



Java™

सी डैक  
CDAC  
पटना | PATNA

# Java Programming Fundamentals

Java is composed of several core elements that together make up its syntax and structure.

1. Keywords
2. Identifiers
3. Operators
4. Literals
5. Punctuators (Separators)
6. Comments

In Java, there are two important language elements:

1. Keyword

and

2. Identifiers

C-DAC Patna

- ❑ Keywords are special/reserved words that are already defined by Java.
- ❑ They have predefined meanings and are used to perform specific tasks.
- ❑ You cannot use keywords as names for your variables, classes, or methods.\

Examples of Java Keywords:

int – used to declare an integer variable

class – used to define a class

public, static, void, if, else, return, for, while

- ❑ Java has 50+ reserved keywords and they are all written in lowercase.

- ❑ Identifiers are names that you (the programmer) give to variables, methods, interfaces, enums, classes and objects, etc.
- ❑ Identifiers help the computer identify and work with data.

Examples of Identifiers:

Example: `studentName`, `marks`, `myMethod`, `Car`, `TotalAmount`.

## Rules for Writing Identifiers:

1. Must begin with a letter (A–Z or a–z), underscore (`_`), or dollar sign (`$`).
2. Cannot start with a number (e.g., `1name` ).
3. Cannot use keywords as identifiers (e.g., `int` as variable name ).
4. Can include letters and digits after the first character.
5. Case-sensitive: `Score` and `score` are different.
6. No restriction on the length

Choose right identifier ?

1. 2name
2. @student
3. student\_name
4. class
5. void
6. total\$
7. double
8. 123abc

C-DAC Patna

## What is a Variable?

In Java, a variable is a named memory location used to store data. It acts as a container for values that can be accessed and modified during the execution of a program. Naming convention is camelCase.

ex. `int x = 10;`

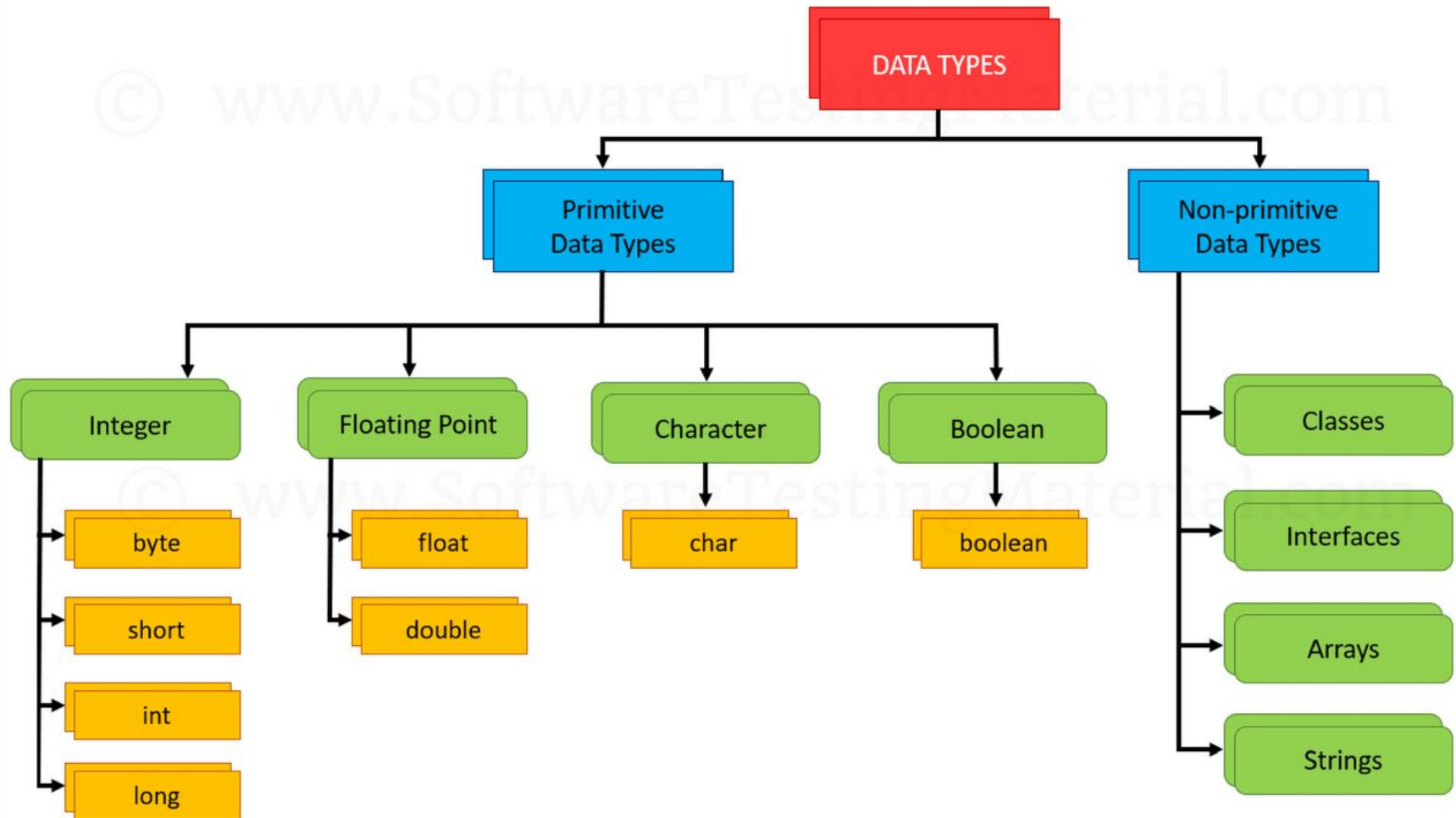
## What is Data Type?

Every variable must be declared with a specific data type (e.g., `int`, `String`, `double`, `boolean`). This data type determines the kind of data the variable can hold and the amount of memory allocated for it.

**There is broadly two category of data types:**

1. Primitive data types
2. non-primitive/reference data types

# Primitive data types



Source: Internet

# Data Types

Boys Kill Monsters Go To Play Every Zoo Yesterday

Datatype	Default Value	Default size	Range (Approx)
boolean	false	1 bit (depends on JVM)	true or false
char	'\u0000'	2 byte	Unicode characters
byte	0	1 byte	-128 to 127
short	0	2 byte	-32,768 to 32,767
int	0	4 byte	-2,147,483,648 to 2,147,483,647
long	0L	8 byte	Big numbers with <b>suffix L</b>
float	0.0f	4 byte	~7 decimal digits <b>(with f)</b>
double	0.0d	8 byte	~15 decimal digits

A literal is a fixed value that is directly written in the Java code — it represents the actual data you want to assign to a variable.

In simple words: A literal is the value itself, not the variable.

```
int age = 20;
```

age → variable

int → data type

20 → literal (this is the actual value stored in the variable)

# Variable

---

`int a,b; // Declaration`

`a = 10, b = 20; // Initialization`

`int c = a + b; // Combined declaration and assignment`

Now Work with different data types.

C-DAC Patna

# Local variable type inference

---

```
var number=11;
```

```
var number;
```

Once a variable declared with var has been initialized, you cannot even change the type of the value assigned to the variable.

var is not a data type — it only tells the compiler to infer the type at compile time.

```
var number=11; //Line 1  
number="Java"; //Line 2
```

```
var number=null;
```

Java is a strongly and statically typed language, where most type checking is done at compile time, with additional runtime checks.

There are 2 types of type conversions:

1. Widening conversion / implicit type casting
2. Narrowing conversion / explicit type casting

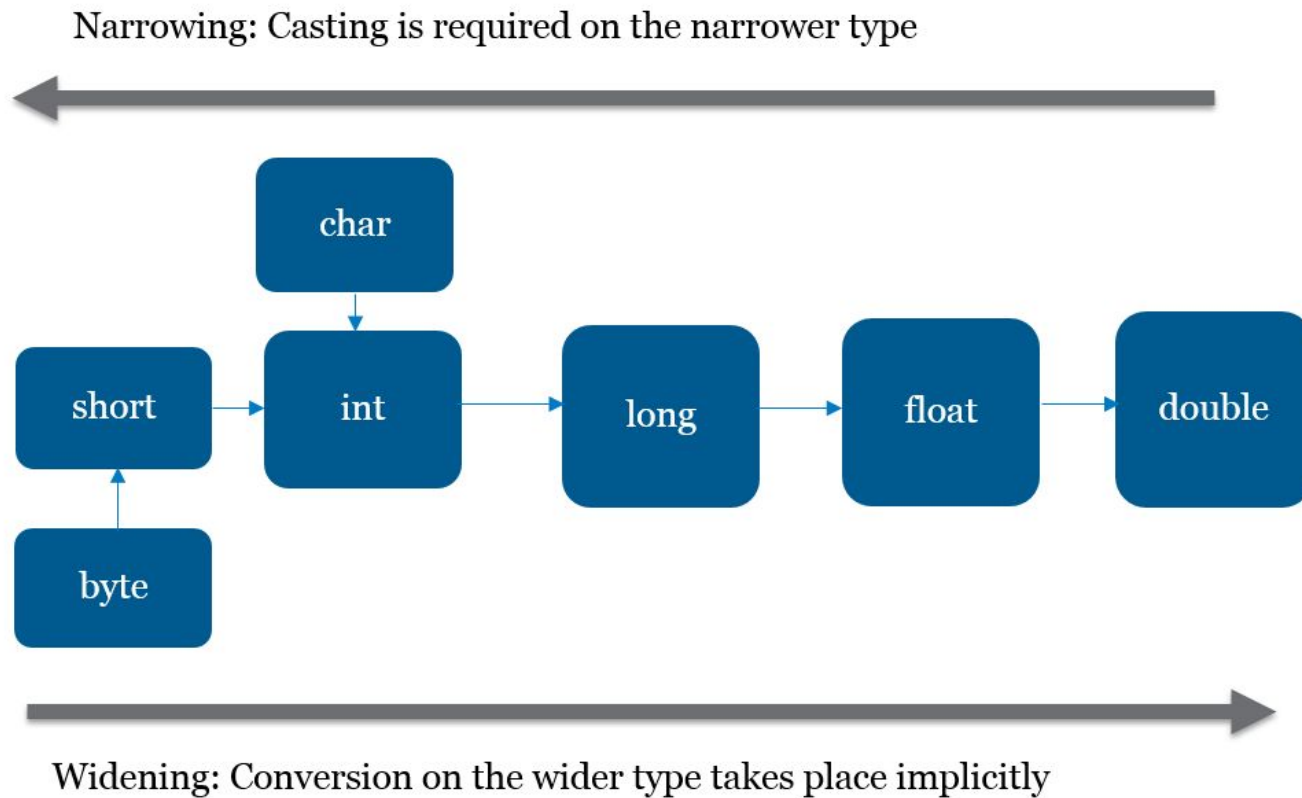
## **Widening conversion / implicit type casting :**

Converting a variable of smaller datatype to the larger datatype, without data loss is known as widening conversion. Automatically done by Java.

## **Narrowing conversion / explicit type casting:**

Converting a variable of larger datatype to the smaller datatype, with some data loss is known as narrowing conversion. Must be manually done by the programmer.

# Type Casting & Conversion



Source: Internet

# Type Casting & Conversion

---

```
int myInt = 9;  
double myDouble = myInt; // Automatic casting: int to double  
System.out.println(myInt);    // 9  
System.out.println(myDouble); // 9.0
```

## AND

```
double myDouble = 9.78;  
int myInt = (int) myDouble; // Manual casting: double to int  
  
System.out.println(myDouble); // 9.78  
System.out.println(myInt);    // 9
```

A constant is a variable whose value cannot be changed after it is initialized. Constant names are written in UPPER\_CASE

In Java, we use the final keyword to declare constants.

```
final double PI = 3.14159;
```

```
PI = 3.14; // Error: Cannot assign a value to final variable
```

C-DAC Patna

- ❑ Comments are lines in your code that Java ignores when running the program.
- ❑ Used to explain what your code is doing.
- ❑ Helps you and others understand the code better.
- ❑ Java compiler does not execute comments.

## Types of Comments in Java:

### 1. Single-Line Comment

```
// This is a single-line comment  
int age = 25; // declaring age
```

### 2. Multi-Line Comment.

```
/*  
This is a multi-line comment.  
You can write explanations here  
that span several lines.  
*/  
int marks = 90;
```

# Java Naming Conventions

Element	Convention	Example
Variable	camelCase	studentName, totalMarks
Constant	ALL_UPPERCASE with —	MAX_SIZE, PI_VALUE

C-DAC Patna

# Programming Indentation Techniques

```
public class Test{  
    public static void main(String[] args){  
        int a=5;  
        if(a>0){  
            System.out.println("Positive");  
        }else{  
            System.out.println("Negative");  
        }  
    }  
}
```

**OR**

```
public class Test {  
    public static void main(String[] args) {  
        int a = 5;  
        if (a > 0) {  
            System.out.println("Positive");  
        } else {  
            System.out.println("Negative");  
        }  
    }  
}
```

C-DAC Patna

- ❑ Use 4 spaces or 1 tab per level of indentation.
- ❑ Every block ({} ) should have its own indentation level.
- ❑ Align closing braces } with the line that opened the block.
- ❑ Keep nested blocks clearly structured.

# Output using System.out

---

```
System.out.println("Hello, World!");
```

C-DAC Patna

## Input Using Scanner Class

1. Import the Scanner class
2. Create Scanner object
3. Use methods like `.nextInt()`, `.nextLine()`, `.nextDouble()`, etc.

```
import java.util.Scanner;
public class InputExample {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in); // Step 1: Create Scanner

        System.out.print("Enter your name: ");
        String name = sc.nextLine();          // Step 2: Read string input

        System.out.print("Enter your age: ");
        int age = sc.nextInt();                // Step 3: Read integer input

        System.out.println("Hello " + name + ", you are " + age + " years old.");
        sc.close();
    }
}
```

1. Write a Java program to declare and initialize all 8 primitive data types and print their values.
  2. Write a Java program to create a final variable(Constant) for the value of PI, assign it 3.14.
  3. Write a program using **var** (Inference) to declare variables of types: int, double, char, and boolean. Print their values.
  4. Write a Java program to demonstrate implicit type conversion (widening) in the following two cases:
    - i. From int to double
    - ii. From long to float
- a. Your program should:
- iii. Declare and initialize an int and a float variable.
  - iv. Assign them to double variables without casting.
  - v. Print both original and converted values.

# Optional Assignments

---

5. Add comments to the Q5 code using all two types of Java comments:
  - a. Single-line comments to explain each line of code.
  - b. A multi-line comment at the top to describe the overall purpose of the program.
  
6. Write a program that declare char variable, Type cast it to int, and prints that int variable.
  
7. Write a Java program that:
  - a. Declares a float variable with a decimal value.
  - b. Type casts it to an int.
  - c. Prints both the original float value and the converted int value.
  - d. Shows the data loss (the decimal part is lost) using print statements.

**THANK YOU!!**

C-DAC Patna