1) What is JAVA?

Answer: Java is a high-level programming language and is platform-independent.

Java is a collection of objects. It was developed by Sun Microsystems.

There are a lot of applications, websites, and games that are developed using Java.1.

2) What are the features of JAVA?

Answer: Features of Java are as follows:

OOP concepts

Object-oriented

Inheritance

Encapsulation

Polymorphism

Abstraction

Platform independent: A single program works on different platforms without any modification.

High Performance: JIT (Just In Time compiler) enables high performance in Java. JIT converts the bytecode into machine language and then JVM starts the execution.

Multi-threaded: A flow of execution is known as a Thread. JVM creates a thread which is called the main thread. The user can create multiple threads by extending the thread class or by implementing the Runnable interface.

3) How does Java enable high performance?

Answer: Java uses Just In Time compiler to enable high performance. It is used to convert the instructions into bytecodes.

4) Name the Java IDE’s ?

Answer: Eclipse and NetBeans are the IDE’s of JAVA.

5) What do you mean by Constructor?

Answer: When a new object is created in a program a constructor gets invoked corresponding to the class.

The constructor is a method which has the same name as the class name.

If a user doesn’t create a constructor implicitly a default constructor will be created.

The constructor can be overloaded.

If the user created a constructor with a parameter then he should create another constructor explicitly without a parameter.

6) What is meant by the Local variable and the Instance variable?

Answer:

Local variables are defined in the method and scope of the variables that exist inside the method itself.

Instance variable is defined inside the class and outside the method and the scope of the variables exists throughout the class.

7) What is a Class?

Answer: All Java codes are defined in a Class. It has variables and methods.

Variables are attributes which define the state of a class.

Methods are the place where the exact business logic has to be done. It contains a set of statements (or) instructions to satisfy the particular requirement.

8) What is an Object?

Answer: An instance of a class is called an object. The object has state and behavior.

Whenever the JVM reads the “new()” keyword then it will create an instance of that class.

9) Difference between String, String Builder, and String Buffer.

Answer: String: String variables are stored in a “constant string pool”.

Once the string reference changes the old value that exists in the “constant string pool”, it cannot be erased.

String Buffer: Here string values are stored in a stack. If the values are changed then the new value replaces the older value.

The string buffer is synchronized which is thread-safe.

Performance is slower than the String Builder.

String Builder: This is the same as String Buffer except for the String Builder which is not threaded safely that is not synchronized.

So obviously the performance is fast.

10) Explain about Public and Private access specifiers.

Answer: Methods and instance variables are known as members.

Public: Public members are visible in the same package as well as the outside package that is for other packages.

Private: Private members are visible in the same class only and not for the other classes in the same package as well as classes in the outside packages.

11) Difference between Default and Protected access specifiers.

Answer:

Default: Methods and variables declared in a class without any access specifiers are called default.

Default members in Class A are visible to the other classes which are inside the package and invisible to the classes which are outside the package.

Protected : Protected is the same as Default but if a class extends then it is visible even if it is outside the package.

12) Difference between HashMap and HashTable.

Answer:

HashMap Hash Table

1) Methods are not synchronized 1) Key Methods are synchronized

2) Not thread safety 2) Thread Sefty

3) Iterator is used to iterate the values 3) Enumerator is used to iterate the values

4) Allows one null key and multiple null values 4) Doesn’t allow anything that is null

5) Performance is high than HashTable 5) Performance is slow

13) Difference between HashSet and TreeSet.

Answer:

HashSet TreeSet

1) Inserted elements are in random order Maintains the elements in the sorted order

2) Can able to store null objects Couldn’t store null objects

3) Performance is fast Performance is slow

14) Difference between Abstract class and Interface.

Answer:

Abstract Class:

- Abstract classes have a default constructor and it is called whenever the concrete subclass is instantiated.

- It contains Abstract methods as well as Non-Abstract methods.

- The class which extends the Abstract class shouldn’t require the implementation of all the methods, only Abstract methods need to be implemented in the concrete sub-class.

- Abstract class contains instance variables.

Interface:

- It doesn’t have any constructor and couldn’t be instantiated.

- The abstract method alone should be declared.

- Classes that implement the interface should provide the implementation for all the methods.

- The interface contains only constants.

15) What is the disadvantage of Synchronization?

Ans:

Synchronization is not recommended to implement all the methods.

Because if one thread accesses the synchronized code then the next thread should have to wait. So it makes a slow performance on the other end.

16) What is meant by Serialization?

Answer:

Converting a file into a byte stream is known as Serialization. The objects in the file are converted to bytes for security purposes. For this, we need to implement a java.io.Serializable interface. It has no method to define.

Variables that are marked as transient will not be a part of the serialization. So we can skip the serialization for the variables in the file by using a transient keyword.

17) What is the purpose of a transient variable?

Answer:

Transient variables are not part of the serialization process. During deserialization, the values of the transient variables are set to the default value. It is not used with static variables.

18) Which methods are used during the Serialization and Deserialization process?

Answer:

ObjectOutputStream and ObjectInputStream classes are higher level java.io. package. We will use them with lower level classes FileOutputStream and FileInputStream.

ObjectOutputStream.writeObject —->Serialize the object and write the serialized object to a file.

ObjectInputStream.readObject —> Reads the file and deserializes the object.

To be serialized, an object must implement the serializable interface. If superclass implements Serializable, then the subclass will automatically be serializable.

19) Difference between Serialization and Deserialization in Java.

Answer:

Serialization

1) Serialization is the process which is used to convert the objects into byte stream from the byte stream.

2) An object is serialized by writing it an ObjectOutputStream.

Deserialization

1) Deserialization is the opposite process of serialization where we can get the objects back

2) An object is deserialized by reading it from an ObjectInputStream.