Abstract class:

1. If a class contains abstract method, then the class has to be abstract
2. The abstract class object cannot be instantiated
3. A class can be made abstract if you don’t want to allow to create its object by adding abstract keyword.
4. A abstract class can have constructor, because child class can call those constructors by using super keyword
5. Usually, abstract classes are used to represent ISA relationship

Interfaces

1. All the methods in interfaces are by default public abstract
2. In interface static methods can be added, but you need to add its implementation(code)
3. In interface method can be implemented. But then the default keyword has to be used
4. All variable in interface is public static final, so cannot be changed
5. One interface can extend any number of interfaces, one class can extend only one class but can implement many interfaces.

Final classes cannot be in extended or inherited

If a method is final, it cannot be overridden

Has a relationship

1. There are 2 types of HAS-A relationship
   1. Containment relationship
   2. Association relationship
2. Containment relationship represents tight coupling, association relationship represents loose coupling
3. For Containment relation use nested classes. For association relation use Separate classes.
4. Nested class can be normal nested class, or it can be static nested class
5. If the nested class is non static.
   1. Then without creating object of outer class you cannot create an object of nested class.
   2. Nested class can access all members of outer class
6. If the nested class is static.
   1. Then without creating object of outer class you can create an object of nested class.
   2. Nested class can access only static members of parent class.