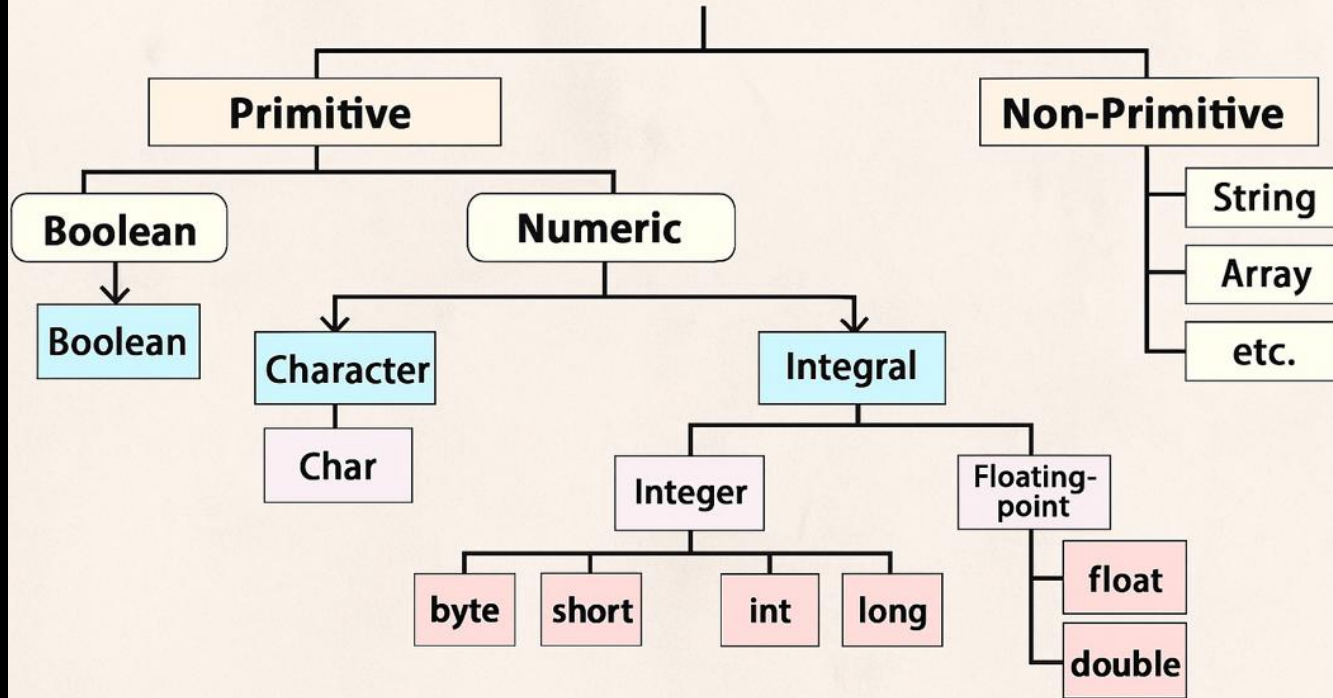


Data type in java

Type	Size	Default Value	Type of Value Stored
byte	1 byte	0	Integral
short	2 byte	0	Integral
int	4 byte	0	Integral
long	8 byte	0L	Integral
char	2 byte	'\u0000'	Character
float	4 byte	0.0f	Decimal
double	8 byte	0.0d	Decimal
boolean	1 bit (till JDK 1.3 it uses 1 byte)	false	True or False

Data Types in Java



◆ camelCase (ऊँट जैसा नाम)

- 📌 **Rule:** Pehla word chhota, baaki words capital letter se start.
- 📌 **Use:** Variables aur methods ke liye Java mein commonly use hota hai.
- ✅ **Examples:**
 - userName
 - totalAmount
 - fetchStudentData()
- 💡 **Samjho:** Jaise ek sentence ko jod ke likhna ho, pehla word chhota baaki bada.

♦ snake_case (साँप जैसा नाम)

- 📌 **Rule:** Sab chhote letters, words ko underscore _ se separate karo.
- 📌 **Use:** Python mein common hai, par Java mein nahi use hota.
- ✅ **Examples:**
 - user_name
 - total_amount
 - student_roll_no

◆ **kebab-case** (हाइफ़न वाला नाम)

- 📌 **Rule:** Sab chhoti letters, beech mein hyphen –
- 📌 **Use:** Mostly URLs ya CSS files mein use hota hai, Java mein nahi.
- ✅ **Examples:**
 - user-name
 - login-status
 - mobile-number

◆ Keep a Good and Short Name (Achha aur Chhota Naam Rakho)

- 📌 **Rule:** Naam meaningful ho, par zyada lamba na ho.
- 📌 **Use:** Har language mein important hota hai. Code samajhne mein asani hoti hai.
- ✅ **Examples:**
 - email, score, isActive ✅ (short + clear)
 - ❌ value1, data123, xyzzy ❌ (confusing)
- 🔄 **Better Examples:**
 - userAge ✅
 - isVerified ✅
 - loginTime ✅



Java Identifier Rules

✓ Rule 1: Allowed Characters

Sirf alphabets (A–Z, a–z), numbers (0–9), \$ aur _ allowed hai.

✓ Valid Examples:

- studentName
- roll_number
- \$_amount

✗ Invalid:

- student-name (hyphen ✗)
- total@marks (special character ✗)

❌ Rule 2: Keywords use nahi kar sakte

Java ke reserved words (like int, class, static) ko identifier ke roop me use nahi kar sakte.

❌ Invalid:

•int = 5; // ❌


✓ Valid:

•intValue = 5; ✓

Rule 3: Digit se Start Nahi Kar Sakta

Identifier number se start nahi hona chahiye.

 Invalid:

•123name 

✓ Valid:

•name123 



Rule 4: Case Sensitive

Java mein name aur Name alag-alag identifiers hote hain.

✓ Example:

```
int name = 10;  
int Name = 20;  
System.out.println(name); // 10  
System.out.println(Name); // 20
```

Rule 5: Length ka Limit Nahi, par Short & Sweet ho

Length ka koi fixed limit nahi hai, par 4–15 characters best hote hain — clear aur easy to read.

✓ **Good:**

- userName,
- isValid,
- totalMarks ✓

✗ **Too Short/Too Long:**

- a ✗ (too short, not meaningful)
- totalMarksOfSemester1ExamHeldInApril2024 ✗ (too long)

Rule No.	Description	Example (✓/✗)
1	Only A-Z, a-z, 0-9, \$, _	roll_1, \$_name ✓
2	No Java keywords	int, class ✗
3	Cannot start with digit	123abc ✗, abc123 ✓
4	Case-sensitive	value ≠ Value
5	No length limit, use optimal size	totalMarks, age ✓



Literal in Java

Literal matlab ek **fixed constant value** jo directly code mein likhi jaati hai. Jab hum kisi variable ko koi value assign karte hain, toh jo value hum likhte hain wo **literal** hoti hai.



Real-Life Analogy:

Jaise tum kisi dost ko bolte ho: "Mujhe 5 samose chahiye"

Yahaan "5" ek **literal** hai — fixed value jise tumne directly bola.



Example samajho:

```
int age = 25;
```

- Yahaan age ek variable hai.
- Aur 25 ek **integer literal** hai (kyunki ye fixed number hai jo directly diya gaya hai).

Types of Literals in Java with Examples:

Type	Example	Explanation
Integer	10, -5	Whole numbers, bina decimal
Floating Point	3.14, -0.5f	Decimal values like float/double
Character	'A', '9'	Single character in single quote
String	"Hello", "Jayesh"	Text enclosed in double quotes
Boolean	true, false	Sirf do hi values hote hain: true/false
Null	null	Object ya reference ke liye khali value

```
int number = 100;      // 100 is an integer literal
float pi = 3.14f;      // 3.14f is a float literal
char grade = 'A';      // 'A' is a char literal
String name = "Jayesh"; // "Jayesh" is a string literal
boolean isPassed = true; // true is a boolean literal
```




Keywords in Java

◆ What are Keywords?

Keywords Java ke predefined words hote hain jinka meaning Java compiler ke liye already fix hota hai.

Inhe **hum variable, method, class name** ke roop mein **use nahi kar sakte**.



Java ke 53 Keywords (Latest with Java 17+)

◆ 1. Control Flow & Looping Keywords

Keyword	Meaning (Hinglish)
if	Condition check karne ke liye
else	Jab if false ho jaye
switch	Multiple condition check
case	Switch block ke andar alag-alag condition
default	Koi case match na ho to ye chalega
for	Loop run karne ke liye
while	Jab tak condition true ho
do	Kam se kam ek baar chalega
break	Loop se bahar nikalne ke liye
continue	Current iteration skip karne ke liye
return	Value return karne ke liye
yield	Switch expression me result return karne ke liye

◆ 2. Access Modifiers & OOP Keywords

Keyword	Meaning (Hinglish)
class	Class banane ke liye
interface	Interface banane ke liye
enum	Enum (constant group) banane ke liye
extends	Inheritance ke liye
implements	Interface implement karne ke liye
new	Naya object banane ke liye
this	Current object ka reference
super	Parent class ka reference
abstract	Incomplete method/class
final	Fixed/constant banata hai
static	Class-level member
package	Classes ka group
import	Dusre package ka class use karne ke liye

◆ 3. Exception Handling Keywords

Keyword	Meaning (Hinglish)
try	Risky code block
catch	Error ko handle karne ke liye
finally	Hamesha chalne wala block
throw	Manually exception dene ke liye
throws	Method ke signature me exception declare karne ke liye

◆ 4. Primitive Data Types

Keyword	Meaning (Hinglish)
int	Integer number
float	Decimal value (small)
double	Decimal value (big)
char	Single character
byte	1-byte value
short	2-byte value
long	Large integer value
boolean	true/false ke liye

◆ 5. Other Useful Keywords

Keyword	Meaning (Hinglish)
public	Har jagah se access allowed
private	Sirf usi class ke andar
protected	Package + subclass ke liye
void	Method kuch return nahi karega
volatile	Multithreading me variable memory sync karne ke liye
transient	Variable ko serialization se exclude karne ke liye
instanceof	Object kis class ka hai, check karne ke liye
assert	Testing ke time condition check karne ke liye
synchronized	Thread safety ke liye
native	Java method linked with native (C/C++) code
strictfp	Floating point calculation consistent banata hai

◆ 6. New & Reserved Keywords in Java 17+

Keyword	Use / Reserved For Future
record	Immutable data class banane ke liye
sealed	Restrict which classes can extend this class
permits	Specifies which classes can extend sealed class
non-sealed	Sealed class ka open extension allow karna
const	Reserved (use nahi hota abhi)
goto	Reserved (Java me use nahi hota)



Total Keywords Count: 53

- 💡 44 = Regular use
- 🔒 9 = New/Reserved keywords



Bonus Tip:

Java keywords are **always lowercase**
IDE mein inka color **highlighted** hota hai.



Java Escape Sequences with Example

Escape Sequence	Description (Hinglish)	Example Code	Output / Effect
\t	Ek tab space insert karta hai	System.out.println("A\tB");	A B (5-6 space ka gap)
\b	Ek backspace insert karta hai (previous char delete)	System.out.println("ABC\bD");	ABD (C hata diya gaya)
\n	Nayi line pe le jata hai	System.out.println("Hello\nWorld");	Hello World
\'	Ek single quote ' print karta hai	System.out.println("\'");	'
\"	Ek double quote " print karta hai	System.out.println("\"Java\"");	"Java"
\\	Ek backslash \ print karta hai	System.out.println("C:\\Path");	C:\Path

jayesh_kande_ ▾ ●

What's
on your
playlist?



Jayesh Kande

16
posts

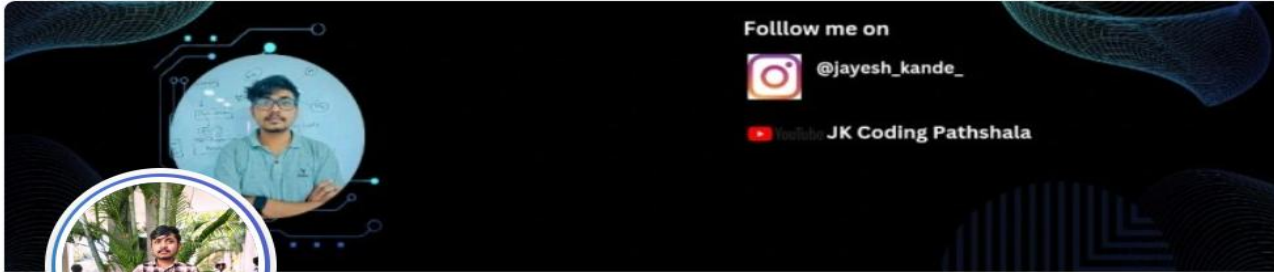
275
followers

276
following

23

रास्ते बदलो, मंजिल नहीं

yt.openinapp.co/0y0qd



Jayesh Kande

Third-Year IT Engineering Student | Aspiring Web Developer
| Java Enthusiast | Data Structures & Algorithms Learner |
Proficient in C, C++, Java, and MERN Stack | AI + Web
Development Project Enthusiast

Nashik, Maharashtra, India · [Contact Info](#)

494 followers · 495 connections



[See your mutual connections](#)

[Join to view profile](#)

[Message](#)



Kbt engineering college nashik



Thank You for Watching!



Follow us on Instagram: **@jayesh_kande_**



Connect with us on LinkedIn: **[Jayesh Kande]**