Java Data Types - Detailed Notes

1. Introduction

Java is a strictly-typed language, which means every variable must be declared with a data type. Data types specify the size and type of values that can be stored in variables.

Java data types are broadly classified into:

- **Primitive Data Types**
- **Non-Primitive (Reference/Object) Data Types**

---

2. Primitive Data Types

- Java has 8 built-in primitive data types:

2.1 Integer Types

These are used to store whole numbers.

- **byte**: The smallest integer type, useful for saving memory.
- **short**: Larger than byte, but still saves memory compared to int when memory is a constraint.
- **int**: Default integer type, most commonly used for integer values.
- **long**: Used when a wider range than int is needed, often denoted with an 'L' at the end.

Data Type	Size	Default Value	Range
byte	1 byte	0	-128 to 127
short	2 bytes	0	-32,768 to 32,767
int	4 bytes	0	-2^{31} to $2^{31}-1$ (~ -2.1B to +2.1B)
long	8 bytes	0L	-2^{63} to $2^{63}-1$

2.2 Floating Point Types

Used to store decimal values.

- **float**: Single-precision 32-bit floating point. Can hold up to 7 values after decimal point.
- **double**: Double-precision 64-bit floating point. Can hold up to 7 values after decimal point. Default choice for decimal values.

Data Type	Size	Default Value	Range
float	4 bytes	0.0f	Approximately ±3.4e−038 to ±3.4e+038
double	8 bytes	0.0d	Approximately ±1.7e−308 to ±1.7e+308

2.3 Character Type

- **char**: Used to store a single 16-bit Unicode character. Useful for storing characters like 'A', '9', or special symbols.

Data Type	Size	Default Value	Range
char	2 bytes	'\u0000'	0 to 65,535 (Unicode characters)

2.4 Boolean Type

- **boolean**: Only two possible values: true and false. Used for simple flags that track true/false conditions.

Data Type	Size	Default Value	Values
boolean	~1 bit	false	true or false

---

3. Non-Primitive Data Types

Non-primitive types are created by the programmer and are also known as **reference types**. These types hold references to objects.

Examples:

- **Strings**: Sequence of characters.
- **Arrays**: Collection of similar types of elements.
- **Classes**: Blueprint for creating objects.
- **Interfaces**: A contract that classes implement.

Example:

```
String name = "John";
int[] marks = {90, 85, 78};
MyClass obj = new MyClass();
```

4. Summary Table

Type	Category	Size	Default Value	Example Values
byte	Integer	1 byte	0	-100, 0, 100
short	Integer	2 bytes	0	-30000, 0, 30000
int	Integer	4 bytes	0	-1,000,000 to +1,000,000
long	Integer	8 bytes	0L	9999999999L
float	Floating	4 bytes	0.0f	3.14f, -0.5f
double	Floating	8 bytes	0.0d	3.14159265359
char	Character	2 bytes	'\u0000'	'A', '9', '#'
boolean	Logical	1 bit	false	true, false

5. Conclusion

Understanding Java data types is fundamental to efficient and safe programming. Choosing the correct data type ensures better performance and memory management.