1)What is JVM ? Why is Java called the “Platform Independent Programming Language”?

Answer). A **Java virtual machine** (**JVM**) is a process virtual machine that can execute **Java** bytecode. Each **Java** source file is compiled into a bytecode file, which is executed by the **JVM**.

**When Java Code is compiled , a byte code(class file) is generated which is independent of the system. This byte code is fed to the JVM (Java Virtual Machine) which resides in the system. The byte code generated by the compiler can be interpreted by any JVM of any machine. Hence it is called Platform independent Language.**

### 2)What is the Difference between JDK and JRE ?

### JVM

JVM (Java Virtual Machine) is an abstract machine. It is a specification that provides runtime environment in which java bytecode can be executed.

JVMs are available for many hardware and software platforms. JVM, JRE and JDK are platform dependent because configuration of each OS differs. But, Java is platform independent.

The JVM performs following main tasks:

* Loads code
* Verifies code
* Executes code
* Provides runtime environment

JRE

JRE is an acronym for Java Runtime Environment.It is used to provide runtime environment.It is the implementation of JVM. It physically exists. It contains set of libraries + other files that JVM uses at runtime

### JDK

JDK is an acronym for Java Development Kit.It physically exists.It contains JRE + development tools

3)What is difference between object oriented programming language and object based programming language?

**Answer )**

**OOP** : follows Polymorphism, Inheritance and Encapsulation(PIE).

* **Examples**: Java, .Net so on. OOP is new comparing with OBP

**OBP** : follows Polymorphism and Encapsulation.

* **Examples**: Ada, Visual Basic (VB), and Fortran 90. OBP is old

**4)** Is Java a pure object orented language?

Answer)

### What are primitive and non-primitive data types?

**Answer:** A [data type](https://pc.net/glossary/definition/datatype) is a classification of data, which can store a specific type of information. Data types are primarily used in computer programming, in which variables are created to store data. Each variable is assigned a data type that determines what type of data the variable may contain.

The term "data type" and "primitive data type" are often used interchangeably. Primitive data types are predefined types of data, which are supported by the programming language. For example, [integer](https://pc.net/glossary/definition/integer), [character](https://pc.net/glossary/definition/character), and [string](https://pc.net/glossary/definition/string) are all primitive data types. Programmers can use these data types when creating variables in their programs. For example, a programmer may create a variable called "lastname" and define it as a string data type. The variable will then store data as a string of characters.

Non-primitive data types are not defined by the programming language, but are instead created by the programmer. They are sometimes called "reference variables," or "object references," since they reference a [memory](https://pc.net/glossary/definition/memory) location, which stores the data. In the [Java](https://pc.net/glossary/definition/java) programming language, non-primitive data types are simply called "objects" because they are created, rather than predefined. While an object may contain any type of data, the information referenced by the object may still be stored as a primitive data type.

**So java is not pure oop..**

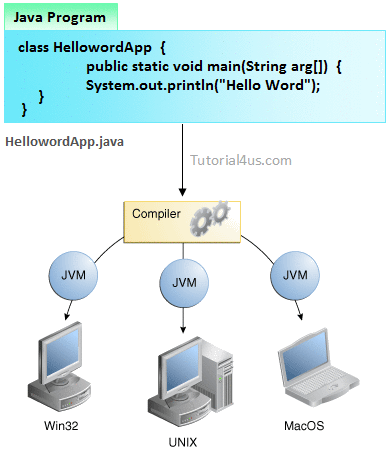
5)Are there any other platform independent language?

Answer)C/C++, C#, Java, Ruby, Python, Perl, PHP, Scheme, Lisp

6) What make a language platform independent?

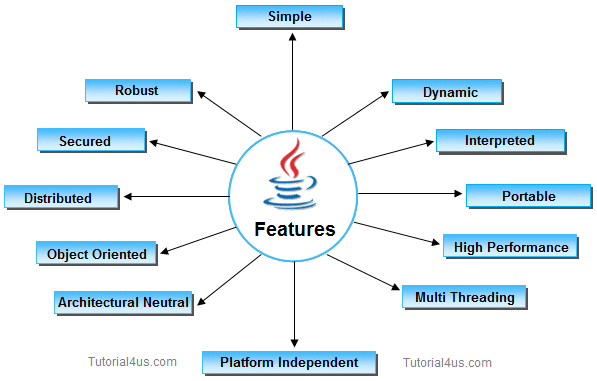
A programming language or technology is said to be platform independent if and only if which can run on all available operating systems with respect to its development and compilation. (Platform represents Operating System).

Java is a platform independent programming language, Because when you install jdk software on your system then automatically JVM are installed on your system. For every operating system separate JVM is available which is capable to read the **.class** file or **byte code**. When we compile your Java code then .class file is generated by javac compiler these codes are readable by JVM and every operating system have its own JVM so JVM is platform dependent but due to JVM java language is become platform independent.



**Note:**Java is platform independent but jvm is platform dependent.

7)What are feature of Java?



**Robust means reliable and no programming language can really assure reliability .It provides the powerful exception handling and type checking mechanism as compare to other programming languages. Compiler checks the program whether there any error and interpreter checks any run time error and makes the system secure from crash. All of the above features makes the java language robust Java compilers are able to detect many problems that would first show up during execution time in other languages.**

**Why Secure?**

**Protection from security attacks**

It allows developers to declare classes or methods as FINAL. We all know that any class or method declared as final can’t be overridden

**Byte code is another thing that makes Java more secure**

Every time when a user compiles the Java program, the Java compiler creates a class file with Bytecode, which are tested by the JVM at the time of program execution for viruses and other malicious files.

**Exception handling concept**

The concept of exception handling enables Java to capture a series of errors that helps developers to get rid of risk of crashing the system.

Why dynamic?

**Java** is considered as **Dynamic** because of Bytecode[a class file]. A source code writen in one platform, the same code can be executed in any platform[ which JDK is installed.]. And it also loads the class files at runtime. anything that happes at runtime is considered as **Dynamic**, so the **java** is.

#### 8)What are the three Edition of Java?

#### ****Java Platform, Standard Edition (Java SE)**** is the Java platform for developing client-side applications, which run on desktops, and applets, which run in web browsers.

#### ****Java Platform, Enterprise Edition (Java EE****) is the Java platform built on top of Java SE, which is used exclusively to develop enterprise-oriented server applications. Server-side applications include servlets, which are Java programs that are similar to applets but run on a server rather than a client. Servlets conform to the Java EE Servlet API.

#### ****Java Platform, Micro Edition (Java ME)**** is also built on top of Java SE. It is the Java platform for developing MIDlets, which are Java programs that run on mobile information devices, and Xlets, which are Java programs that run on embedded devices