

## 1. What do you mean by class and object?

**Answer:** **Class:** A class is a blueprint or prototype or template from which objects are created. A class is description of same type of object which has common attribute and behaviors and common relationship with other classes. Example –

```
class MyClass { //field, constructor, and method declarations }
```

**Object:** An object is an instance of a class. It has both state and behavior. creates an object –

```
Point originOne = new Point(23, 94);
```

## 2. What is Encapsulation?

**Answer:** Encapsulation is the mechanism that binds together code and data it manipulates and keeps both safe from outside interference and misuse.

## 3. What is Inheritance?

**Answer:** Inheritance is the mechanism for creating one or more subtypes from an existing type. In Java technology, a class represents a type. Inheritance allows us to create subclasses from existing class

Class a {...}

Class b extends a {...}.

## 4. What are the benefits of inheritance?

Benefits of inheritance are as follows:

- \* Enables the creation of specialized types.
- \* Eliminates duplication. (Reusability)
- \* Assists maintainability.

## 5. What do you mean by narrowing and widening? Give example in each case.

**Answer: Narrowing:** Converting a boarder data type to narrower data type is called narrowing, which loss precision.

Ex: double d= 10.55555;

Int x= (int) d;

**Widening:** Converting a narrower data type to a boarder data type without loss of information is called widening.

Ex: int x=3, y=2;

Double d= (double)x/y;

## 6. What do you mean by "super ()" and "this"?

**Answer:** "super ()" refers to invoking of super class constructor.

Example: class B extends A{

```
B(int b){super(10); //call super class}}
```

"this" refers to current class object.

Example: class A(A){}

```
A(int a){this(); //call to self}}
```

## 7. What do you mean by method overloading and method overriding?

**Answer: Method overloading:** It means methods within a class have the same name but they have different parameter lists. In the code sample, draw (String s) and draw (int i) are distinct and unique methods because they require different argument types.

**Method overriding:** It means the ability of a subclass to override a method allows a class to inherit from a super class whose behavior is "close enough" and then to modify behavior as needed. The overriding method has the same name, number and type of parameters, and return type as the method it overrides.

## 8. What is the difference between Interface and Abstract class?

**Answer:**

Interface	Abstract class
Java interface should be implemented using keyword "implements";	A Java abstract class should be extended using keyword "extends".
A Java class can implement multiple interfaces	It can extend only one abstract class.
Variables declared in a Java interface is by default final and static.	An abstract class may contain non-final variables.
Methods does not contain body part	Methods may or may not contain body part

## 9. What do you mean by polymorphism?

**Answer:** Polymorphism is the feature that allows one interface to be used for general class actions. It enables the same method to behave differently on different class.

## 10. What is JVM?

**Answer:** JVM means the Java Virtual Machine. JVM is a platform-dependent execution environment that converts Java byte code into machine language and executes it. (A JVM implementation executes Java technology applications. Which consist of compiled Java classes.) Compiled Java classes consist of byte code, so a JVM implementation loads the classes that compose a Java Technology application and executes the byte code contained in the classes.

**11. Define package. What are the advantages of package? Write down the name of default name of package.**

**Answer:** A package is a Java mechanism for organizing (related) class in a same directory. A package provides a namespace for the classes it contains.

**Advantages of Packages**

- To bundle classes and interfaces
- Namespace collision is minimized.
- Packages provide reusability of code.

The default package name is java.lang.

**12. What is the use of abstract, final and static keyword?**

**Answer: Use of abstract:**

- a. With class –indicates that the class cannot be instantiate.
- b. With method – indicates that the method must be over riding in the sub class of that abstract class.

**Use of final:**

- a. With class –indicates that the class cannot be subclass
- b. With variable–indicates that the value first assign cannot be change.
- c. With method–indicates that method body cannot be changed.
- d. With object of a class –indicates that object reference cannot be changed but value can be changed.

**Use of static keyword:**

- a. Static variable of a class are accessed by static methods only.
- b. Static methods of a class are invoked by static methods only.
- c. Static methods does not have "this".
- d. Static method cannot be overridden by non-static method.

**13. What do you mean by logical and short-circuit operator or What is the difference between & and && ?**

**Answer:** The difference between & and && is that the conditional or short-circuit operator (&&) will not bother to evaluate the right hand operand if the left hand operand false.

But the logical operator (&) will evaluate both the operator.

**14. What is a constructor? What do you mean by default constructor?**

**Answer:** A class contains constructors that are invoked to create objects from the class blueprint. Constructor declarations look like method declarations—except that they use the name of the class and have no return type. For example

```
Class Constructor() {  
    String symbol;  
    Constructor(String stockSymbol) {symbol = stockSymbol}  
}
```

Default constructor is a no argument constructor. If we do not declare a constructor the Java programming language provides it that takes no arguments and has an empty body. For example

```
Class Constructor() {  
    String symbol;  
    //no constructor declared  
}
```

**15. What is an Array? How many ways can create an Array in java?**

**Answer:** Array is collection of homogeneous data (In the Java programming language, an array is an object) even when the Array is made up of primitive types and as with other class type.

Array are used to group objects of the same type.

Array can be declared of any type, either primitive or class.

- An Array of char primitive can be declared as follows:

Char[] s;

- An Array point class object can be declared as follows:

Point[] p;

Array can be declared using the square brackets after the variable name.

Char s[];

Point p[];

**16. Write down the five key words in java.**

**Answer:** Five key words of java are as follow –

1. for 2. do 3. while 4. if 5. Else

**What is difference between primitive data type and wrapper class?**

**Answer:** Java supports eight basic data types which known as primitive types. Those are byte, short, integer, long, float, double, char, boolean. Java programming language provides wrapper classes to manipulate primitive data elements as objects. Such data elements are wrapped in an object created around them. Each primitive data types has corresponding wrapper class in the java.lang package.

The following two statements illustrate the difference between a primitive data type and an object of a wrapper class:

```
int x = 25;
Integer y = new Integer(33);
```

**18. What is the difference between local variable, instance variable and static variable?**

**Answer:** **Local variable:** Variables that are declared in a function are called local variables. They are called local because they can only be referenced and used locally in the function in which they are declared. In the method below miles is a local variable. For example-

```
private static double convertKmToMi(double kilometers) {
    double miles = kilometers * MILES_PER_KILOMETER;
    return miles;
}
```

**Instance variable:** Instance variables are any variables, without "static" field modifier, that are defined within the class body and outside any class's methods body. Instance (field) variables can be seen by all methods in the class.

```
class A {
    B b=new B();
}
```

**Class/static variable:** Class/static variables are declared with the static keyword in a class, but outside a method. There is only one copy per class, regardless of how many objects are created from it.

```
public class MyClass {
    public static final int MY_CONSTANT = 0;
}
```

**19. What is the use of finalized () and finally block?**

**Answer:** The automatic garbage collector calls the finalized() method that is eligible for garbage collector before actual destroying the object. A finally block is an optional block that exist after the last catch block and always executed wheatear or not exception is caught.

**20. Java supports how many types of access modifier explain them.**

**Answer:** Java supports four types of access modifier

- Public
- Protected
- Private
- Default

**Accessibility Criteria:**

Modifier	Same Class	Same Package	Subclass	Universe
Private	Yes			
Default	Yes	yes		
Protected	Yes	yes	yes	
public	Yes	yes	yes	yes

**21. What is the difference between 'equals()' methods and '==' operator?**

**Answer:**

**Equals():**

equals() method checks the equality of the content.

**== :**

"==" checks the equality of object reference.

**22. Explain the difference between pass by value and pass by pass by reference.**

**Answer:** **Pass by Reference** means the passing the address itself rather than passing the value .

**pass by value** means passing a copy of the value as an argument. In Java the arguments are always passed by value.

**23. What is an Exception?**

**Answer:** An *exception* is a throw able, which occurs during the execution of a program that disrupts the normal flow of the program's instructions. The Java programming language uses *exceptions* to handle errors and other exceptional events. Java programming language provides two broad categories of exceptions known as checked and unchecked exceptions.

When java program failed to execute program then it give throwable message .Throwable divide into two parts:

Exception and Error . Exception throw exception and Error throw

**24. Explain the purpose of garbage collection that the JVM uses.**

**Answer:** The essential purpose of garbage collection is to reclaim the memory space that was occupied by the objects that are no longer required for the application. Garbage collection is done in java by a program called garbage collector. An object in the application is subjected to garbage collected when it is unreachable to the application.

#### 25. Checked Exceptions vs. Unchecked Exceptions.

**Answer: Checked Exceptions**

- A checked exception is a subclass of Exception excluding class Runtime Exception and its subclasses.
- Compiler checks to see if these exceptions have been properly caught or not. Else the code doesn't compile. Thus, a program is forced to deal with the situations where an exception can be thrown.
- Checked exceptions must be either declared or caught at compile time.

**Unchecked Exceptions**

- Unchecked exceptions are Runtime Exception and all of its subclasses along with class java.lang.Error and its subclasses also are unchecked.
- A program does compile without these exceptions being handled during compile time.

#### 26. What do you mean by Assertion?

Assertion is a subclass of Error class. By the assertion we can test unusual condition during testing of a program.

By Java\_1.4 Test, we can enable Assertion. By default Assertion is disabled.

#### 27. What do you mean by Inner class ?

**Answer:** The Java programming language allows you to define a class within another class. Such a class is called a *nested class* and is illustrated here:

```
class OuterClass { ...
    class NestedClass { ...
    }
}
```

#### 28. What do you mean by autoboxing?

**Answer:** During assignment, the automatic transformation of primitive type (int, float, double etc.) into their object equivalents or wrapper type (Integer, Float, Double, etc) is known as Autoboxing.

```
int x = 25;
Integer y = x; //It is autoboxing.
```

#### 29. What is type casting?

**Answer:** Conversion of data from one type to another type is known as type casting. In java one object reference can be type cast into another object reference. For example

```
public class CastExample
{
    public static void main(String arg[])
    {
        String s="27";
        int i=Integer.parseInt(s);
    }
}
```

#### 30. What do you mean by instance of operator?

**Answer:** The instance of operator is used to check the type of an instance of an object.

Eg:

```
String s = "xyz"
```

If (s instanceof java.lang.String) returns TRUE.