**Nested Classes**



**Static nested class**

* A static nested class isn’t usually referred to as an inner class, because it isn’t associated with an object of the outer class
* Could exist on its own
* Can be defined with any access level, but its accessibility depends on accessibility of Outer class
* Can access only static members of outer class
* Can define static and instance objects
* To access its static members, you don’t have to create an object
* To access its instance members, you must create an object
* Can define constructors
* Outer class can access its members

**Inner class**

* Needs instance of Outer class in order to be accessed
* It can access all members of outer class, even private
* It cannot define static methods, only static final variables
* Can be defined with any access level and can create members with any access level

It can be created at:

- Within an outer class, as an instance member  
- Within a method of an outer class  
- Within a static method of an outer class  
- Outside the outer class

**Anonymous inner class**

* You can use an anonymous inner class to return a value from a method
* Can be used to override methods of some class
* Cannot define other methods than ones defined in class or interface it extends
* When defined within the method it can access only its final members

**Local Inner Class**

* A nested class within a method
* Two same name classes can be defined in two different methods of an outer class
* Cannot have access modifiers
* Can have its own constructors, methods and variables. Static variables can only be final variables (methods are not allowed)
* Instance can be created only within a method in which the inner class is defined
* Can access all variables of the method