Q- What does **programming language** means?

1. Language which is used to communicate or give instruction to computer.

Q- What is a **program**?

1. Set of instruction given to computer to solve a problem.

Application -Set of programs combined together to solve a specific problem for the user.

**Types of Application**

3 Types

1. Standalone application(Desktop application) – Application which does not require internet. For ex- Notepad, calculator.
2. Web Application – Application which requires internet to work. For ex – Amazon, Instagram etc.
3. Mobile Application – Specificially designed for the mobile.

Father of JAVA is James Gosling, made by Sun microsystem in 1995 which is now owned by oracle.

Q- What is java?

Java is object oriented programming language and which is used to develop standalone Application and web application.

Features of java

* Platform independent –because of the BYTE CODE as Java does not compile your high level code to low level code directly, in Java high level code is first converted to byte code which is platform independent and then this byte code we can give it to any machine having JVM which will convert this byte code to low level code which will be machine dependent.

That is why we say Java follows WORA principle(Write Once, Run Anywhere)

* Secure – because of BYTE CODE VERIFIER because byte code verifier will checks that this byte code is generated by javac compiler or not, only if the byte code is generated by javac compiler, it will allow JVM to convert that byte-code into Low-level code otherwise any modification happened in the byte code, byte code verifier will not allow JVM to convert it into low level code.
* Robust – because of features like exception handling and automatic memory management.
* Automatic memory management - In java programmer do not have to free the memory manually like we do in c,c++ ,here JVM(which has one component garbage collector) will free the memory, whose memory reference(address) has been lost.

NOTE

High level code – Machine independent

Low level code- Machine dependent

Byte code – Machine independent

JVM – Machine dependent

Platform means combination of software + hardware

Windows intel processor

Linux ryzen processor

Any program or software which can run on any hardware(processor) or software(operating system) will be called platform independent.

**JAVA ARCHITECTURE**

.java file .class file .exe file

(source code) (BYTE CODE) (Low level code)

JAVAC compiler JVM

(Compile our source code Here byte code verifier It interpretate

into byte code.) will check if this .class byte code

file is generated by javac into machine/low

compiler or not, if found level code.

generated by javac

compiler then allow JVM

to convert it into low-level

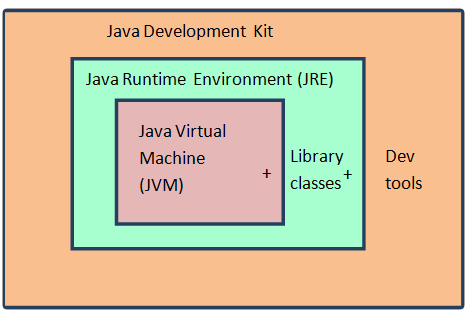
code otherwise not.

**JAVA ARCHITECTURE COMPONENTS**

JDK

JRE

JVM



**JDK(JAVA DEVELOPMENT KIT)**

It consist of development tools + JRE

Development tools are needed to develop a java program. For ex- javac compiler, debugger, monitoring tools etc ..

**JRE(JAVA RUNTIME ENVIRONMENT)**

So once we develop the java program we need an environment where we can run our java program, so JRE provide run- time environment to run our java program.

JRE contains Class libraries + JVM

Class libraries are collection of predefined classes which we use in our java program.For ex- Scanner class, Arrays class.

**JVM (JAVA VIRTUAL MACHINE)**

JVM converts byte code into low level code.

JVM has certain components like

1. Class loader – class loader loads the class into memory area.
2. Memory area – Stack, heap, class area, method area, pc-register.
3. Execution engine – Interpreter + JIT(just-in-time) compiler + GC(garbage collector)

1) Interpreter – line by line interpretate our code into binary.

2) JIT (just in time compiler) – it optimizes the performance of JVM

Q- how?

1. The Just-In-Time (JIT) Compiler in Java enhances the efficiency of the interpreter by

converting frequently used bytecode into binary code for the interpreter. The JIT compiler detects "hotspots" (frequently executed bytecode) and convert it into binary code.This allows the Java Virtual Machine (JVM) to execute the compiled code directly, avoiding repeated interpretation.

For ex- There is a loop which is going on for 100 times. So interpreter will come and do 5 to 10 repeatation of converting it into binary. Now JIT compiler will detect these repeated operation, so JIT compiler will come and convert these lines into binary code for once and all. So interpreter does not have to repeated operation.

So it will increase the efficiency of our interpreter.

3) Garbage collector – garbage collector will free the memory whose address has been lost.

**KEYWORDS**

Keywords are those words which are predefined & whose meaning is known by the compiler already.

For ex- public , static , if , for , int .

Note- All keywords in java are in lowercase.

**IDENTIFIER**

Name given to an entity in our program is called Identifier.

Entity means class, interface, variable, function etc.

**Rules** for Identifier

* Identifier cannot start with numbers.
* Identifier cannot have space.
* Only $ and \_ is allowed in special character.
* Keyword cannot be identifier.