1. What is private access specifier?
---> The private modifier specifies that the member can only be accessed in its own class.
The access specifiers are used to define the access restriction on the class and members of a class.
The private access modifier is the most restrictive access level. Class and interfaces cannot be private.
Members that are declared private can be accessed outside the class.

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??
--->
The keyword "this" is used in setter method to refer to the current object

4.difference between localvariable and member variable/instance variable.
--->
1)Local Variables
• Local variables are variables which are declared within a method .
• They wont get any default value and they must be initialized.
2)INSTANCE VARIABLE/MEMBER VARIABLE
• Instance variables are variables which are declared inside the class but outside the method
• Instance variables are part of object
• Instance variables always get default value.

5.what is reference varialble?
--->
Reference variable is a variable that points to the object created of a given class and allows to access the value of an object

6.syntax of creating an object?
--->
ClassName referencevariable_name; reference variable should be same a class name with first letter uppercase.
referencevariable_name=new ClassName();
Example: Student student; //creates reference variable that points to the memory location
student=new student() ;//creates the object of the class (allocates the memory for the class properties).

7.explain in detail what happens when we create an object?
--->
When we create an object ,memory is alloacted to object to hold the properties of the object and also the reference is created which points to the that memory location.

8.what is class?
--->
Class is the blueprint of an object and it is a logical entity

9.what is object?

--->
Object is an instance of a class and it is real time entity.

10.what are the default values of all the datatypes?
--->
Data type Default value :
• boolean false
• char ’\u0000’
• byte 0
• int 0
• short 0
• float 0.0f
• long 0L
• double 0.0d
• String null

11.difference between the static methods and instance method?
--->
Static methods: static methods does not need object creation to be called. they can be called with class name or by the method name itself. Instance methods: Instance methods are the moethods which are not declared as static. they can be called only with the help of object.

12.Syntax of accessing the member variable in the main?
--->
object.variablename; Example: student.studentName;

13.Syntax of instance method defination?
--->
Access _specifier return_ type methodName() { method body; }
• * Access specifier -- specifies the scope of the method that is who can access this method.
• * return_type --returns a value of specified data type from the method.
• * methodName--should be given as per the purpose and should following method naming conventions.
* methody body-- set of statements performing particular task can be given in the method body
object.variablename; Example: student.studentName;

14.syntax of static method defination?
--->
Access _modifier static return_type methodName() { method body; }
• Access specifier -- specifies the scope of the method that is who can access this method.
• static--its is non access modifier and makes the method to be accessed without creating object.
• return_type --returns a value of specified data type from the method.
• methodName--should be given as per the purpose and should following method naming conventions.
• methody body-- set of statements performing particular task can be given in the method body

15.difference between actual parameter and formal parameter?
--->
• * Actual parameters are those parameters that are specified in the calling function.
• *Formal parameters are those parameters that are declared in the called function.

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

• We need parameters to the methods to an input to the method.
• It is necessary to pass the data to methods that are working with data

17.why we need the return statement and return type to the method.
--->
• Return statement used to return the value from a method and the flow of program execution comes out of it goes back to the caller method.
• Return type returns a value of expected data type from the method

18.Method can be private.(true or false)
--->
True.

19.what is the error message if we access private variable or method out side the class?
--->
The filed Class.variable is not visible; Example:The field Student rollNo is not visible;