1. What is Programming Language?
Ans: Language is a mode of communication used to share ideas and opinions with each other. And a Program is a set of instructions written in one place to convey to the computer how to perform a specific task. A programming language is a computer language used by programmers (developers) to communicate with computers. A programming language is mainly used to develop desktop applications, websites, and mobile applications. There are three broad divisions of programming language, namely, High-Level, Assembly-Level, and Machine-Level.

2. Why do we need a programming language?
Ans: Any Programming Language provides a method to interact with computers in a better way. Computers only understand Binary Language, i.e, 0s and 1s but it is not human-friendly. So we use a Programming Language to make it easier for humans to write and then translate it into Binary form so that computers understand it. Programming Language directs a computer to complete the commands written in it over and over again, so people do not have to do the task repeatedly. And also, Programming makes many real-world uses/applications come true and it enhances the power of computers, mobile solutions, and the internet.

3. What are the features of Java?
Ans: Some of the features of Java are:
It is simple in terms of building complex applications.
Java is a Class-based, Object-Oriented Language which means that all programs in Java have to follow this approach which makes it closer to the real world.
It is a portable/platform-independent/architecture-neutral language meaning that it follows the Write Once Run Anywhere(WORA) feature. Java code is written once in a system and compiled into bytecode which can be then executed by any platform given that Java interpreter is available on that system making it easily portable.Java code that runs on one platform does not need to be recompiled to run on another platform.
Java is a Robust(follows multi-threaded feature) and Dynamic language.
Object-Oriented - The features of object-oriented programming are supported by Java. Its object model is straightforward and flexible.
Platform independent - Because Java and C++ are platform independent, application programs created in one Operating system can run on any other Operating system. C and C++, however, are platform-dependent languages, making it impossible for application programs created in one Operating system to run in any other Operating system.
Simple - Because Java incorporates many C/C++ capabilities, it is simple to understand.
Secure - Java offers a variety of defences against malware and viruses. It guarantees that neither damage nor security will be compromised.
Portable - We have the idea of portability in Java. Java allows the same software to run on various platforms.
Robust - It assists us in identifying potential errors as soon as feasible during program development.
Multi-threaded - Java's multithreading programming capability enables you to create a program that executes multiple tasks concurrently.

Distributed -Java maintains the TCP/IP protocol and is therefore suitable for distributed Internet environments.

4. What is an Object?
Ans: An object is an entity with state and behaviour, such as a chair, bike, marker, pen, table, or car. It could be intellectual or physical (tangible and intangible).
It has 3 characteristics:
State: represents an object's data (value)
Behaviour: represents how an object behaves (or how it functions)
Identity: Usually, a distinct ID is used to implement an object's identification. The external user cannot see the value of the ID. However, the JVM uses it internally to uniquely identify each object.
Different ways of Object definitions

- An object is a real-world entity.
- An object is a runtime entity.
- The object is an entity that has a state and behaviour.
- The object is an instance of a class.

5. What is a class?
Ans: A class is a collection of items with similar characteristics. It serves as a model or blueprint from which things can be made. It makes sense as a whole. It cannot be bodily.
A class's instances are objects. A class serves as a model or blueprint from which new objects can be made. Therefore, a class's instance (or result) is an object.
In Java, a class could include:

- Fields
- Methods
- Constructors
- Blocks
- Nested class and interface

6. Explain about the main() method in Java?
Ans: The main () is the starting point for JVM to start the execution of a Java program. Without the main () method, JVM will not execute the program. The syntax of the main () method is:
public: It is an access specifier. We should use a public keyword before the main () method so that JVM can identify the execution point of the program.
public: An access specifier, that is. Before calling the main() method, we need to use the public keyword to let the JVM know where the program is actually being executed. Before the main() method, if we use private, protected, and default, the JVM won't be able to see it.
static: You can make a method static by using the keyword static. We should call the main() method without creating an object. Static methods are the method which is invoked without creating the objects, so we do not need any object to call the main() method.

void: In Java, every method has the return type. The void keyword acknowledges the compiler that the main() method does not return any value.
main(): It is a default signature that is predefined in the JVM. It is called by JVM to execute a program line by line and end the execution after completion of this method. We can also overload the main() method.
String args[]: The main() method also accepts some data from the user. It accepts a group of strings, which is called a string array. It is used to hold the command line arguments in the form of string values.

main(String args[])

Here, agrs[] is the array name, and it is of String type. It means that it can store a group of strings. Remember, this array can also store a group of numbers but in the form of a string only. Values passed to the main() method are called arguments. These arguments are stored in an args[] array, so the name args[] is generally used for it.