Types of variables in Java

- **Variables** : A variable provide us with named storage that our programs can manipulate. Each variable in Java has a specific type, which determines the size and layout of the variable's memory.

Syntax for declaring variable:

```
data_type variable = value;
```

Exm:→ `int a = 10;`

`double Pi = 3.14159;`

`Char a = 'a';`

- **Constant** : During the execution of a program, value of variable may change. A constant represents permanent data that never changes.

Syntax for declaring constant:

```
Static final datatype ConstantName = Value;
```

Exm:- `Static final float Pi = 3.14159;`

⊛ 3 kinds of variables in Java

- **Local variable**

(i) A variable that is declared inside the

method is called local variable.

(ii) Local variables are created when the method, Constructor or block is entered and the variable will be destroyed once it exits the method, Constructor or block.

(iii) Access modifiers cannot be used for local variables.

(iv) There is no default value for local variables, so local variables should be declared and an initial value should be assigned before the first use.

- Instance variable

(i) A variable that is declared inside the class but outside the method is called instance variable. It is not declared as static.

(ii) Instance variables are created when an object is created with the use of the keyword 'new' and destroyed when the object is destroyed.

(iii) Access modifiers can be given for instance variables.

(iv) Instance variables have default values. For numbers the default value is 0, For Booleans it is false and For object references it is null. Values can be assigned during the declaration or within the constructor.

(v) Instance variables (or Non-static variables) can be accessed only through reference variables (or) address variable. It cannot be accessed by using class name.

• Class/static variables

(i) A variable that is declared as static is called static variable. It cannot be local.

(ii) Class variables also known as static variables are declared with the 'static' keyword in a class, but outside a method, constructor or a block.

(iii) Memory is allocated only once for the static variables, irrespective of no. of objects created.

(iv) Static variables can be accessed using reference variable, or using class name.

Exm:→

```
Class A
{
int data = 50; // instance variable
Static int m = 100; // static variable
Void method()
{
int n = 90; // Local variable
}
}
```