******

* *High level programming language*
* *Developed by Sun micro systems-1995*
* *Team headed by James Gosling*
* *Write once, Run anywhere.*
* *Source code -> Byte Code-> Object Code*

***Java is:-***

* Object Oriented
* Platform independent
* Simple
* Secure
* Portable
* Robust
* Multithreading
* Interpreted
* High performance
* Distributed
* Dynamic

***OOPs Concepts:-***

* Polymorphism
* Inheritance
* Abstraction
* Encapsulation
* Classes
* Objects
* Instances
* Methods
* Message passing

***Data Types:-***

|  |  |
| --- | --- |
| * Int | 4 byte |
| * float | 4 bytes |
| * char | 2 bytes |
| * double | 8 bytes |
| * long | 8 bytes |
| * Short | 2 bytes |
| * Byte | 1 byte |
| * Boolean | 1 byte |

***Type casting:-***

* Changing the type of one data type to another

***Condition Statements:-***

* If
* If-else
* Switch

***Looping Statement:-***

* For
* While
* Do-while

***Break:-***

* Used to jump out of the loop

***Continue:-***

* Breaks one iteration in a loop

***Arrays:-***

* Used to store multiple vales in a single variable

***Static variable:-***

* Used to refer common property of each object

***This operator:-***

* Reference variable which points to the current object

***Super:-***

* Also a reference variable which points to the super class object

***Method overloading:-***

* Same name- different parameter

***Method overriding:-***

* Used in inheritance
* Same name- same parameters
* Between 2 classes been inherited

***Inheritance:-***

* Capability of one class to inherit properties from another class

***Access Modifiers:-***

* Default- Scope is within the package
* Public- anywhere
* Private- Within a class
* Protected- Within the same package, inheritance is needed to access the variables from outside the package

***Abstraction:-***

* Hiding the implementation to outside world.
* Keyword abstract
* Applicable only to classes and methods(not to variables)
* There should be one abstract method which is also abstract
* There is no definition

***Interface:-***

* 100% abstract
* Keyword- implements
* One class can implements more than one interfaces

***Exception Handling:-***

* Run time errors
* Types:-
  + ArrayIndexOutOfBoundException
  + FileNotFoundException
  + ArithmeticException
  + NumberFormatException
  + NullPointerException
  + ClassCastException
* Try, catch, finally, throws throw
* Compulsory things need to write in the finally block

***File i/o:-***

* File is a class in java
* We can create a file
* Delete a file
* Restrict access to a file
* Copy file
* File f=new File();
* Read(), write()
* Stream:- flow of data or serious of data byte stream and character stream.

***Collection in Java:-***

* Class in java
* It can contain heterogeneous elements and modify the size dynamically
* Collection is an interface
* Types:-
  + List – ArrayList, LinkedList[it contains ordered elements but duplicate values]
  + Map – HashMap[Key-value pair]
  + Set –Treeset HashSet[No duplicates]
  + Queue – priority queue

***Wrapper class:-***

* Provide the mechanism to convert primitive into object and object into primitive
* 2 types:-
  + Autoboxing:- automatic conversion of primitive to object
  + Outboxing:- automatic conversion of objects

|  |  |
| --- | --- |
| * int | ***Integer*** |
| * boolean | ***Boolean*** |
| * char | ***Character*** |
| * float | ***Float*** |
| * double | ***Double*** |
| * short | ***Short*** |
| * long | ***Long*** |
| * byte | ***Byte*** |

***Multithreading:-***

* Thread is a class which allows to perform multitasking
* Deadlock:- When some process need a resource which is actually not available
* 2 ways to perform multitasking:
  + By extending thread class
  + By implementing runnable interface
* Thread priority is taken by JVM

***Generics:-***

* It enables single method declaration to a set of related methods or with a single class declaration a set of related types, respectively.

***Inner classes:-***

* Class inside another class
* Used for increasing reusability of codes
* 4 types of inner classes
  + Static inner class
  + Method inner class
  + Anonymous inner class
  + Regular inner class