

UNIT 2

2 marks

1. What is meant by Inheritance and what are its advantages?

Inheritance is a relationship among classes, wherein one class shares the structure or behavior defined in another class. The advantages of inheritance are reusability of code and accessibility of variables and methods of the super class by subclasses.

2. What is the use of super keyword? (Nov/Dec 2020)(Apr/May 2021)

This is used to initialize constructor of base class from the derived class and also access the variables of base class like `super.i = 10`.

3. What is the difference between superclass and subclass?

A super class is a class that is inherited whereas sub class is a class that does the inheriting.

4. What is protected function?

Protected members that are also declared as static are accessible to any friend or member function of a derived class. Protected members that are not declared as static are accessible to friends and member functions in a derived class only through a pointer to, reference to, or object of the derived class.

5. Define super class and subclass?

Super class is a class from which another class inherits. Subclass is a class that inherits from one or more classes.

6. What is role of access modifier in the member of a class in Java?

A private member is only accessible within the same class as it is declared. A member with no access modifier is only accessible within classes in the same package. A protected member is accessible within all classes in the same package and within subclasses in other packages.

7. What is protected method?

A protected method can be called by any subclass within its class, but not by unrelated classes. Declaring a method protected defines its access level. The other options for declaring visibility are private and public. If undeclared, the default access level is package.

8. What is final modifier?

The final modifier keyword makes that the programmer cannot change the value anymore. The actual meaning depends on whether it is applied to a class, a variable, or a method. final Classes- A final class cannot have subclasses.

final Variables- A final variable cannot be changed once it is initialized.

final Methods- A final method cannot be overridden by subclasses.

9. Why creating an object of the sub class invokes also the constructor of the super class? When inheriting from another class, `super()` has to be called first in the constructor. If not, the compiler will insert that call. This is why super constructor is also invoked when a Subobject is created. This doesn't create two objects, only one Sub object. The reason to have super constructor called is that if super class could have private fields which need to be initialized by its constructor.

10. What is an Abstract Class?

Abstract class is a class that has no instances. An abstract class is written with the expectation that its concrete subclasses will add to its structure and behavior, typically by implementing its abstract operations.

11. What are inner class and anonymous class?

Inner class: classes defined in other classes, including those defined in methods are called inner classes. An inner class can have any accessibility including private. Anonymous class: Anonymous class is a class defined inside a method without a name and is instantiated and declared in the same place and cannot have explicit constructors.

12. What is an Interface?

Interface is an outside view of a class or object which emphasizes its abstraction while hiding its structure and secrets of its behavior.

13. What is the difference between a static and a non-static inner class?

A non-static inner class may have object instances that are associated with instances of the class's outer class. A static inner class does not have any object instances.

14. What is the difference between abstract class and interface?

- a) All the methods declared inside an interface are abstract whereas abstract class must have at least one abstract method and others may be concrete or abstract.
- b) In abstract class, key word abstract must be used for the methods whereas interface we need not use that keyword for the methods.

Abstract class must have subclasses whereas interface can't have subclasses

15. Difference between class and interface. (Nov/Dec 2020)(Apr/May 2021)

Class and Interface both are used to create new reference types. A class is a collection of fields and methods that operate on fields. An interface has fully abstract methods i.e. methods with nobody. An interface is syntactically similar to the class but there is a major difference between class and interface that is a class can be instantiated, but an interface can never be instantiated.

16. What is an interface and state its use?(Nov/Dec 2019)

Interface is similar to a class which may contain method's signature only but not bodies and it is a formal set of method and constant declarations that must be defined by the class that implements it. Interfaces are useful for: a) Declaring methods that one or more classes are expected to implement b) Capturing similarities between unrelated classes without forcing a class relationship. c) Determining an object's programming interface without revealing the actual body of the class.

17. What is a cloneable interface and how many methods does it contain?

It is not having any method because it is a TAGGED or MARKER interface.

18. What is object cloning? (Nov/Dec 2018)(Nov/Dec 2019)

It is the process of duplicating an object so that two identical objects will exist in the memory at the same time

19. Define Package.

To create a package is quite easy: simply include a package command as the first statement in a Java source file. Any classes declared within that file will belong to the specified package. The package statement defines a name space in which classes are stored. If you omit the package statement, the class names are

put into the default package, which has no name.

20. Define Array list class.

The ArrayList class extends AbstractList and implements the List interface. ArrayList is a generic class that has this declaration:

```
class ArrayList<E>
```

Here, E specifies the type of objects that the list will hold. An ArrayList is a variable-length array of object references. That is, an ArrayList can dynamically increase or decrease in size. Array lists are created with an initial size. When this size is exceeded, the collection is automatically enlarged. When objects are removed, the array can be shrunk.

21. What is String in Java? Is String is data type?

String in Java is not a primitive data type like int, long or double. String is a class or in more simple term a user defined type. String is defined in java.lang package and wrappers its content in a character array. String provides equals() method to compare two String and provides various other method to operate on String like toUpperCase() to convert String into upper case, replace() to replace String contents, substring() to get substring, split() to split long String into multiple String.

22. Brief Inner class in Java with its syntax.

Java inner class or nested class is a class which is declared inside the class or interface.

We use inner classes to logically group classes and interfaces in one place so that it can be more readable and maintainable.

Syntax of Inner class

```
class Java_Outer_class{
    //code
    class Java_Inner_class{
//code } }
```

23. State the conditions for method overriding in Java. (APR/MAY 2019)

Following are the conditions to be considered while overriding a method properly

- The argument list should be exactly the same as that of the overridden method.
- The return type should be the same or a subtype of the return type declared in the original overridden method in the superclass. The access level cannot be more restrictive than the overridden method's access level. For example: If the superclass method is declared public then the overriding method in the subclass cannot be either private or protected.

24. Write the syntax for importing packages in a java source file and give an example.(APR/MAY 2019)

Creating a package in java is quite easy, simply include a package command followed by name of the package as the first statement in java source file.

```
package mypack; public class
employee
{
    String empId;String
    name;
}
```

The above statement will create a package with name mypack in the project directory. Java uses file system directories to store packages.

UNIT 2
16 marks

1	Define Inheritance. With diagrammatic illustration and java programs illustrate the different types of inheritance with an example (Nov./Dec.2018)(Nov/Dec 2019)
2	What is interface? With an example explain how to define and implement interface. (Nov/Dec 2020)(Apr/May 2021)
3	Differentiate method overloading and method overriding. Explain both with an example program.
4	Explain about the object and abstract classes with the syntax.(Nov/Dec 2019)
5	Discuss in detail about inner class. With its advantages.
6	What is meant by object cloning? Explain it with an example.
7	Explain how inner classes and anonymous classes works in java program.
8	What is a Package? What are the benefits of using packages? Write down the steps in creating a package and using it in a java program with an example.
9	Explain arrays in java with suitable example.
10	How Strings are handled in java? Explain with code, the creation of Substring, Concatenation and testing for equality.
11	Write a Java program to create a student examination database system that prints the mark sheet of students. Input student name, marks in 6 subjects. This mark should be between 0 and 100. (Nov./Dec.2018)
	<p>If the average of marks is ≥ 80 then prints Grade <u>A</u></p> <p>If the average of marks is < 80 and ≥ 60 then prints Grade <u>B</u></p> <p>If the average of marks is < 60 and ≥ 40 then prints Grade <u>C</u></p> <p>Else prints Grade <u>D</u></p>
12	Explain hierarchical and multi-level inheritances supported by Java and demonstrate the execution order of constructors in these types.(Apr/May 2019)
13	<p>i) Explain simple interfaces and nested interfaces with examples</p> <p>ii) Present a detailed comparison between classes and interfaces (Apr/May 2019)</p>