1. **Give a brief history of Java.**

Java Programming Language was written by **James Gosling** along with two other person ‘**Mike Sheridan**’ and ‘**Patrick Naughton**’, while they were working at Sun Microsystem. Initially it was named oak Programming Language.

When Java 5 was released, once again it was having a lots of changes for the developer/marketers and need a new name. The next number sequence was 3, but calling Java 1.5 as Java 3 was confusing hence a decision was made to keep number and till now the legacy continues.

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

A java will only run on any JVM (Java Virtual Machine). Literally you can run same Java code on Windows JVM, Linux JVM, Mac JVM or any other JVM practically and e same result every time.

Typically, the compiled code is the exact set of instructions the CPU requires to “execute” the program. In Java, the compiled code is an exact set of instructions for a “virtual CPU” which is required to work the same on every physical machine.

So, in a sense the designer of e Java Language decided that the language and the compiled code was going to be platform independent, but since the code eventually has to run on a physical platform, they opted to put all the platform dependent code in JVM.

1. **Differentiate JDK, JRE and JVM.**

**JDK** (Java Development Kit) is the tool that is used to create a program, this has the one you see in the workspace that has different command tool to perform program creation. **JRE** (Java Runtime Environment) is where the compiling and running the program takes place, it compiles the written code, and it create a class then load it from the hard disk to interpret the written code and produce an output. **JVM** (Java Virtual Machine) is the one responsible in reads compiled byte codes that are platform-independent. It perform three main tasks, the loading of the code, verifying the code and the execution of the code.

1. **Why the main method is declared static?**

Declaring it as static is necessary because **main ()** is called by the JVM before any objects are made. Since it can be directly invoked via the class. Similarly, we use static sometimes for user defined methods so that we need not to make objects. **Void** indicates that the **main ()** method being declared and does not return value.

1. **Is Java purely object-oriented? Explain.**

Java is not purely object-oriented because it supports Primitive datatype such as int, byte, long and others to be used. Contrast with a pure Object-oriented language like Smalltalk, where there are no primitive types, and Boolean, int and method are all objects.