1. What is Programming language?

It is a set of instructions that instruct a computer how to carry out a task is the process of programming. It is easy to understand by human. There is numerous computer programming language available for use in programming.

2. Why do we need a programming language?

Here are some reasons why we need a programming language are the following:

- Communication with Computers: We need a programming language to communicate with computers because computer understands only machine level language (consists binary code i.e., 0, 1) which is very hard to understand for human then it come into picture as an intermediary, allowing human to write instructions in a more readable and understandable format.
- Abstraction: Programming languages provide a level of abstraction that hides the complexity of hardware details. This allows developers to focus on solving problems and creating software without having to deal with low-level hardware intricacies.
- Efficiency: Programming languages enable efficient communication and collaboration among developers. They provide tools and features that help programmers write code more quickly and with fewer errors.
- Problem Solving: Programming languages are essential for expressing algorithms and solving computational problems. They allow developers to create solutions to a wide range of tasks, from simple calculations to complex artificial intelligence algorithms.
- Portability: Code written in a programming language can often be executed on different platforms with minimal modifications. This portability is crucial for developing software that can run on various devices and operating systems.
- Maintainability: Programming languages support the creation of modular and maintainable code. This is essential for managing and updating software over time, especially in large-scale projects.
- Automation: Programming languages enable the automation of repetitive tasks and processes. This is particularly important for tasks that are time-consuming or prone to human error.

3. What are the features of java?

Java is a versatile, object-oriented programming language known for its platform independence, simplicity, and robustness.

Main key features of java are the following:

- Platform Independence (Write Once, Run Anywhere - WORA): Java code can run on any device that has a Java Virtual Machine (JVM). This is possible due to the "compile once, run anywhere" philosophy, making Java highly portable across different platforms.
- Object-Oriented: Java is designed based on object-oriented programming principles, emphasizing concepts such as encapsulation, inheritance, and polymorphism. This helps in organizing and structuring code in a modular and reusable way.
- Simple and Easy to Learn: Java was designed to be beginner-friendly, with a syntax similar to C++ but without some of its complexities. It includes automatic memory management through garbage collection, which simplifies memory handling for developers.
- Robust and Secure: Java's strong type checking, exception handling, and memory management contribute to its robust nature. Additionally, Java's security features, such as the sandbox environment for applets, help protect systems from malicious code.
- Multithreading: Java supports multithreading, allowing concurrent execution of multiple threads within a program. This is crucial for developing efficient and responsive applications, especially in scenarios with parallel processing requirements.
- Distributed Computing: Java provides extensive libraries for networking, allowing developers to create distributed applications easily. The Remote Method Invocation (RMI) and Java Naming and Directory Interface (JNDI) are examples of Java's support for distributed computing.
- Dynamic and Extensible: Java supports dynamic loading of classes and dynamic compilation, enabling the addition of new functionalities during runtime. This extensibility is useful for creating modular and flexible applications.
- Rich Standard Library: Java comes with a comprehensive standard library (Java API) that includes a wide range of classes and packages for various functionalities, from I/O operations to networking, data structures, and more.
- Community Support: Java has a large and active community of developers. This community support includes forums, documentation, and numerous third-party libraries and frameworks that enhance Java development.

- Automatic Memory Management (Garbage Collection): Java manages memory automatically through garbage collection, freeing developers from manual memory management tasks and reducing the risk of memory-related errors.

4. What is an object?

An object is an entity with state and behavior or an instance of a class.

5. What is a class?

A class is collection of objects that have similar traits, behavior and attributes.

6. Explain about the main() method in java?

In java, the main() method is a special method that serves as the entry point for java application. When we run a java program, JVM looks for the main() method in the class specified as the starting point, and it executes the code within this method. If it were not there, it would not be run by JVM because it would not be recognized by JVM.