📘 Java – Notes - 01

Topic: Data Types, Basic Programs & OOP Introduction

Introduction to OOP Concepts (From Class Notes on Page Image)

The board notes mention:

🔹 Encapsulation

- Wrapping data and methods into a single unit
- Achieved using **class**

🔹 Polymorphism

- An object behaving differently in different situations.
- Same method, different behavior

🔹 Inheritance

- Acquiring properties & behavior of Existing another class

🔹 Abstraction

- Hiding implementation details
- Showing only essential features

Class and Object Example

Student Class

```java
public class Student {
    int rollno;
    String name;
    long contact;
}
```

Object Creation

```java
public class StudentDemo {
    public static void main(String[] args) {
        Student s = new Student();
    }
}
```

Explanation

- `Student` → Class
- `s` → Object
- `new` → Keyword to create object
- Object stores data of class variables

Data Types in Java

Definition

Data types define the **type of data** a variable can store in Java.

Java data types are divided into:

1. **Primitive Data Types**
2. **Non-Primitive (Reference) Data Types**

Primitive Data Types (8 Types)

Primitive types store simple values and are predefined by Java.

Data Type	Size	Description	Example
`byte`	1 byte	Small integer	`byte a = 10;`
`short`	2 bytes	Small integer	`short s = 200;`
`int`	4 bytes	Most commonly used integer	`int x = 5000;`
`long`	8 bytes	Large integer	`long l = 100000L;`
`float`	4 bytes	Decimal number	`float f = 5.5f;`
`double`	8 bytes	Large decimal number	`double d = 10.99;`
`char`	2 bytes	Single character	`char c = 'A';`
`boolean`	1 bit*	True/False	`boolean b = true;`

Important Points

- `int` and `double` are most commonly used in real applications.
- `float` values require **f** at the end (e.g., `5.5f`).
- `long` values require **L** at the end (e.g., `100000L`).
- `char` uses **single quotes (' ')**.
- `String` is **non-primitive (reference type)**.

Example Program – Data Types

```
public class DataTypeExample {
    public static void main(String[] args) {
        int age = 20;
        double salary = 25000.50;
        char grade = 'A';
        boolean isPassed = true;

        System.out.println(age);
        System.out.println(salary);
        System.out.println(grade);
        System.out.println(isPassed);
    }
}
```

Basic Java Program – Hello World

```
public class HelloWorld {
    public static void main(String arg[]) {
        System.out.println("Hello World");
    }
}
```

Structure Explanation

- `public` → Access modifier
- `class` → Defines class
- `main()` → Entry point of program
- `System.out.println()` → Prints output

Key Observations from Board Notes (Page Image)

- Everything in Java is written inside **class**
- Java Installation vs VS Code / Eclipse
- Mention of tools like:
    - IntelliJ
    - STS (Spring Tool Suite)
- Emphasis on:
    - Variables
    - Memory concept
    - Compilation & execution