

**1.What is statically typed and Dynamically typed Programming Language?**

=>First, dynamically-typed languages perform type checking at runtime, while statically typed languages perform type checking at compile time. This means that scripts written in dynamically-typed languages can compile even if they contain errors that will prevent the script from running properly

Second, statically-typed languages require you to declare the data types of your variables before you use them, while dynamically-typed languages do not.

Consider the two following code examples:

// Java example

int num;

num = 5;

// Python example

num = 5

Both examples do the same thing: create a variable called num and assign it the value 5. The difference lies in the first line of the Java example, int num;, which defines num's data type as int. Java is statically-typed, so it expects its variables to be declared before they can be assigned values. Python is dynamically-typed and determines its variables' data types based on their values.

Dynamically-typed languages are more flexible and can save you time and space when writing scripts.

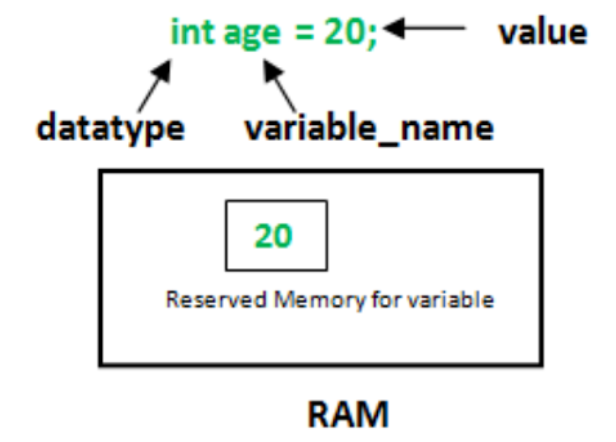
**2.What is the variable in Java?**

=>A variable is the title of a reserved region allocated in memory. In other words, it may be referred to as the name of a memory location.

* It is a container that holds the value while the Java program is executed.
* Each variable should be given a unique name to indicate the storage area.
* A variable is assigned with a data type.

**3.How To Assign a Value To Variable?**

=>Type variable\_name [= value];   
The variable\_name is the name of a variable. We can initialise the variable by specifying an equal sign and a value (initialization i.e. assigning an initial value, is optional). However, the compiler never assigns a default value to an uninitialized local variable in Java.



Here, rate is an int data type variable with the value 40 assigned to it.   
In the example above, the variable can only hold integer values, as indicated by the int data type.

Here, we assigned a value to the variable during the declaration process. However, as stated before, it is optional.

Variables can be declared and assigned separately. Example,

int rate; rate = 40;

**Changing values of variables**

Interestingly, a variable's value can also be changed in the program. Look at the example below :

int rate = 50;

System.out.println(rate);// 50

rate = 60;

System.out.println(rate);// 60

Initially, the value of rate was 50 but it has changed to 60 after the last update, rate=60.

**4. What are Primitive Data types in Java?**

=>A primitive type is predefined by the language and is named by a reserved keyword.

**1.Boolean Type**

The Boolean data type can have two values– true or false and hence are typically used in true/false situations.

For example, Boolean flag=true;

**2. Byte Type**

Values for the byte data type range from -128 to 127 (8-bit signed two's complement integer, you will know more about it once we move to programs and applications)

A byte type is used in place of an int to save memory when it is certain that the value of a variable will be between -128 and 127.

For example, byte range=105;

**3. Short Type**  
The short data type can have values ranging from -32768 to 32767 (16-bit signed two's complement

integer)

If the value of a variable is certain to be between -32768 and 32767, short is used in place of other integer data types (int, long).

For example, short loss=-50;

**4. Int Type**  
Values for the int data type range from 231 to 231-1(32-bit signed two's complement integer, you will know

about it as we move to programs.

In Java SE 8 and later, you can use the int data type to represent an unsigned 32-bit integer, which has a minimum value of 0 and a maximum value of 232-1.

For example,   
int profit=5000;

**5. Long Type**  
Values for the long data type range from - 263 to 263-1 (64-bit signed two's complement integer).

You can use an unsigned 64-bit integer with a minimum value of 0 and a maximum value of 264-1 if you're using Java 8 or later.

For example:   
long profit=455559990;

**6. Double Type**

The double data type is a 64-bit floating-point data type with double precision.

It should never be used for exact values like currency.

For example:   
double height=12.5;

**7. Float Type**  
The float data type is a 32-bit single-precision floating-point value. If you're curious, you can learn more

about single-precision and double-precision floating-point. It should never be used for precise values like money.

For example:   
float depth=-32.3f;

**8. Char Type**  
It's a Unicode (an international character encoding standard that provides a unique number for every

character across languages and scripts) 16-bit characters  
W The char data type has a minimum value of 'u0000' (0) and a maximum value of 'uffff'.

For example: char temp=’a’;

**5.What are the Identifiers in Java?**

=>An identifier is a name given to a package, class, interface, method, or variable. All identifiers must have different names.   
In Java, there are a few points to remember while dealing with identifiers :

* **Rule 1 −** All identifiers should begin with a letter (A to Z or a to z), $ and \_ and must be unique.
* **Rule 2 −** After the first character/letter, identifiers can have any combination of characters.
* **Rule 3 −** A keyword cannot be used as an identifier.
* **Rule 4 −** The identifiers are case-sensitive.
* **Rule 5 -** Whitespaces are not permitted.
* Examples of legal identifiers: rank, $name, \_rate, \_\_2\_mark.
* Examples of illegal identifiers: 102pqr, -name.

These variables, identifiers etc. consume memory units. Before proceeding ahead, let us have a look at the memory unit concept too. Here, we will only focus on the relevant concept of memory.

**6.List the Operators in Java?**

=>Operator in java is a symbol that is used to perform operations. For example: +, -, \*, / etc.

There are many types of operators in Java which are given below:

* Unary Operator,
* Arithmetic Operator,
* Shift Operator,
* Relational Operator,
* Bitwise Operator,
* Logical Operator,
* Ternary Operator and
* Assignment Operator.

**7. Explain about Increment and Decrement operators and give an examples**

**=>Increment operator(++)**is used to increment a value by 1. There are two varieties of increment operator:

* **Post-Increment(a++):**Value is first used for computing the result and then incremented.
* **Pre-Increment(++a):** Value is incremented first and then the result is computed.

**Decrement operator(--)**is used to Decrement a value by 1. There are two varieties of decrement operator:

* **Post-Decrement(a--):**Value is first used for computing the result and then decremented.
* **Pre-Decrement(--a):** Value is decremented first and then the result is computed.

**Example:**

class Main {

public static void main(String[] args) {

int a = 5, int b = 6;

int c = a++;//post increment

int d = ++a;//pre increment

int e = b—-;//post decrement

int f = –b;// pre decrement

System.out.println(c);

System.out.println(d);

System.out.println(e);

System.out.println(f);

}

}