----------day2

1. What are the main principles of Object-Oriented Programming?

Abstraction, encapsulation, polymorphism, and inheritance

2. What is the difference between Object Oriented Programming language and Object Based Programming language?

Object-oriented language supports all the features of OOPs and Object-based language does not support all the features of OOPs like Polymorphism and inheritance

3. In Java what is the default value of an object reference defined as an instance variable in an Object?

Instance variables have default values. For numbers, the default value is 0, for Booleans it is false, and for object references it is null.

4. Why do we need constructor in Java?

Constructor in java is used to create the instance of the class. Constructors are almost like methods except for two things - its name is the same as the class name and it has no return type.

5. Why do we need default constructor in Java classes?

A default constructor in Java is created by the compiler itself when the programmer does not create any constructor. The purpose of the default constructor is to initialize the attributes of the object with their default values.

6. What is the value returned by Constructor in Java?

No, constructor does not return any value.

7. Can we inherit a Constructor?

No, constructor cannot be inherited in java.

In case of inheritance, child/sub class inherits the state (data members) and behavior (methods) of parent/super class.

8. Why constructors cannot be final, static, or abstract in Java?

The child class inherits all the members of the superclass except the constructors. In other words, constructors cannot be inherited in Java therefore you cannot override constructors. So, writing final before constructors makes no sense. Therefore, java does not allow final keyword before a constructor.

----------day3

1. What is the purpose of ‘this’ keyword in java?

This keyword refers to the current object in a method or constructor. The most common use of this keyword is to eliminate the confusion between class attributes and parameters with the same name (because a class attribute is shadowed by a method or constructor parameter).

2. Explain the concept of Inheritance?

Inheritance is a mechanism of acquiring the features and behaviors of a class by another class. The class whose members are inherited is called the base class, and the class that inherits those members is called the derived class.

3. Which class in Java is superclass of every other class?

The class named Object is the super class of every class in Java.

4. Why Java does not support multiple inheritance?

prevent ambiguity.

5. In OOPS, what is meant by composition?

A composition in Java between two objects associated with each other exists when there is a strong relationship between one class and another. Other classes cannot exist without the owner or parent class. For example, A Human class is a composition of Heart and lungs.

6. How aggregation and composition are different concepts?

In Aggregation, objects can remain in the scope of a system without each other. In a composition relationship, objects cannot remain in the scope of a System without each other.

7. Why there are no pointers in Java?

Java doesn't have pointers (in the C/C++ sense) because it doesn't need them for general purpose OOP programming. Furthermore, adding pointers to Java would undermine security and robustness and make the language more complex.

8. If there are no pointers in Java, then why do we getNullPointerException?

NullPointerException is a RuntimeException. In Java, a special null value can be assigned to an object reference. NullPointerException is thrown when program attempts to use an object reference that has the null value.

9. What is the purpose of 'super' keyword in java?

The super keyword refers to superclass (parent) objects. It is used **to call superclass methods, and to access the superclass constructor**. The most common use of the super keyword is to eliminate the confusion between superclasses and subclasses that have methods with the same name.

10. Is it possible to use this() and super() both in same constructor?

"this()" and "super()" **cannot be used inside the same constructor**, as both cannot be executed at once (both cannot be the first statement). "this" can be passed as an argument in the method and constructor calls.

11. What is the meaning of object cloning in Java?

Object cloning refers to **the creation of an exact copy of an object**. It creates a new instance of the class of the current object and initializes all its fields with exactly the contents of the corresponding fields of this object.