1. **Classes and objects:-** Here class is the template of an object and an objects is a real-world entity where all the data members and functions are stored. Until the objects are not created the class does not occupy the space in the heap. Creating a class also comes under encapsulation.
2. **Access modifiers:-**There are 4 access modifiers public, private, protected, and default. we have to declare a class as public or abstract. And we can declare data members or variables of a class as public, private, protected, and default. In the default modifier, you can access the data members outside the package. In the method inside a class, you can specify public, private, and protected. Private data members and methods are accessible within the same class.
3. **Static keyword:-**It is a keyword that is independent of objects Its allocates memory when a class is created.
4. **Abstraction:-**It is a very interesting topic. It is used to apply the common method in inheriting classes. And you cannot create an abstract class object in the main function. But you can create a reference of abstract class and achieve polymorphism.
5. **Constructor**:- Constructor is a function that has the same name as a class name and it doesn’t have any return types. There are two types of constructors one is default and one is parameterized.
6. **Constructor overloading:**-In Constructor Overloading there are several constructors which have different parameters It may of different data types. This overloading is used for assigning different data members of the same class. Access modifiers of the constructors are public, private, and protected.
7. **Java Interface:**-It is purely abstract and needs to be implemented in a class. In Interface, we use the **implement** keyword to inherit an Interface to another class like **Extends.** The implemented class must contain all incomplete methods of an interface. We can achieve multiple inheritance in interface but not in class. Variable inside interface is final. Cant be changed.
8. **Variable-length Argument list:-** It’s like an array we can pass multiple arguments in methods through a single variable.
9. **Encapsulation:-**Wrapping up methods and class variables into a single unit and can access through a variable which is known as objects.
10. **Inheritence Is-A:-**Its just like a child-parent relationship. like BMW Is-A car here BMW is a child class and car is apparent class.
11. **Inheritence Has-A:-**A class which has an object of another class. Like Engine has a class and BMW has-A an engine.
12. **Method Overriding;-**When parent class and child class both have the same function and call the function through polymorphism or normal then method overriding can be achieved.
13. **Method Overloading:-**A class having some functions and the function having the same name and different parameters then method overloading can be achieved.