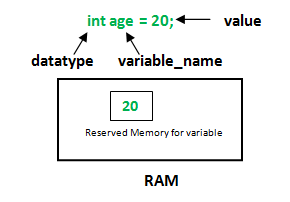
Assignment on Java

1. Variables:

A **variable** is a container which holds the value while the **java** program is executed. A **variable** is assigned with a datatype. **Variable** is a name of memory location.

We can declare variables in java as follows:

[](http://cdncontribute.geeksforgeeks.org/wp-content/uploads/Variables-in-Java.png)

There are three types of **variables in java**: local, instance and static.

1. Local Variable: A variable is declared inside the body of the method is called local variable. The scope of these variables exists only within the block in which the variable is declared. i.e. we can access these variable only within that block.
2. Instance Variable: Instance variables are non-static variables and are declared in a class outside any method, constructor or block. Unlike local variables, we may use access specifiers for instance variables. If we do not specify any access specifier then the default access specifier will be used.
3. Static Variable: Static variables are also known as Class variables. These variables are declared similarly as instance variables, the difference is that static variables are declared using the static keyword within a class outside any method constructor or block.
4. Methods

A method is a collection of statements that perform some specific task and return result to the caller. A method can perform some specific task without returning anything. Methods allow us to **reuse** the code without retyping the code.



1. Static

The **static keyword** in Java is used for memory management mainly. We can apply java static keyword with variables, methods, blocks and nested class. The static keyword belongs to the class than an instance of the class.

The static can be:

1. Variable (also known as a class variable)
2. Method (also known as a class method)
3. Block
4. Nested class
5. Object

An entity that has state and behavior is known as an object e.g. chair, bike, marker, pen, table, car etc. It can be physical or logical (tangible and intangible). The example of an intangible object is the banking system.

**Object Definitions:**

* An object is *a real-world entity*.
* An object is *a runtime entity*.
* The object is *an entity which has state and behavior*.
* The object is *an instance of a class*.

An object has three characteristics:

I) identity

ii) State

* 1. Behavior

There are 3 ways to initialize object in java.

1. By reference variable
2. By method
3. By constructor
4. Class

A class is a group of objects which have common properties. It is a template or blueprint from which objects are created. It is a logical entity. It can't be physical.

A class in Java can contain:

* **Fields**
* **Methods**
* **Constructors**
* **Blocks**
* **Nested class and interface**

1. Constructor

In Java, a constructor is a block of codes similar to the method. It is called when an instance of the object is created, and memory is allocated for the object.

It is a special type of method which is used to initialize the object.

Rules for cresting java constructor:  
There are three rules defined for the constructor.

1. Constructor name must be the same as its class name
2. A Constructor must have no explicit return type
3. A Java constructor cannot be abstract, static, final, and synchronized

There are two types of java constructor:

1. Default constructor: A constructor is called "Default Constructor" when it doesn't have any parameter.
2. Parameterized constructor: A constructor which has a specific number of parameters is called a parameterized constructor.
3. Access Modifier

As the name suggests access modifiers in Java helps to restrict the scope of a class, constructor , variable , method or data member. There are four types of access modifiers available in java:

1. Default – No keyword required
2. Private
3. Protected
4. Public
5. Data Types

Data types specify the different sizes and values that can be stored in the variable. There are two types of data types in Java:

1. **Primitive data types:** The primitive data types include Boolean, char, byte, short, int, long, float and double.
2. **Non-primitive data types:** The non-primitive data types include Classes, Interfaces, and Arrays.
3. Array

Normally, an array is a collection of similar type of elements that have a contiguous memory location.

**Java array** is an object which contains elements of a similar data type. It is a data structure where we store similar elements. We can store only a fixed set of elements in a Java array.

Array in java is index-based, the first element of the array is stored at the 0 index.

There are two types of array.

1. Single Dimensional Array
2. Multidimensional Array
3. Selection Statement:

There are three types of selection statements.

if statement:

* Performs an action, if a condition is true; skips it, if false.
* Single-selection statement—selects or ignores a single action (or group of actions).

if…else statement:

* Performs an action if a condition is true and performs a different action if the condition is false.
* Double-selection statement—selects between two different actions (or groups of actions).

switch statement:

* Performs one of several actions, based on the value of an expression.
* Multiple-selection statement—selects among many different actions (or groups of actions).

1. **Three types of errors**

There are basically three types of errors that you must contend with when writing computer programs:

* Syntax errors
* Runtime errors
* Logic errors

Generally speaking, the errors become *more difficult* to find and fix as you move down the above list.

**Syntax errors**

In effect, syntax errors represent *grammar errors* in the use of the programming language.  Common examples are:

* Misspelled variable and function names
* Missing semicolons
* Improperly matches parentheses, square brackets, and curly braces
* Incorrect format in selection and loop statements

**Runtime errors**

Runtime errors occur when a program with no syntax errors asks the computer to do something that the computer is unable to reliably do.  Common examples are:

* Trying to divide by a variable that contains a value of zero
* Trying to open a file that doesn't exist

There is no way for the compiler to know about these kinds of errors when the program is compiled.

**Logic errors**

Logic errors occur when there is a design flaw in your program.  Common examples are:

* Multiplying when you should be dividing
* Adding when you should be subtracting
* Opening and using data from the wrong file
* Displaying the wrong message