Java Descriptive Questions & Answers (30)

1.What do you mean by object and class?

Ans. Any type of class and variable is called object. Object have state and behaviors.

Class is a template for creating object. Class defines the properties and behaviors for objects.

2: What do you mean by encapsulation?

Answer: *encapsulation is a process of packaging variables and methods into a single unit.* The main advantage of encapsulation is the ability to hide the implementation of a class.

3.What do you mean by inheritance?

Ans. Inheritance can be defined as the process where one class acquired the properties of another. We have to use extends key word for creating inheritance.

4: What are the benefits of Inheritance?

Answer:

- A code can be used again and again
- It helps to avoid redundancy of code in subclass by inheriting from super-class.

5. What do you mean by narrowing and widening? Give an example of them.

When we cast dig data type into small data type is called narrowing.

Ex: double d = 10.55555;

```
int x = (int)d;
```

When we cast small data type into dig datatype is called widening.

```
Ex:       int x =3, y = 2;
Double d =(double)x/y;
```

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

this( ) & super( ) is a keyword.

"super( )" is used to invoke the super class's methods and constructors.

"this( )" is used to invoke a constructor of a same class , its pointing the same class object.

7: What do you mean by method overloading and method overriding?

Answer:

When two and more methods are declared in the same class, method name is same but different parameter is called method overloading.

Example:

```
int sum(int x, int y) {
        //bodypart;
}
Int sum(int x, int y, int z){

    }
```

over-ridding - same method names with same arguments and same return types associated in a class and its subclass.

Example:

```
interface A{
int sum(int x, int y);
}
class B implements A{
public int sum(int x, int y){
        }
}
```

8. Write down the difference between interface and abstract class?

Interface: An interface is a collection of abstract methods.

- **(i) Using key word "Interface " have to declared a name of Interface.**
- **(ii) Support multiple inheritance.**
- **(iii) It can be final and static.**

Abstract:

- **(i) There is no body in abstract method.**
- **(ii) It must be over-ridden.**
- **(iii) It can never be final and static.**
- **(iv) It allows abstract method and no abstract method.**

9: What do you mean by polymorphism?

Polymorphism in Java is a concept by which we can perform a *single action in different ways*. Polymorphism is derived from 2 Greek words: poly and morphs. The word "poly" means many and "morphs" means forms. So polymorphism means many forms.

There are two types of polymorphism in Java: compile-time polymorphism and runtime polymorphism. We can perform polymorphism in java by method overloading and method overriding.

10: What is JVM?

Answer:

JVM stands for Java Virtual Machine. This is nothing but a software. JVM creates a run time system internally that helps the execution of code by –

(a) Loading the class file.
(b) Loading the byte code.
(c) Performing the garbage collection.

/\/\/\/\/\/\/\/\/\/\/\

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

Answer:

A package is a collection of classes and interfaces.

1) It helps us to about naming conflict.

2) Package name can be use to identify our classes.

The default package name in java is java.lang

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

Answer:

Abstract:

(i) There is no body in abstract method.
(ii) It must be over-ridden.
(iii) It can never be final and static.
(iv) It allows abstract method and no abstract method.

Use of final:

a. With class- indicates that the class cannot be subclass

b. With variable – indicates that the value first assign can not be change.

c. With method – indicates that method body cannot be changed.

Use of static keyword:

static is a non-access modifier in Java which is applicable for the following:

1. blocks
2. variables
3. methods
4. nested classes

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 operands.

14: What is a constructor? What do you mean by default constructor?

Answer:

Constructor:

If a method in a class have the same name in that class and must not return a value then we can consider the special method as a constructor. The primary purpose of a constructor is initializing the instant variable.

Default constructor:

If we do not define any constructor for a class the compiler will supply a default constructor in the class, which actually does nothing. The default constructor is also described as no argument constructor. Because it requires no argument to be specified when it called. When we create an object of a class the default constructor is automatic invoked.

Ex: TestClass tc = new TestClass();

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

Answer:

Array:

An **array** is an indexed collection of data elements of the same type.

We can declare an array in several ways.

```
int[] arr = new int[5];
int arr[] = new int[5];
Object arr3= Array.newInstance(Integer.TYPE, 4);
int[] arr = {0,1,2,3,4};                    //dense array
int[][] arr2 = new int[length1][length2];          //2D array
Object list1 = new Object[5];       //An array of objects
```

16: Write down ten keywords in Java.

Answer:

strictfp, switch, synchronized, native, transient, volatile, while, instanceof, goto, enum, throws etc.

17: What is the difference between primitive data type and wrapper class?

Answer:

Java provides eight primitive data types. Primitive data types in java are not object. In order to manipulate this values/data types as objects the java.lang package provides a wrapper class for each of this primitive data types. All wrapper classes are final.

Primitive Data Type	Wrapper Class
boolean	Boolean

byte	Byte
char	Character
short	Short
int	Integer
long	Long
float	Float
double	Double

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

Answer:

Local variable :

All the variables declared inside a method which have local scope in that method are called local variable. Local variable are also called stack variable because they live on stack.

Instance variable:

The variables that declared inside a class but outside any methods which have global scope are called instance variable.

Static variable:

The variable which are declared with a modifier “static” and which scope belongs to class are called static variable.

19: What is the use of finalize ( ) 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 whether or not exception is caught.

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

Answer:

Java supports four types of access modifier.

a. public
b. protected
c. private
d. 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.

Q-22: Explain the difference between pass by value and pass by reference.
Answer:

Pass by Value

Actual parameter expressions that are passed to a method are evaluated and a value is derived. Then this value is stored in a location and then it becomes the formal parameter to the invoked method. This mechanism is called pass by value and Java uses it.

Pass by Reference

In pass by reference, the formal parameter is just an alias to the actual parameter. It refers to the actual argument. Any changes done to the formal argument will reflect in actual argument and vice versa.

23: What is an Exception?
Answer:

An exception is an unwanted or unexpected event, which occurs during the execution of a program at run time, that disrupts the normal flow of the program's instructions.

Q-24: Explain the purpose of garbage collection that the JVM uses.
Answer:
Garbage collection in Java identifies and discards the objects that are no longer needed by a program so that their resources can be reclaimed and reused. A Java object is subject to garbage collection when it becomes unreachable to the program in which it is used. The finalize() method is called by garbage collector when it determines no more references to the object exists.

Q-25: Checked Exceptions vs. Unchecked Exceptions.
Answer:
Checked Exceptions

- A checked exception is subclass of Exception excluding class RuntimeException 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 RuntimeException 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.

Q-26: What do you mean by Assertion?
Answer:
Assertions are simple check assumption made at the beginning of the program to ensure the program is true throughout provided by the Java language. For example, the range of age should be in between 18 and above; or cannot be more than 50.

An Assertion contains a Boolean expression which is set to be true in the beginning of the program.
Syntex:
assert<Boolean exp.>; // simple form
assert <Boolean exp.>:<message exp.>; //augmented form

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 Inner *class*.

```
class OuterClass {
   ...
   class InnerClass {
      ...
   }
}
```

28: What do you mean by auto boxing?
Answer:

The automatic conversion of primitive int type into a wrapper class object is called autoboxing. It does not require to type cast the int value. The modification of primitive wrapper objects is done directly. The following example illustrates autoboxing:

```
int number;
Integer intObject;
number = 1;
intObject = 2;
number = intObject;
intObject = number;
```

29: What is type casting?

Answer:

Type casting means to explicitly convert one type to another.
For example, the following lines:

```
double x = 5.0;
int y = x;
```

will produce an error message, because of a possible loss of precision (any decimals will get lost when converting to int). The following will work however:

```
double x = 5.0;
int y = (int) x;
```

The programmer is forcing the Java compiler to accept the conversion; saying, in effect: "Please do this anyway, I know what I am doing".

30: What do you mean by instanceof operator?

Answer:

The instanceof 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.
```