1. What is **JAVA**?

JAVA is a versatile and truly object-oriented language, which was developed by "Sun microsystems" and it was released in 1995, java is now owned and maintained by Oracle Corp.
Machine independency and rich standard library makes java more powerful and popular.

Why Use JAVA?
Java is a popular choice for various types of application, including web application, embedded system etc.
Not to mention it is a machine independent language which makes this frontend choice for every developer.

2. What is **JVM**?

It is a Java Virtual Machine and it is responsible to convert Java **Bytecode** into machine language.
Along with this it provides a runtime environment for java code.

3. What is **JRE**?

Stands for Java Runtime Engine. It is a software environment which is necessary to execute any java application on any system. JRE is a part of JDK
It can not be use to develop application but it is used for executing them. JRE can not be used independently

4. What is **JDK**?

Stands for Java Development Kit. A java program can not run without a JDK in a system. It contains both JRE and JVM.
It includes java development tools

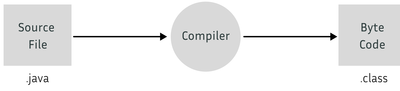
5. What is **Bytecode**?

Bytecode is different from machine code. Although a machine can not understand a bytecode. It is just the result of a compiled java program. It is saved with '.class' extension and it can run on any system if the system has JVM installed. It is an intermediate level of code and machine code is low-level of code.

6. How Java codes **RUN**?

It's done in two phase

1. Compilation



First we make a java program and save with '.java' extension. Compiling is done by 'javac' and compiler converts the code into bytecode. It is an intermediate form of java code and then the bytecode is saved in files with '.class' extension. Which is further used for execution of java code. Once bytecode is prepared, we can run that without the source code and even in any platform.

2. Execution

Now the work of JVM is to convert the bytecode into machine code. Finally the program is fully compiled and ready to show output in console screen.

7. Syntax

```
import java.util.*;
public class Classname
{
    public static void main(String args[])
    {
        //code goes here
    }
}
```

import java.util.*; this is the foremost line and we can use this to import inbuild classes in our code. 'java.util' stats that the package we want to import is java and '*' means we are importing everything from java **package**. We done this, so that we don't need to import every class. ' ; ' is a terminator of line. which is necessary to be in the end of every statement

public class 'public, is an access modifier that is used to control the visibility of classes, object etc. and 'public class' means that the class can be accessed by other classes.

Classname name of the class is to be written here and make sure that it is equal to the file you have saved.

static this is the keyword and it indicates that the method belongs to the class itself rather than to an instance of the class.

void this means that the main does not return any value.

main name of the method which will execute. Every program can have only one 'main'. this is the entry place where compiler starts compiling.

String args[] this is the parameter of main method. It's is an array of string used to pass command line arguments to the program.