***PRACTICAL 3: QUESTIONS***

Q1. What is class and object in OOP?

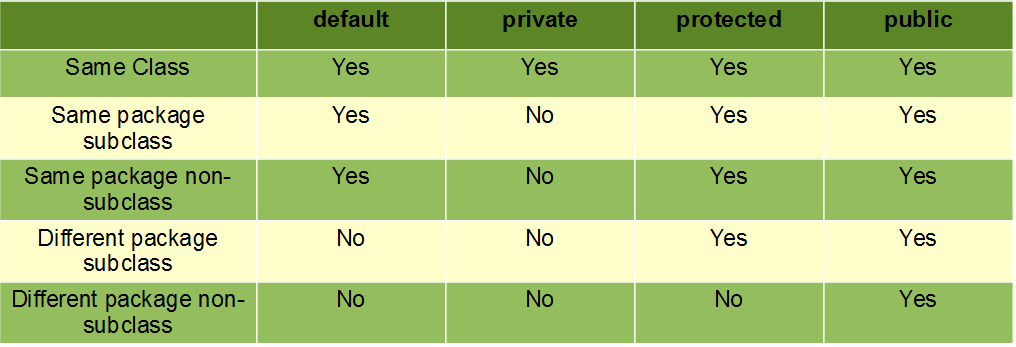
A class is a user defined blueprint or prototype from which objects are created. It represents the set of properties or methods that are common to all objects of one type

Object is a basic unit of Object-Oriented Programming and represents real life entities. A typical Java program creates many objects, which as you know, interact by invoking methods.

Q2. How to define any class in JAVA. Explain every class component in detail?

1. **Access Modifiers**: A class can be public or has default access (
2. **Class keyword:**class keyword is used to create a class.
3. **Class name:** The name should begin with an initial letter (capitalized by convention).
4. **Superclass(if any):** The name of the class’s parent (superclass), if any, preceded by the keyword extends. A class can only extend (subclass) one parent.
5. **Interfaces(if any):** A comma-separated list of interfaces implemented by the class, if any, preceded by the keyword implements. A class can implement more than one interface.
6. **Body:** The class body is surrounded by braces, { }.

Q3. What are the different access specifiers in JAVA and explain in details with examples?

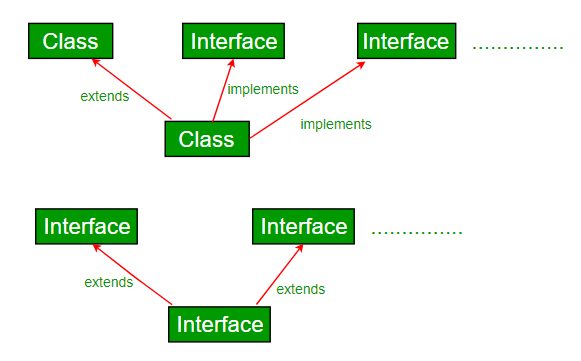


Q4. What is package and interface in JAVA?

Key Differences Between Packages and Interfaces in Java

1. A package is a group of classes and interfaces together whereas, an interface is a group of abstract methods.
2. Package is created using a keyword **package** whereas, an interface is created using a keyword **interface**.
3. If a class or an interface inside a package is to be used the package is to be imported while an interface has to be implemented.

Q5. What is the signature for sub class and class have implementation of interface?



Graphical user interface, text, application

Description automatically generatedText

Description automatically generated with low confidence

Q6. What is the signature of method in JAVA? Explain each of its components in detail?

Syntax :

Return\_type method\_name (Parameter list)

{

Method body

// variables defined inside this method will be called as local variables as their scope is limited till this function only

}

Q7. How to create and initialize the object in JAVA?

Syntax :

Class\_name obj\_name = new Class\_name();

Q8. How to access the members of class using objects

**class** demo{

**public** **int** x = 20;

}

**public** **class** Rough {

**public** **static** **void** main(String[] args) {

demo d1 = **new** demo();

System.***out***.println(d1.x);

}

}

Output : 20

Note : here the access specifier of the data member cannot be private.

Q9 What is difference between C++ and JAVA class

|  |  |
| --- | --- |
| JAVA | C++ |
| 1. In Java after the end of class semicolon should not be inserted   Example :  class Rectangle{  } | * In C++ after the end of the class semicolon should be inserted   Example :  class rectangle{  }; |
| 1. The name of the class should always begin with a capital letter | * It is not a compulsion to start the name of the class with a capital letter |
| 1. Object of the java class is created only by dynamic method   Example  Rect r1 = new Rect(); | * Object of the C++ class can be created using static and dynamic ways   Example:  Rect r1;  Rect r1 = new Rect() |
| 1. The class can only be public or default | * The class can be public, private or protected |
| 1. If nothing is mentioned as the access specifier then the access specifier default is activated | * If nothing is mentioned as the access specifier, then the access specifier private is activated |
| 1. There is no concept of virtual class in Java.   Instead it uses implement concepts | * Virtual class exists in C++ |
| 1. To inherit from a super class in java we use the following syntax   Class B extends A | * Syntax:   Public class B: class A,class C |
| 1. The parent class in java is called as super class and the child class is called as sub class |  |
| 1. If we want multiple inheritance then we use implement instead of extend   Syntax:  Class B implements A,D | * Syntax:   Public class B: class A,class C |

In java data fields are called fields and data functions are called as methods while in C++ they are called as data member and member function

Q10. Can we create object for main method class?

**public** **class** Rough {

**public** **static** **void** main(String[] args) {

Rough r1 = **new** Rough();

System.***out***.println("HELLO");

}

}

YES