Java Keywords:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **abstract** | **assert** | **boolean** | **break** | **byte** |
| **case** | **catch** | **char** | **class** | **const** |
| **continue** | **de**  **fault** | **do** | **double** | **else** |
| **enum** | **extends** | **final** | **finally** | **float** |
| **for** | **goto** | **if** | **implements** | **import** |
| **instanceof** | **int** | **interface** | **long** | **native** |
| **new** | **package** | **private** | **protected** | **public** |
| **return** | **short** | **static** | **strictfp** | **super** |
| **switch** | **synchronized** | **this** | **throw** | **throws** |
| **transient** | **try** | **void** | **volatile** | **while** |

Declare an **array** variable in one of two ways:

* With [] after the variable type: int[] values;
* With [] after the variable name: int values[];

Declaring arrays: String[] args; int[] numbers; int[] values = new int[10];

The **System.arraycopy()** method is an efficient way to copy the existing elements to the new array. For example:

Strings:

The String class has many methods, including: length, replace, substring, indexOf, equals, trim, split, toUpperCase, endsWith, etc.

Class/Object/Instance example:

//class

Class person {

//constuctor

Person() {}

//objects

Talk() {}

main {

//instance of class Person

Person Todd();

}//main

}//peson

<https://alfredjava.wordpress.com/2008/07/08/class-vs-object-vs-instance/>

**public** means that the method is visible and can be called from other objects of other types. Other alternatives are private, protected, package and package-private. See here for more details.

**static** means that the method is associated with the class, not a specific instance (object) of that class. This means that you can call a static method without creating an object of the class.

**Void -** means that the method has no return value. If the method returned an int you would write int instead of void

**Function overloading** (also method overloading) is a programming concept that allows programmers to define two or more functions with the same name and in the same scope.

An **overflow** error will occur if an element is attempted to be added to an already full stack or queue.

**NULL** means no value as in "unassigned". An object that is not instantiated should be NULL.   
In databases NULL means unassigned = no value It is different from 0 which is a value.

**Polymorphism** is the ability of an object to take on many forms. The most common use of polymorphism in OOP occurs when a parent class reference is used to refer to a child class object.

**Static:** It means that **only one instance of a static field exists**[1] even if you create a million instances of the class or you don't create any. It will be shared by all instances.

**Static** means you don’t need an instance of the class to call the method

Since static methods also do not belong to a specific instance, they can't refer to instance members (how would you know which instance Hello class you want to refer to?). staticmembers can only refer to static members. Instance members can, of course access staticmembers.

*Side note:* Of course, static members can access instance members **through an object reference**.

**Recursion** in java is a process in which a method calls itself continuously. A method in java that calls itself is called recursive method.

A **stack** is a data structure that allows data to be inserted (a 'push' operation), and removed (a 'pop' operation). Many **stacks** also support a read ahead (a 'peek' operation), which reads data without removing it. A **stack** is a LIFO-queue, meaning that the last data to be inserted will be the first data to be removed.

A **queue** is designed to have elements inserted at the end of the **queue**, and elements removed from the beginning of the **queue**.

A **heap** is a partially sorted binary tree.

A **Binary Search Tree** (BST) is a tree in which all the nodes follow the below-mentioned properties −

* The left sub-tree of a node has a key less than or equal to its parent node's key.
* The right sub-tree of a node has a key greater than to its parent node's key.

A **linked list** is a linear collection of data elements, called nodes, each pointing to the next node by means of a pointer. It is a data structure consisting of a group of nodes which together represent a sequence.

 A **hash table** uses a **hash** function to compute an index into an array of buckets or slots, from which the desired value can be found.

Sorting algorithms:

* Bucket sort
* Bubble sort – swaps each element starting from beginning
* Insertion sort
* Selection sort
* Heapsort
* Mergesort

**Tread vs. Process:**

1) A program in execution is often referred as process. A thread is a subset(part) of the process.

2) A process consists of multiple threads. A thread is a smallest part of the process that can execute concurrently with other parts(threads) of the process.

3) A process is sometime referred as task. A thread is often referred as lightweight process.

**algorithms, and memory allocation**