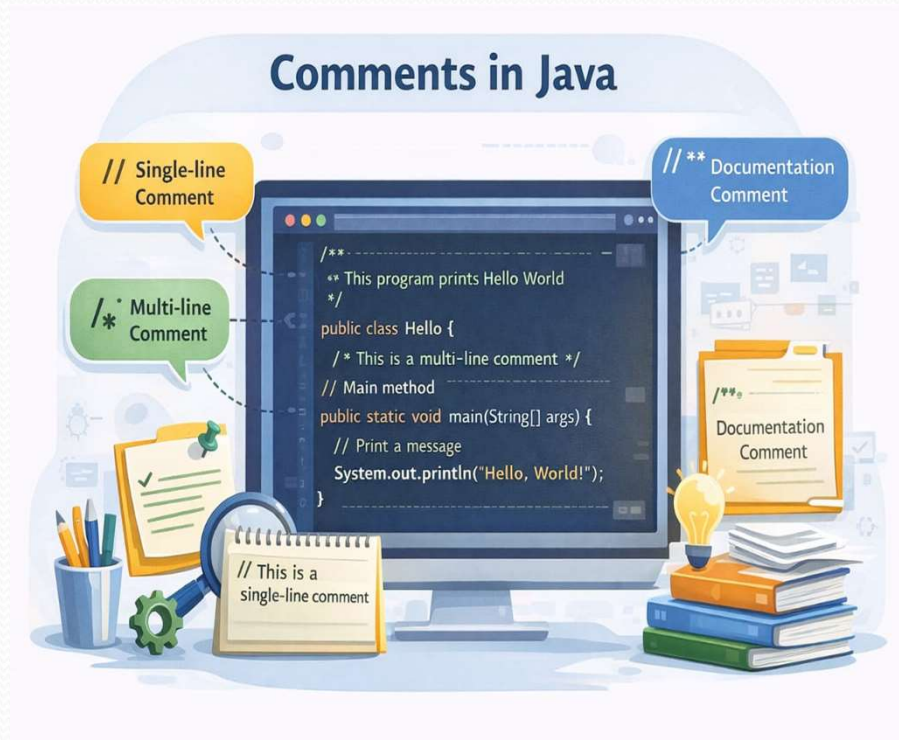
Enter your choice (1-4): 2

Highest Mark: 90

- **Explanation:**
- **Array** → `int[] marks = {85, 72, 90, 66, 58};`
  - Stores multiple values in a single variable.
  - Accessed using index numbers (0 to length-1).
- **Switch Case**
  - Used to perform different operations based on user choice.
  - Each case performs a different array operation.
  - `break` prevents execution from continuing to other cases.

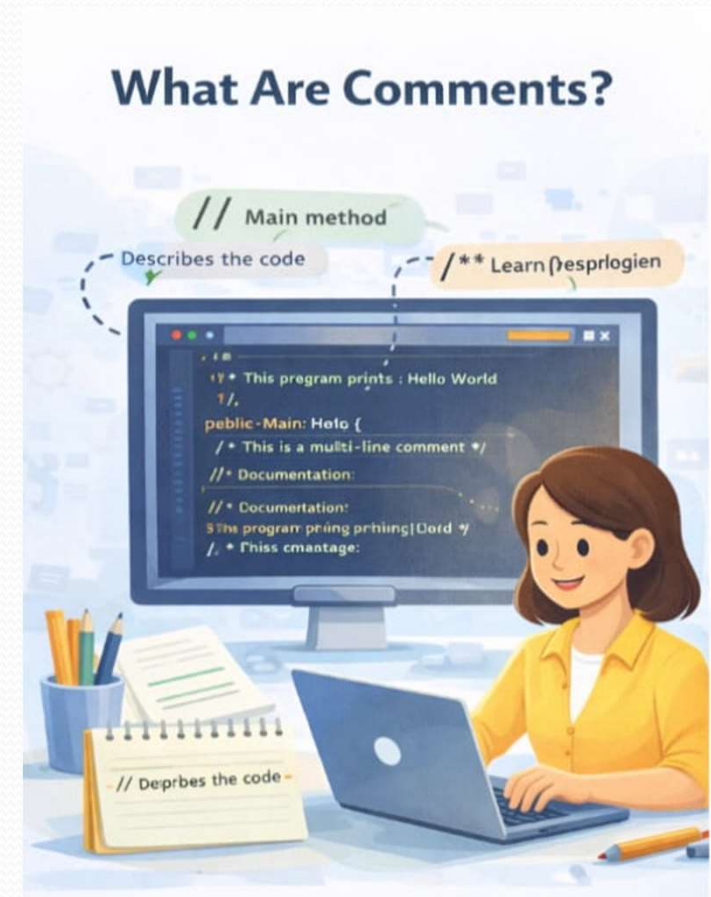
# Introduction to Comments

- Comments are explanatory statements written inside Java programs.
- They help programmers understand the program logic.
- Comments make programs easier to read and learn.
- They are mainly used during development and learning



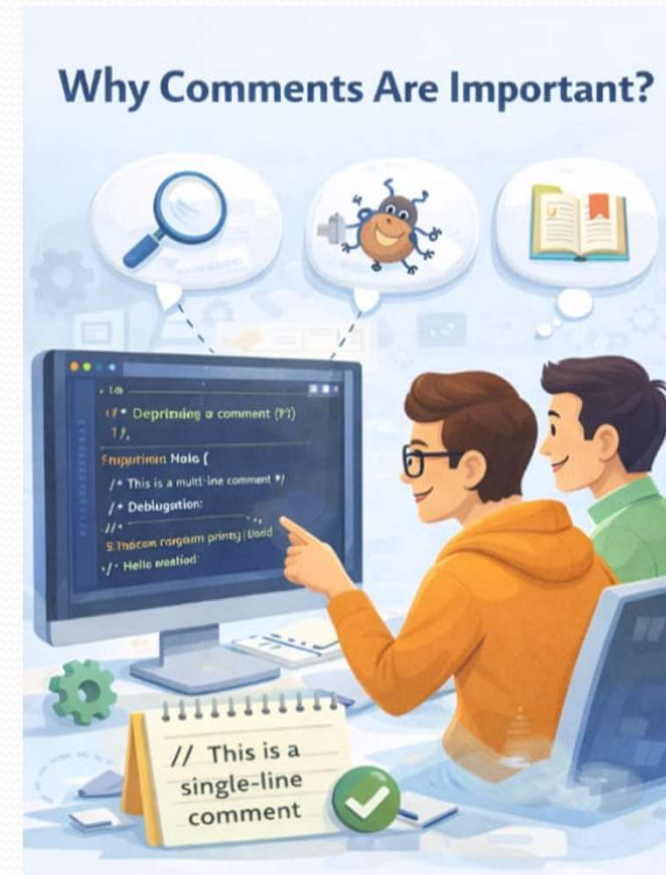
# What Are Comments?

- Comments are non-executable lines in Java source code.
- The Java compiler ignores comments during execution.
- Comments do not affect program output.
- They are used to describe the working of code



# Why Comments Are Important?

- Improves readability of programs.
- Helps beginners understand coding logic.
- Makes debugging and modification easier.
- Acts as documentation for future use



# Advantages Of Comments

- Reduces confusion in large programs.
- Helps teamwork in software development.
- Saves time during maintenance.
- Useful for teachers, students, and developers.

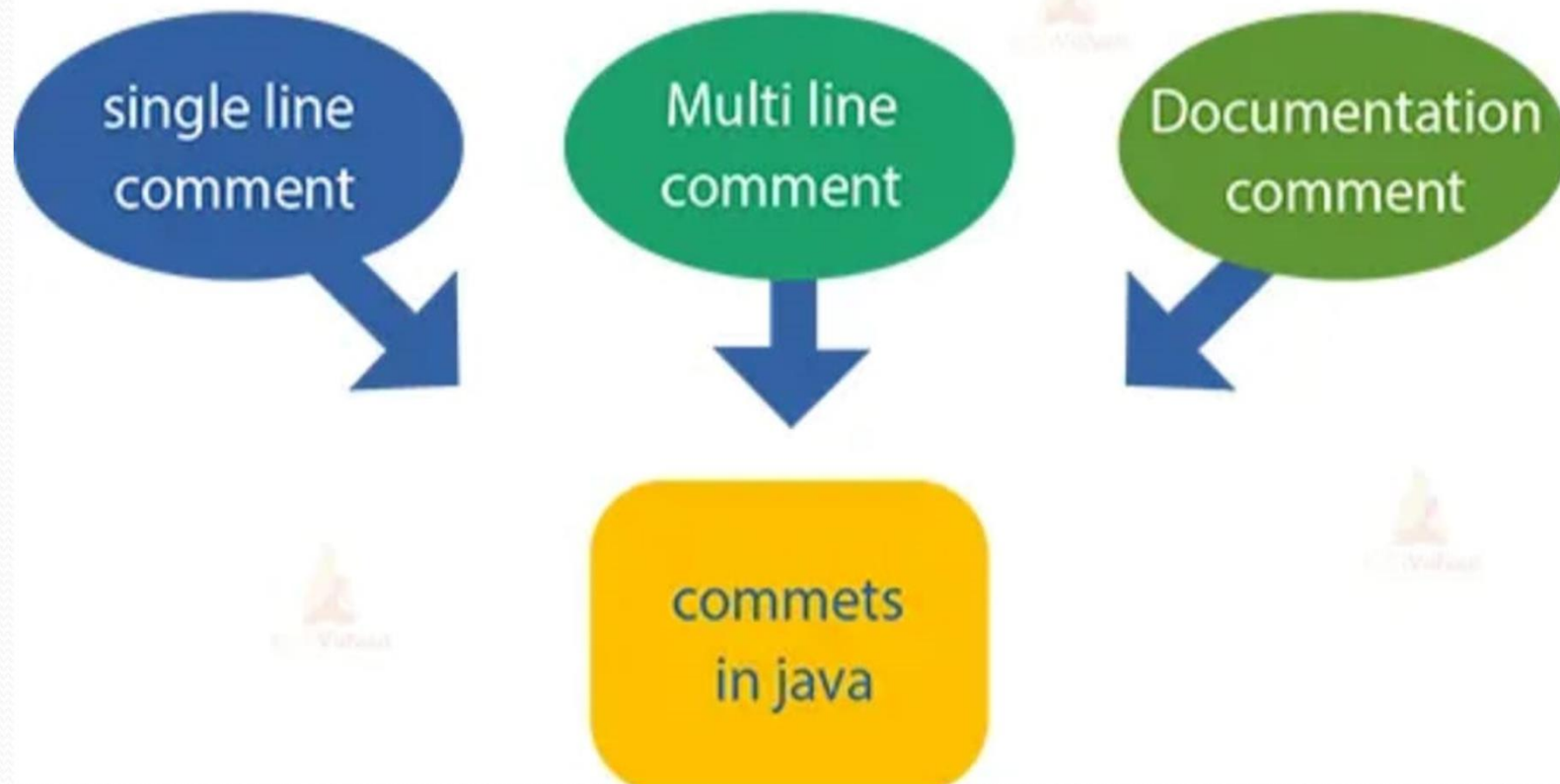
# Types of Comments In Java

- Java supports three types of comments.
- Each type is used for a specific purpose.
- Proper usage improves code quality.

## Types:

1. Single-line Comment.
2. Multi-line Comment.
3. Documentation Comment.

# Type of comments in java chart

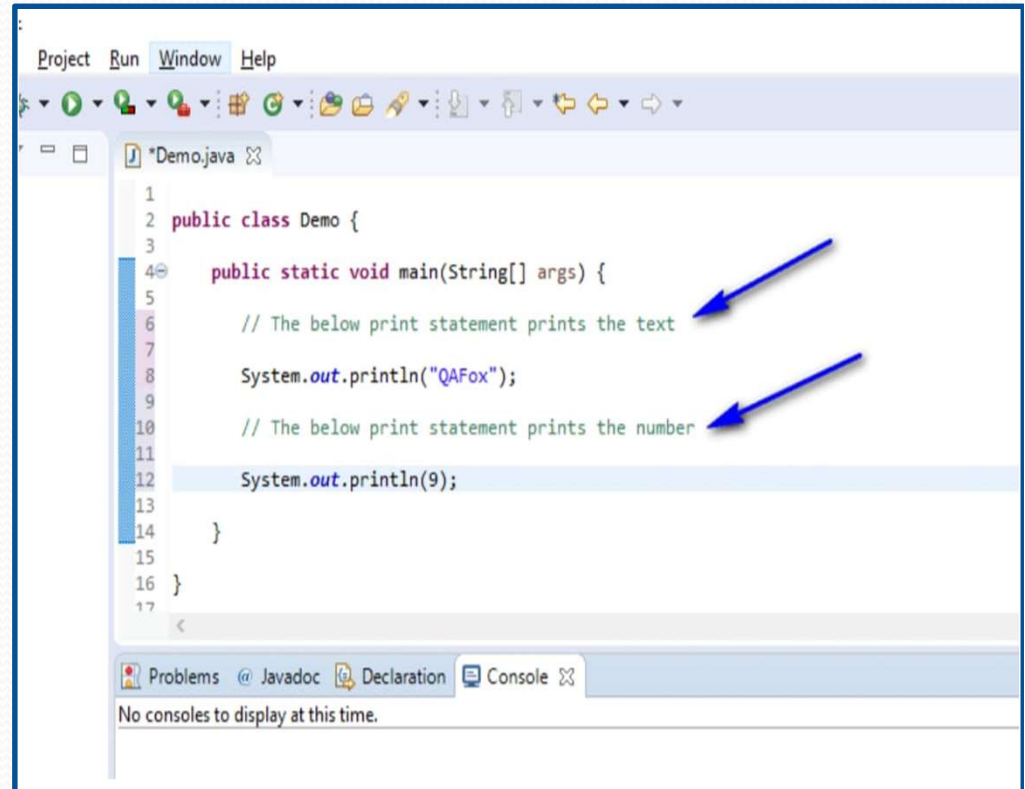


# Single-line Comment

- Begins with //
- Used for short explanations.
- Commonly placed above or beside code.
- Ignored by the compiler

## Syntax:

// This is a single-line comment



```
1  
2 public class Demo {  
3  
4     public static void main(String[] args) {  
5  
6         // The below print statement prints the text  
7  
8         System.out.println("QAfox");  
9  
10        // The below print statement prints the number  
11  
12        System.out.println(9);  
13  
14    }  
15  
16 }  
17
```

# Multi-line Comment

- Starts with /\* and ends with \*/.
- Used for long explanations.
- Can cover multiple lines.
- Mostly used at the beginning of programs

## Syntax:

/\* This is a multi-line comment \*/

```

1 package exp;
2 public class CommentsInJava {
3     public static void main(String[] args) {
4         String greetings[] = { "welcome", "to", "javabeat.net", "let's", "learn", "java" };
5         /*
6          System.out.println("The Size of Given Array is ==> " + greetings.length);
7          for (int i=0; i<greetings.length; i++)
8          {
9              System.out.println("Array Element at index " + i + " is :" + greetings[i]);
10         }
11         */
12         String str = String.join(" ", greetings);
13         System.out.println("The Concatenated String ==> " + str);
14     }
15 }
    
```

Multi-Line Comment

```

<terminated> CommentsInJava [Java Application] C:\Users\HP\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_17.0.7.v20230425-1502\jre\bin
The Concatenated String ==> welcome, to, javabeat.net, let's, learn, java
    
```

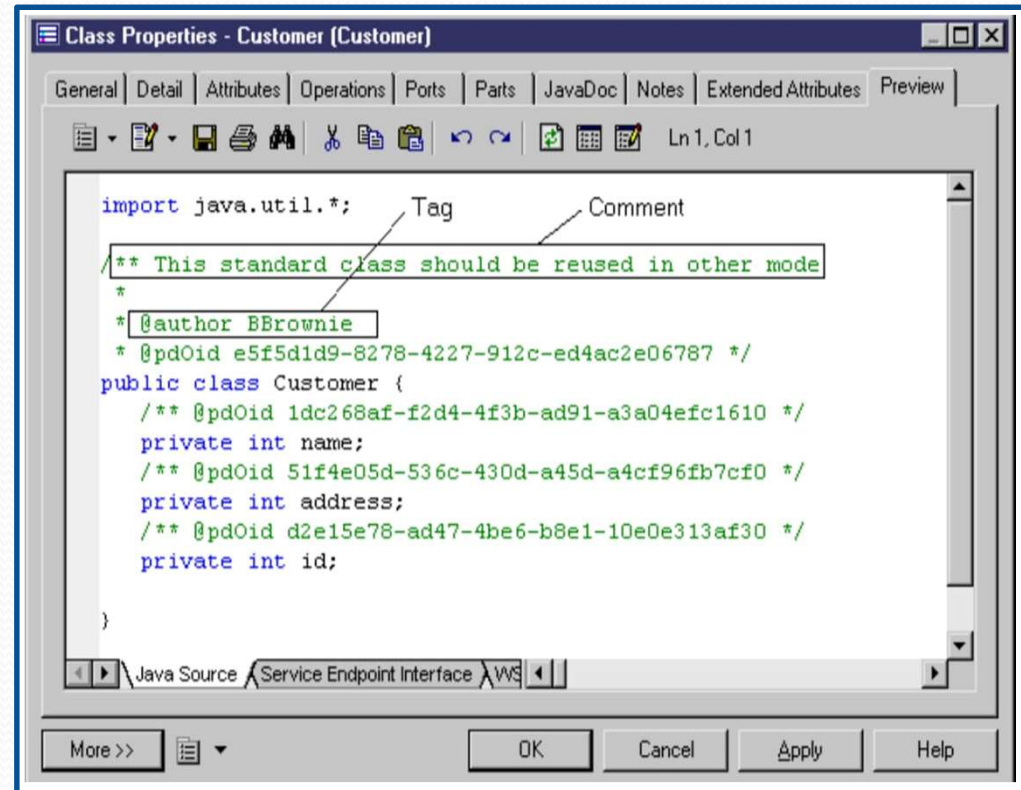
Output

# Documentation Comment

- Starts with `/**` and ends with `*/`.
- Used to generate documentation.
- Processed using Java doc tool.
- Used in professional Java projects

## Syntax:

```
/**
 *This program prints Hello
World
 */
```

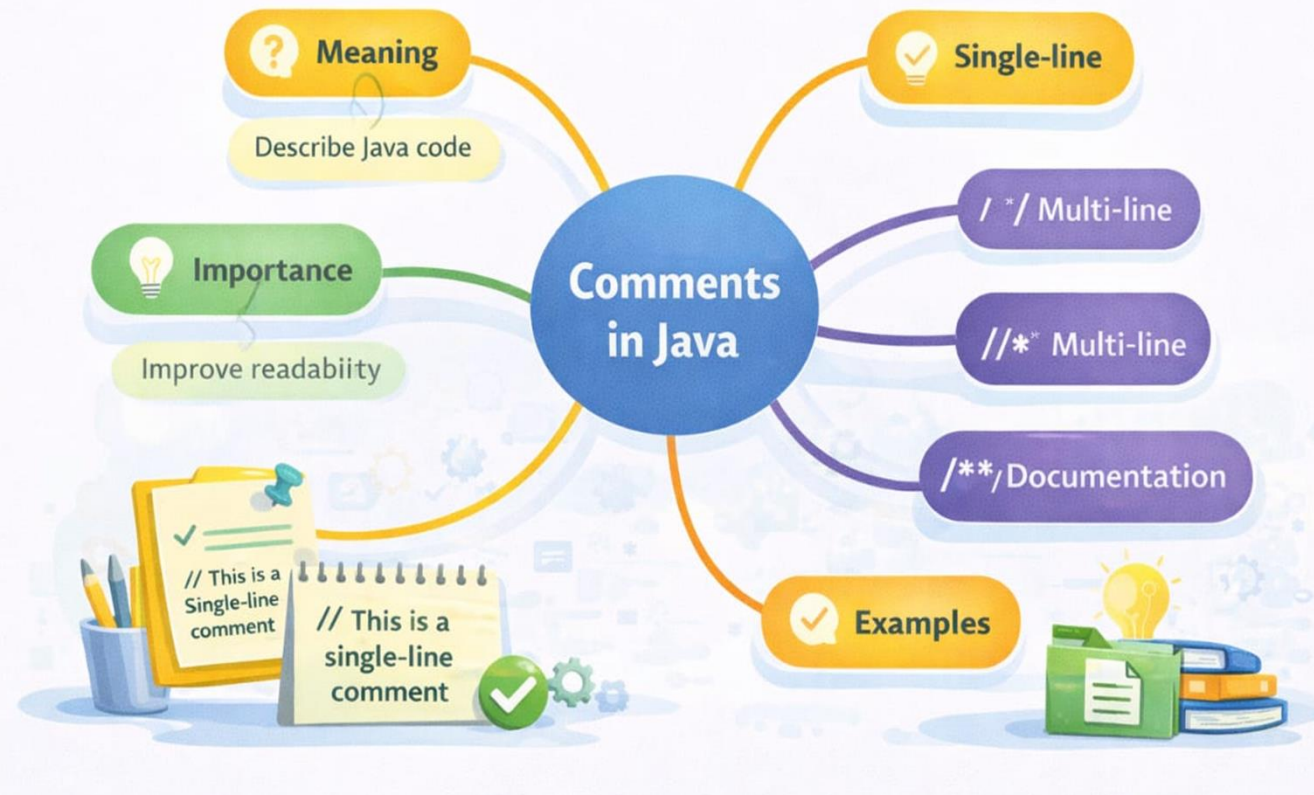


## Comparison of Comment Types

- Single-line → Short and simple explanation.
- Multi-line → Long description.
- Documentation → Official documentation.

All comments are ignored by the compiler

## Comments in Java Mind Map



## t

**1. Comments are \_\_ by the compiler.**

Answer: ignored

**2. Choose the correct answer.**

**Which comment is used for documentation?**

- a) //
- b) /\* /
- c) /\* \*/
- d) ##

**3. What is a comment in Java?**

Answer : A comment is a note written in a program to explain the code and is ignored by the compiler.

# References

- <https://docs.oracle.com/javase>
- <https://www.javatpoint.com/java-comments>
- <https://www.geeksforgeeks.org/comments-in-java>

# Thank You

