1. **What is java?**

**Java is a high-level, class-based, object-oriented programming language that is designed to have as few implementation dependencies as possible.**

1. **What do you mean by platform independence?**

**Answer.** Platform independence means that we can write and compile the java code in our platform(e.g. Windows) and can execute the class on any other supported platform.

1. **What is the difference between JDK, JRE, and JVM?**

**Answer. JVM** is an acronym for Java Virtual Machine; it is an abstract machine that provides the runtime environment in which Java bytecode can be executed. It is a specification that specifies the working of Java Virtual Machine.

**JRE** stands for Java Runtime Environment. It is the implementation of JVM. The Java Runtime Environment is a set of software tools that are used for developing Java applications. It is used to provide the runtime environment.

**JDK** is an acronym for Java Development Kit. It is a software development environment that is used to develop Java applications and applets. It physically exists. It contains JRE + development tools.

1. **What are packages?**

**Answer.** Package in Java is a mechanism to encapsulate a group of classes, sub-packages and interfaces. Packages are used for: Preventing naming conflicts. For example there can be two classes with the name Employee in two packages, college..

### **What is an object?**

**Answer.** The Object is the real-time entity having some state and behaviour. In Java, an Object is an instance of the class having the instance variables like the state of the object and the methods as the behaviour of the object. The object of a class can be created by using the new keyword.

1. **What is the difference between superclass and subclass?**

**Answer.**

| **Superclass vs Subclass** | |
| --- | --- |
| **When implementing inheritance, the existing class from which the new classes are derived is the Superclass.** | **When implementing inheritance, the class that inherits the properties and methods from the Superclass is the Subclass.** |
| **Synonyms** | |
| **Superclass is known as a base class, parent class.** | **The subclass is known as the derived class, the child class.** |
| **Functionality** | |
| **A superclass cannot use the properties and methods of the Subclass.** | **A subclass can use the properties and methods of the Superclass.** |
| **Single-Level-Inheritance** | |
| **There is one Superclass.** | **There is one Subclass.** |
| **Hierarchical Inheritance** | |
| **There is one Superclass** | **There are many Subclasses.** |
| **Multiple Inheritance** | |
| **There are many Superclasses.** | **There is one Subclass.** |

1. **What is a deadlock?**

**Answer.** Deadlock is a situation that occurs in OS when any process enters a waiting state because another waiting process is holding the requested resource. Deadlock is a common problem in multi-processing where several processes share a specific type of mutually exclusive resource known as a soft lock or software.

1. **What are method overloading and method overriding?**

**Answer.**

| 1) | Method overloading is used *to increase the readability* of the program. | Method overriding is used *to provide the specific implementation* of the method that is already provided by its superclass. |
| --- | --- | --- |
| 2) | Method overloading is performed *within the class*. | Method overriding occurs *in two classes* that have an IS-A (inheritance) relationship. |
| 3) | In the case of method overloading, the *parameters must be different*. | In the case of method overriding, the *parameter must be the same*. |
| 4) | Method overloading is an example of *compile-time polymorphism*. | Method overriding is an example of *run time polymorphism*. |

1. **What is the meaning of the words public, static and void?**

**Answer. public** − This is the access specifier that states that the method can be accessed publicly.

**static** − Here, the object is not required to access static members.

**void** − This states that the method doesn’t return any value.

1. **What are the pillars of OOPs?**

### **Can we overload the constructors?**

**Answer.** Yes, the constructors can be overloaded by changing the number of arguments accepted by the constructor or by changing the data type of the parameters.

1. **What do you understand from a thread in Java?**

**Answer.** A Thread is a very light-weighted process, or we can say the smallest part of the process that allows a program to operate more efficiently by running multiple tasks simultaneously.

1. **What are the differences between C++ and Java?**

**Answer.**

| **Comparison Index** | **C++** | **Java** |
| --- | --- | --- |
| **Platform-independent** | **C++ is platform-dependent.** | **Java is platform-independent.** |
| **Mainly used for** | **C++ is mainly used for system programming.** | **Java is mainly used for application programming. It is widely used in window, web-based, enterprise and mobile applications.** |

| **Multiple inheritance** | **C++ supports multiple inheritance.** | **Java doesn't support multiple inheritance through class. It can be achieved by** [**interfaces in java**](https://www.javatpoint.com/interface-in-java)**.** |
| --- | --- | --- |
| **Operator Overloading** | **C++ supports** [**operator overloading**](https://www.javatpoint.com/cpp-overloading)**.** | **Java doesn't support operator overloading.** |

| **Call by Value and Call by reference** | **C++ supports both call by value and call by reference.** | **Java supports call by value only. There is no call by reference in java.** |
| --- | --- | --- |

1. **How many types of memory areas are allocated by JVM?**

**Answer.**

1. Class(Method) Area: Class Area stores per-class structures such as the runtime constant pool, field, method data, and the code for methods.

2. Heap: It is the runtime data area in which the memory is allocated to the objects

3. Stack: Java Stack stores frames. It holds local variables and partial results, and plays a part in method invocation and return. Each thread has a private JVM stack, created at the same time as the thread. A new frame is created each time a method is invoked. A frame is destroyed when its method invocation completes.

4. Program Counter Register: PC (program counter) register contains the address of the Java virtual machine instruction currently being executed.

5. Native Method Stack: It contains all the native methods used in the application.

1. **What is Platform?**

**Answer.** A platform is the hardware or software environment in which a piece of software is executed. There are two types of platforms, software-based and hardware-based. Java provides a software-based platform.

### **What is the static variable?**

**Answer.** The static variable is used to refer to the common property of all objects (that is not unique for each object), e.g., The company name of employees, college name of students, etc. The static variable gets memory only once in the class area at the time of class loading.

### **What is the constructor?**

**Answer.** The constructor can be defined as the special type of method that is used to initialise the state of an object. It is invoked when the class is instantiated, and the memory is allocated for the object. Every time an object is created using the new keyword, the default constructor of the class is called. The name of the constructor must be similar to the class name. The constructor must not have an explicit return type.

### **What is aggregation?**

**Answer.** Aggregation can be defined as the relationship between two classes where the aggregate class contains a reference to the class it owns. Aggregation is best described as a has-a relationship.

### **What is the interface?**

**Answer.** The interface is a blueprint for a class that has static constants and abstract methods. It can be used to achieve full abstraction and multiple inheritances. It is a mechanism to achieve abstraction.

### **What is Exception Handling?**

**Answer.** Exception Handling is a mechanism that is used to handle runtime errors. It is used primarily to handle checked exceptions. Exception handling maintains the normal flow of the program.

### **What are the various access specifiers in Java?**

**Answer.** In Java, access specifiers are the keywords that are used to define the access scope of the method, class, or variable. In Java, there are four access specifiers given below.

* **Public** The classes, methods, or variables which are defined as public, can be accessed by any class or method.
* **Protected** can be accessed by the class of the same package, or by the sub-class of this class, or within the same class.
* **Defaults** are accessible within the package only. By default, all the classes, methods, and variables are of default scope.
* **Private** The private class, methods, or variables defined as private can be accessed within the class only.

1. **What is synchronisation?**

Synchronisation is the mechanism that ensures that only one thread is accessed the resources at a time.