1. What is private access specifier?

Private access specifier allows a class to hide its member variables and member functions from other functions and objects. Only functions of the same class can access its private members. Even an instance of a class cannot access its private members.

2. what are getter and setter methods? why do we need them?

Getter and Setter are methods used to protect your data and make your code more secure. Getter returns the value (accessors), it returns the value of data type int, String, double, float, etc. For the program's convenience, getter starts with the word "get" followed by the variable name.
While Setter sets or updates the value (mutators). It sets the value for any variable used in a class's programs. and starts with the word "set" followed by the variable name. Getter and Setter make the programmer convenient in setting and getting the value for a particular data type. In both getter and setter, the first letter of the variable should be capital.

3. why this keyword in the setter method??

While Setter sets or updates the value (mutators). It sets the value for any variable used in a class's programs. and starts with the word "set" followed by the variable name. Getter and Setter make the programmer convenient in setting and getting the value for a particular data type.

4. difference between localvariable and member variable/instance variable.

Instance Variable: These variables are declared within a class but outside a method, constructor, or block and always get a default value.

- These variables are usually created when we create an object and are destroyed when the object is destroyed.
- We may use an access specifier, for instance, variable, and if no access specifier is specified, then the default access specifier is used.
- Each and every object will have its own copy of instance variables.

Local Variable: These variables are declared within a method but do not get any default value.

- They are usually created when we enter a method or constructor and are destroyed after exiting the block or when the call returns from the method.
- Its scope is generally limited to a method and its scope starts from the line they are declared. Their scope usually remains there until the closing curly brace of the method comes.
- The initialization of the local variable is mandatory.

5. what is reference varialble?

A reference variable is a variable that points to an object of a given class, letting you access the value of an object. An object is a compound data structure that holds values that you can manipulate. A reference variable does not store its own values.

6. syntax of creating an object?

The object is a basic building block of an OOPs language. In Java, we cannot execute any program without creating an object. There is various way to create an object in Java that we will discuss in this section, and also learn how to create an object in Java.
Java provides five ways to create an object.

o Using new Keyword

o Using clone() method

o Using newInstance() method of the Class class

o Using newInstance() method of the Constructor class

o Using Deserialization

7. explain in detail what happens when we create an object?

All objects in Java programs are created on heap memory. An object is created based on its class. class as a blueprint, template, or a description how to create an object. When an object is created, memory is allocated to hold the object properties. An object reference pointing to that memory location is also created. To use the object in the future, that object reference has to be stored as a local variable or as an object member variable.

The Java Virtual Machine (JVM) keeps track of the usage of object references. If there are no more reference to the object, the object can not be used any more and becomes garbage. After a while the heap memory will be full of unused objects. The JVM collects those garbage objects and frees the memory they allocated, so the memory can be reused again when a new object is created.

8. what is class?

A class in the context of Java is a template used to create objects and to define object data types and methods.
Classes are categories, and objects are items within each category.
All class objects should have the basic class properties.
Core properties include the actual attributes/values and methods that may be used by the object.

9. what is object?

Objects are required in OOPs because they can be created to call a non-static function which are not present inside the Main Method but present inside the Class and also provide the name to the space which is being used to store the data.

10. what are the default values of all the datatypes?

Byte 0

Short 0

Int 0

Long 0

Float 0.0

Double 0.0

Boolean false

Char \u0000’ or null

11. difference between the static methods and instance
method?

Instance Method:
Instance method are methods which require an object of its class to be created before it can be called. To invoke a instance method, we have to create an Object of the class in which the method is defined.
Memory allocation: These methods themselves are stored in Permanent Generation space of heap but the parameters (arguments passed to them) and their local variables and the value to be returned are allocated in stack. They can be called within the same class in which they reside or from the different classes defined either in the same package or other packages depend on the access type provided to the desired instance method.:
• Instance method(s) belong to the Object of the class, not to the class i.e. they can be called after creating the Object of the class.
• Instance methods are not stored on a per-instance basis, even with virtual methods. They’re stored in a single memory location, and

they only “know” which object they belong to because this pointer
is passed when you call them.

- They can be overridden since they are resolved using dynamic binding at run time.

Below is the implementation of accessing the instance method:

Static Method:

Static methods are the methods in Java that can be called without creating an object of class. They are referenced by the class name itself or reference to the Object of that class.

Memory Allocation:

They are stored in the Permanent Generation space of heap as they are associated with the class in which they reside not to the objects of that class. But their local variables and the passed argument(s) to them are stored in the stack. Since they belong to the class, so they can be called to without creating the object of the class.

- Static method(s) are associated with the class in which they reside i.e. they are called without creating an instance of the class i.e ClassName.methodName(args).
- They are designed with the aim to be shared among all objects created from the same class.
- Static methods can not be overridden, since they are resolved using static binding by the compiler at compile time. However, we can have the same name methods declared static in both superclass and subclass, but it will be called Method Hiding as the derived class method will hide the base class method.

12. Syntax of accessing the member variable in the main?

sqlcmd is what you call an instance variable. Because the variable that you have declared in main method is a local variable and accessible to main method only. For accessing variable between methods declare it as an instance varibale, which is accessible by all class members. main.

13. Syntax of instance method defination?

We use an object and dot (.) operator to execute the block of code or action defined in the instance method.

14. syntax of static method defination?

Access_modifier static void methodName() { // Method body. } The name of the class can be used to invoke or access static methods.

15. difference between actual parameter and formal parameter?

The key difference between Actual Parameters and Formal Parameters is that Actual Parameters are the values that are passed to the function when it is invoked while Formal Parameters are the variables defined by the function that receives values when the function is called.

16. why we need the parameter or arguments to the methods?

Argument

An argument is a value passed to a function when the function is called. Whenever any function is called during the execution of the program there are some values passed with the function. These values are called arguments. An argument when passed with a function replaces with those variables which were used during the function definition and the function is then executed with these values. Let’s look at some examples for easy understanding.

Parameter

A parameter is a variable used to define a particular value during a function definition. Whenever we define a function we introduce our compiler with some variables that are being used in the running of that function. These variables are often termed as Parameters. The parameters and arguments mostly have the same value but theoretically, are different from each other.

17. why we need the return statement and return type to the method.

A return statement ends the execution of a function, and returns control to the calling function. Execution resumes in the calling function at the point immediately following the call. A return statement can return a value to the calling function.

18. Method can be private.(true or false)

True