Extra Class: Notes

📓 Java Notes — Data Types, Variables & Operators (Complete Syntax + Examples)

Beginner-friendly + interview-ready notes.

---

1. Data Types in Java

Java data types are divided into:

1. Primitive Data Types
2. Non-Primitive Data Types

A. Primitive Data Types

Type	Size	Description	Example
byte	1 byte	Small integer	100
short	2 bytes	Short integer	200
int	4 bytes	Integer	1000
long	8 bytes	Large integer	10000L
float	4 bytes	Decimal number	3.14f
double	8 bytes	Large decimal	3.14159
char	2 bytes	Character	'A'
boolean	1 bit	True/False	true

B. Non-Primitive Data Types

These store references to objects.

Examples:

- String
- Array
- Class
- Object
- Interface

Example:

```
String name = "Abhishek";
```

2. Variables in Java

A variable stores data in memory.

A. Variable Declaration Syntax

```
datatype variableName;
```

Example:

```
int age;
float marks;
char grade;
```

B. Variable Initialization

```
datatype variableName = value;
```

Example:

```
int age = 20;
float marks = 85.5f;
char grade = 'A';
```

Note: Float requires **f** at the end.

C. Multiple Variables

```
int a = 5, b = 10, c = 15;
```

D. Types of Variables in Java

1. Local Variable

Declared inside method.

```
void show() {
    int x = 10;
}
```

2. Instance Variable

Declared inside class but outside method.

3. Static Variable

Shared among objects.

```
static int count = 0;
```

E. Rules for Naming Variables

1. Cannot start with number.
2. No spaces.
3. Cannot use keywords.
4. Use meaningful names.
5. Case-sensitive.

Correct:

```
studentAge
totalMarks
```

Wrong:

```
2age
total marks
```

F. Example Program

```
class Main {
    public static void main(String[] args) {

        int age = 20;
        float marks = 85.5f;
        char grade = 'A';

        System.out.println(age);
        System.out.println(marks);
        System.out.println(grade);
    }
}
```

---

3. Operators in Java

Operators perform operations.

A. Arithmetic Operators

Operator	Meaning
+	Addition
-	Subtraction
*	Multiplication
/	Division
%	Modulus

Example:

```
int a = 10, b = 3;
System.out.println(a + b);
```

B. Relational Operators

Operator	Meaning
==	Equal
!=	Not equal
>	Greater
<	Less
>=	Greater equal
<=	Less equal

Example:

```
System.out.println(5 > 3);
```

C. Logical Operators

Operator	Meaning
&&	AND
!	NOT

Example:

```
int age = 20;

if(age > 18 && age < 30)
    System.out.println("Valid Age");
```

D. Assignment Operators

Operator	Meaning
=	Assign
+=	Add & assign
-=	Subtract & assign
*=	Multiply assign
/=	Divide assign

Example:

```
int x = 5;
x += 3;
```

E. Increment / Decrement Operators

Operator	Meaning
++	Increase by 1
--	Decrease by 1

Example:

```
int a = 5;
a++;
```

Pre vs Post Increment

```
int a = 5;
System.out.println(++a); // 6

int b = 5;
System.out.println(b++); // 5
```

F. Ternary Operator

Shortcut for if-else.

Syntax:

```
condition ? value1 : value2;
```

Example:

```
int a = 10, b = 20;
int max = (a > b) ? a : b;
```

G. sizeof Equivalent in Java

Java doesn't have `sizeof`, but:

```
Integer.BYTES
```

Example:

```
System.out.println(Integer.BYTES);
```

Complete Example Program

```
class Main {
    public static void main(String[] args) {

        int a = 10, b = 3;

        System.out.println("Addition: " + (a + b));
        System.out.println("Subtraction: " + (a - b));
        System.out.println("Multiplication: " + (a * b));
        System.out.println("Division: " + (a / b));
        System.out.println("Modulus: " + (a % b));

        a++;
        System.out.println("After increment: " + a);
    }
}
```