OOP JAVA

- ~~Allen Kay~~   Core
- Father - Alan Kay

+ **Java Buzzwards**

① Simple → Syntax

② Object oriented →

③ Architectural Neutral → cos it doset generate code for targeting a system... does for JVM.

④ Portable → size of DT on all the sys is constant unlike c, can execute same code on all system (.class) (WORA).

⑤ Robust → error handling, strong memory management (garbage) strongly typed language - (compile time & runtime)

⑥ Dynamic → adaptable to envir, dynamic class loading at runtime.

⑦ Multithreaded → Perform multiple tasks simultaneously.

⑧ Secure →

⑨ High Performance → JIT,

⑩ Distributed →

* | Access modifier |

Public , Private , Protected , Default - No Keyword reqaird

Static
- To restrict the scope of class, constructo, variable, method or data members, or can say the visibility control of class/enum/interface.

	Same Package			Different Package		
Access modifier	same class	sub class	Non-Sub class	sub-class	Non-Sub class	
Private	A	NA	NA	NA	NA	
Default	A	A	A	NA	NA	
Protected	A	A	A	A	NA	
Public	A	A	A.	A	A	

* 12 modifiers

Public , Private , Protected , default , static , final, synchronized , adstract , native , strictfp , transient , volatile.

* Components of Java:-

JDK - A software development environment used to develop Java applications

JRE
JVM - Provides libraries, JVM needed to run application written in Java.

JVM - Executes Java Bytecode on any platform.

* <u>Meaning of Main method</u>

- The main method is entry point of any Java application.

- public static void main (String [] args) {}

- Public : Accessible from anywhere.
- Static : Can be run without creating instance of class
- void : Returns no values.
- main : The name of method.
- String Args [] : An array of cmd-line argument.

* <u>Data types</u>

* <u>Primitive data types</u> → (java.lang)

byte	1 byte	0	-128 to 127
boolean	1 bit	false	T or F
short	2 byte	'0000'	-32,768 to 32,767
char	2 byte	'u0000'	Single character/ASCII val
int	4 byte	0	in billions (2 b)
float	4 byte	0.0f	6,7 decimal
long	8 byte	0L	9×10^{18}
double	8 byte	0.0d	15 decimal

* <u>Local variable</u> does not consist of default value, it should be initialized. (var inside method)
* <u>Instance var</u> — Inside class but outsid method.

* <u>Static var</u> — Belong to class, single copy, class level access,

* **Loader**

- The class loader is a part of JVM that loads the class file into memories.

Types:-

@ **Bootstrap class loader**:- Loads the core Java liberary (rt.jar = java.lang, String, util)

ⓑ **Extension class loader**:- Loads classes from the extension libraries (javax.crypto.Cipher)

ⓒ **System class loader**:- Loads classes from the class path.
- application specific classes.

☆ **Components of JVM** : (JDK)

@ Java compiler → Compils java source code into bytecode.

ⓑ JVM → Executes Java byte code

ⓒ Java API libraries → A set of standard libraries (java.lang, java.util).

ⓓ Development tools → javadoc, jar, javap.

(☆) **Difference between JDK, JVM & JRE**

JDK → It is an development kit, used to develop Java applications.

JRE → Consist of JVM & standard lib but no development tools, it is used to run java application.

JVM → An abstract machine that executes the bytecode. (WORA)

* **Memory management of JVM**

· Heap memory → Stores Objects & their references.

Stack memory → Stores method call frames, local variables & return address.

Garbage collection → Automatically reclaims memory... no longer referenced.

eg: Arith obj = new Arith ();
                      ↑
              reference variable

obj creation = new arith (); allocates memory

Heap → object
Stack → reference to that object.

• Method area → Stores info about classes,
                      eg (structure , methods , fields)
              - also the metadata.

★ What are JIT compilers :-

Just In Time compiler -
                            It compiles the
byte code   into   Native code machine   at
runtime   to   improve   performance.

- Interview a general -

* We cannot override a method with a more
restrictive access modifier, eg. A protected
method in superclass can be overridden
with protected or Public modifier. in subclass.

- Is it possible to make Java class private?
→ No, a top-level class cannot be
protected or Private, should be public
for top level, or default.
→ Top-level → a class which is not nested
                inside other class.