JAVA LEVEL-1 NOTES — COMPLETE STUDY MATERIAL

1. What is Java? (Detailed Explanation)

Java is a high-level, object-oriented programming language designed to run on any device using the JVM (Java Virtual Machine). It is widely used in mobile apps, enterprise systems, backend servers, and desktop software. Java is popular because: • It is platform independent (Write Once, Run Anywhere) • It is secure and stable • It supports OOP (Object-Oriented Programming) • It provides automatic memory management • It has a huge community and libraries

2. How Java Works Internally

Java follows a simple execution flow: 1. You write code in a .java file 2. The compiler converts it into .class bytecode 3. JVM executes the bytecode Because JVM exists on Windows, Mac, Android, Linux — the same program runs everywhere.

3. Basic Structure of a Java Program

Every Java program must contain a class and main() method:

```
public class Hello { public static void main(String[] args) {
System.out.println("Hello, Java!"); } }
```

Explanation:

• class Hello → Defines a Java class • main() → Entry point, program starts here • System.out.println → Prints output to screen

4. Variables & Data Types (Deep Understanding)

Variables store data. Each variable has a type that decides what values it can hold. Primitive Data Types: • int → whole numbers • double → decimal numbers • char → single characters • boolean → true/false Reference Types: • String • Arrays • Objects

Example:

```
int age = 20; double height = 5.9; char grade = 'A'; boolean passed = true; String name = "Niket";
```

5. Operators in Java

Java provides operators to perform logical and mathematical operations.

Arithmetic: + - * / % Comparison: == != > < >= <= Logical: && || ! Assignment: = += -= Example:

```
int a = 10, b = 5; System.out.println(a + b); // 15 System.out.println(a > b); // true
```

6. Taking Input in Java (Scanner)

Scanner is used to accept user input from keyboard.

```
import java.util.Scanner; Scanner sc = new Scanner(System.in); System.out.print("Enter your age: "); int age = sc.nextInt(); System.out.println("Age: " + age); sc.close();
```

Common Mistakes:

• Forgetting semicolon • Using == instead of .equals() for strings • Filename not matching class name

7. Essential Programs for Level-1

• Print name • Add two numbers • Swap two numbers • Area of circle • Simple interest • Even/odd checker • Greatest of two numbers • Positive/negative checker

8. Practice Questions

Q1: Write a program to calculate the square of a number Q2: Take user's name and greet them Q3: Find perimeter of rectangle Q4: Convert temperature from Celsius to Fahrenheit Q5: Take 3 numbers and print average

9. Summary — What You Should Know Before Level 2

✔ Java structure ✔ Data types ✔ Variables ✔ Operators ✔ Input/Output ✔ Basic logic You are now ready for Level 2 (IF-ELSE).