

Understanding Variables, Naming Conventions, and Examples

Variable?

A variable is a storage container for storing the data.

- A **variable** is like a **jar** in your kitchen.
- You **write a name** on the jar (like *Sugar*).
- You **put something inside** (like *2 kg sugar*).
- Later, you can **change it** (remove sugar and put *rice*).



Example:

```
int sugar = 2
```

int = in there represent type of jar (it can hold only sugar/rice/pulse).

sugar = label of the jar.

2 = thing stored inside the jar (now it sugar).

It has three parts:

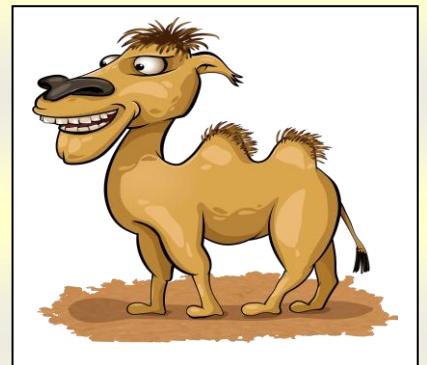
1. Type → kind of data (int, float, String)
2. Name → label of the box
3. Value → actual data inside the box

When writing variable names in Java, follow these rules:

- Must start with a letter or **I** or **\$** (not with a number).
- Can contain **letters, digits, _, \$.**
- Case-sensitive → `age` and `Age` are different variable.
- **No spaces allowed.**
- Use **camelCase** for naming.
- Reserved words (like Java keywords, such as `int` or `boolean`) cannot be used as variable.
- Variable names should be descriptive.

CamelCase

- camelCase is a style of writing names (like variable names) in programming.
- The first word starts with a small letter.
- Every next word starts with a Capital letter.
- Looks like a “camel’s hump”  → that’s why it’s called camelCase.



Good Variable Names

- studentName → clear, uses camelCase
- totalMarks → meaningful name
- isLoggedIn → boolean variable (starts with *is*)
- averageScore → self-explanatory
- maxHeight → describes what it stores
- bankBalance → meaningful and readable
- userEmail → clear purpose
- numberOfBooks → descriptive
- priceAfterDiscount → long but meaningful
- cityName → tells what data it stores

✖ Bad Variable Names (Break Rules or Hard to Read)

- 123name → ✖ cannot start with numbers
- total marks → ✖ spaces not allowed
- StudentName → ✖ PascalCase (not for variables)
- stuNAME → ✖ mixed random capitalization
- name@user → ✖ special characters not allowed (except _ or \$)
- Name → ✖ should start with lowercase for variables
- abc123 → ✖ meaningless, hard to understand
- variable → ✖ too generic, doesn't explain purpose

Syntax (Imp)

Primitive data type

byte, short, int, long, float, double, char, Boolean, String

Declaration of variable

<data type> <variableName>;

```
int totalMarks ,numberOfBooks ;  
totalMarks = 450;  
numberOfBooks = 12;
```

Declaration + initialization

<data type> <variableName> = <value> ;

```
int totalMarks = 450;  
int numberOfBooks = 12;  
or
```

```
int totalMarks = 450, numberOfBooks = 12;
```

→ To declare more than one variable of the same type, you can use a comma-separated list:

Example of variable

```
String studentName = "Amit"  
String userEmail = "amit@example.com";  
String cityName = "Patna";  
  
int totalMarks = 450;  
int numberOfBooks = 12;  
  
boolean isLoggedIn = true;  
  
double averageScore = 89.5;  
float priceAfterDiscount = 299.99f;  
  
long bankBalance = 1500000L;  
  
short maxHeight = 180;
```

How to print variable in java?

System.out.println(`studentName`);

System.out.println(`userEmail`);

System.out.println(`cityName`);

System.out.println(`totalMarks`);

System.out.println(`numberOfBooks`);

Output

Amit

amit@example.com

Patna

450

12

How to print variable in java with description/message?

```
System.out.println("Student Name: " + studentName);
```

```
System.out.println("Total Marks: " + totalMarks);
```

```
System.out.println("Is Logged In: " + isLoggedIn);
```

```
System.out.println("Average Score: " + averageScore);
```

```
System.out.println("Max Height: " + maxHeight);
```

Output

Student Name: Amit

Total Marks: 450

Is Logged In: true

Average Score: 89.5

Max Height: 180

Hands on
Thankyou!!