1. **What is Java?**

Ans:- Java technology is both a programming language and a platform from ***Oracle Corporation****.*

1. **Why Java is called as “Platform”?**

Ans:- A platform is basically the hardware or software environment in which a program runs. Java provides software-based platform i.e. JVM, which can run applications developed using the Java Programming Language.

1. **What are the features of Java ?**

Ans:- Java has several powerful features that make it one of the most popular programming languages. Here are the key features:

**1. Platform Independence**

* Java follows the **"Write Once, Run Anywhere" (WORA)** principle.
* Java programs are compiled into **bytecode**, which can run on any system with a **Java Virtual Machine (JVM)**.

**2. Object-Oriented**

* Java follows **OOP principles** like **Encapsulation, Inheritance, Polymorphism, and Abstraction**.
* Everything in Java is treated as an **object**.

**3. Simple & Easy to Learn**

* Java’s syntax is **similar to C++ but removes complexities** like pointers and multiple inheritance.

**4. Secure**

* Java uses a **secure execution environment** with features like:
  + Bytecode verification
  + No explicit memory management (no pointers)
  + Security Manager for access control

**5. Robust (Strong & Reliable)**

* Java has strong memory management (Garbage Collection).
* It provides **exception handling** to handle runtime errors.

**6. Multi-threaded**

* Java supports **multithreading**, allowing multiple tasks to execute concurrently.

**7. High Performance**

* Though Java is slower than C/C++, it uses **Just-In-Time (JIT) Compilation** to improve performance.

**8. Distributed Computing**

* Java supports **Remote Method Invocation (RMI)** and **Enterprise Java Beans (EJB)** for distributed applications.

**9. Dynamic and Extensible**

* Java can dynamically load new classes at runtime.
* It supports dynamic linking with external libraries.

**10. Automatic Memory Management (Garbage Collection)**

* Java has an **automatic garbage collector** to free memory occupied by unused objects.

**11. Rich API and Large Standard Library**

* Java provides a vast set of APIs for **I/O operations, networking, utilities, database access (JDBC), XML parsing**, etc.

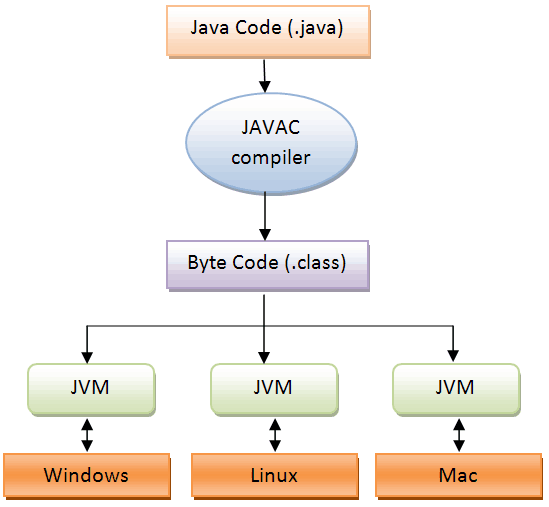
**12. Strong Community Support**

* Java has **huge community support**, with frequent updates, frameworks, and tools.

Would you like a detailed explanation of any of these features? 🚀

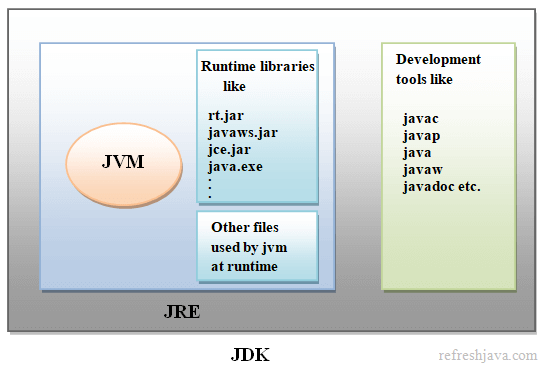
1. **How java is platform independent ?**

Ans:- Java program, once compiled , can be run on any platform without recompiling.



**5) What does JDK contain?**

Ans:-



6) **What is JVM ? What it does?**

Ans:- A Java virtual machine (JVM), an implementation of the Java Virtual Machine Specification, interprets compiled Java binary code (called bytecode) for a computer's processor (or "hardware platform") so that it can perform a Java program's instructions. Java was designed to allow application programs to be built that could be run on any platform without having to be rewritten or recompiled by the programmer for each separate platform.

JVMs are available for various platforms (i.e.JVM is platform dependent).

It is:

1. **A specification** where working of Java Virtual Machine is specified. But implementation provider is independent to choose the algorithm. Its implementation has been provided by Sun and other companies.
2. **An implementation** Its implementation is different for different platforms.
3. **Runtime Instance** Whenever you write java command on the command prompt to run the java class, an instance of JVM is created.

The JVM performs following operation:

* Loads code
* Verifies code
* Executes code

7) **Why JVM is called as “virtual machine”?**

Ans:- The JVM is "virtual" because it is generally implemented in software on top of a "real" hardware platform and operating system. All Java programs are compiled for the JVM. Therefore, the JVM must be implemented on a particular platform before compiled Java programs will run on that platform.

**8)What is the lifetime of Java Virtual Machine ?**

Ans:- When a Java application starts, a runtime instance of JVM is born. When the application completes, the instance dies.

**9)Is JRE different for different Platforms ?**

Ans:- yes

**10)Difference bet’n C++ and java in terms of object** **creation.**

Ans:- in C++ object can be created on stack as well as on heap (because java has given importance to late binding from the very beginning)

**11)What is automatic garbage collection ?**

Ans:- Automatic garbage collection is the process of looking at heap memory, identifying which objects are in use and which are not, and deleting the unused objects. An in use object, or a referenced object, means that some part of your program still maintains a pointer to that object. An unused object, or unreferenced object, is no longer referenced by any part of your program. So the memory used by an unreferenced object can be reclaimed.

**12)Can we force garbage collection programmatically?**

Ans:- no, we can at the most request for Garbage Collection using “System.gc()” or “Runtime.gc()”.

* 1. **What is the signature of main function in java ?**

Ans:- public static void main(String args[]) or public static void main(String …args)

**14)Who invokes main() function ?**

Ans:- JVM invokes main() function.

* 1. **Explain java data types.**

Ans:

Primitives

Data Type Size Description

byte **1 byte** Stores whole numbers from **-128 to 127**

short **2 bytes** Stores whole numbers from **-32,768 to 32,767**

int **4 bytes** Stores whole numbers from **-2,147,483,648 to 2,147,483,647**

long **8 bytes** Stores whole numbers from **-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807**

float **4 bytes** Stores fractional numbers. Range **1.40239846e-45f to 3.40282347e+38f**

double **8 bytes** Stores fractional number

Range

4.94065645841246544e-324 to 1.79769313486231570e+308s.

boolean **1 bit**  Stores true or false values

char **2 bytes** Stores a single character/letter or ASCII values

range - 0 to 65535

Reference Types:

Classes and interfaces

* 1. **How many types of comments are allowed in Java?**

Ans:

in java we have 3 types of comments

a) single line

//

b) multiline

/\* \*/

c) document comment ( in order to create documentation )

/\*\* \*\*/

* 1. **What is unicode character in java**

ans:

Unicode is a **universal character encoding standard** that represents almost all characters from different languages and symbols worldwide. In Java, characters are stored using **Unicode (UTF-16 encoding)**.

**Key Points:**

* Java uses **char** data type to store Unicode characters.
* Each char in Java takes **2 bytes (16 bits)** to support Unicode characters.
* Unicode characters are written as **\uXXXX**, where XXXX is a **hexadecimal code**.
  1. **What is object oriented programming**

Ans:

Object-Oriented Programming (**OOP**) is a programming paradigm based on the concept of **objects** that contain **data (attributes)** and **behavior (methods)**.

* 1. **What is class and object ? What are their real life examples?**

Ans:

Class is just a blueprint , an idea based on which object/s can be constructed. An **object** is an **instance of a class**. It has a **state (values of attributes)** and **behavior (methods of the class).**

**Real-Life Examples of Class and Object**

| **Real-World Entity** | **Class (Template)** | **Object (Instance)** |
| --- | --- | --- |
| **Car** | Car class | Toyota, BMW, Honda |
| **Mobile Phone** | Phone class | iPhone 14, Samsung Galaxy S23 |
| **Human** | Person class | John, Alice, Ravi |
| **Bank Account** | BankAccount class | Savings account, Checking account |