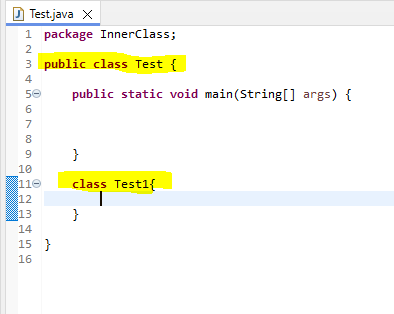
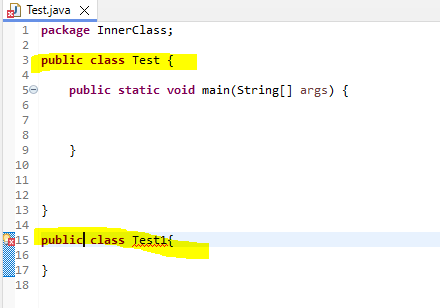
**Inner Class**

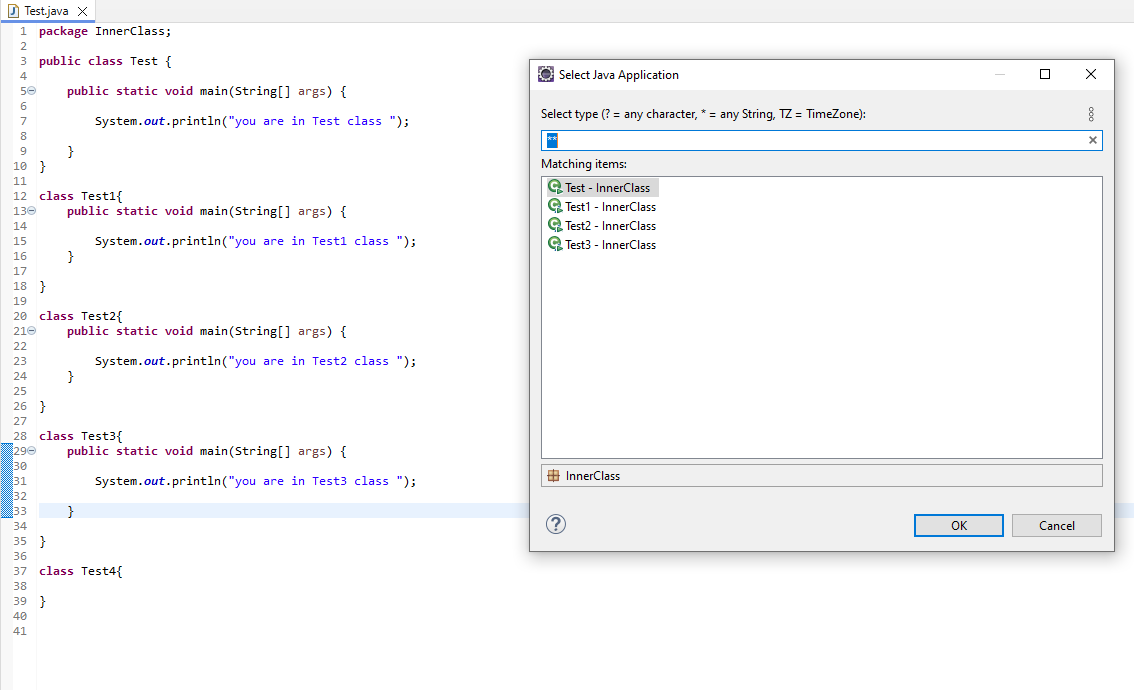
* We can have more than one class in a java file.



* We can't have more than one public class in java file. Because when we compile that file name should be same as class name. If we keep multiple public classes in one java file jvm will get confuse to which name it should save.



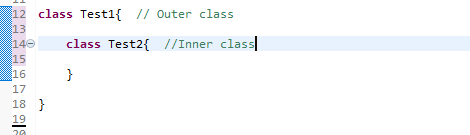
* In java file if we have 3 classes when we execute in bin folder those many class files will generate.
* Egg: we created 4 classes in single java file after execution in bin folder we can see 4 class files.
* In java file we can have multiple classes and inside class we can have main methods also. But while execution time it will ask which class you what to execute.



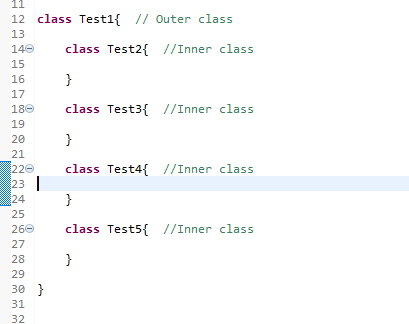
* Inside java file we have multiple classes but we won't call them as inner class.

There are 4 types on inner classes

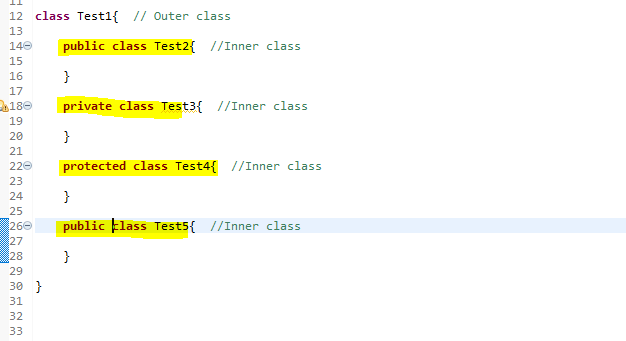
* Nested Non-static inner classes
* Nested Static inner classes
* Inside one class if we write one more class then it is called inner class.



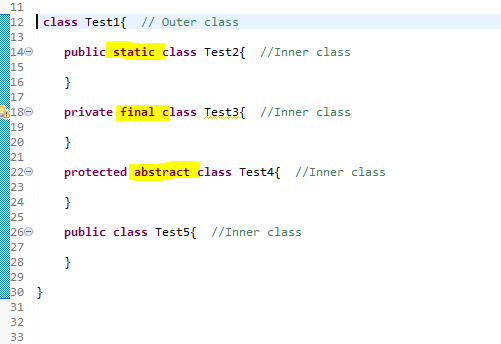
* Inside class we can have multiple inner classes. But all class names should be unique.



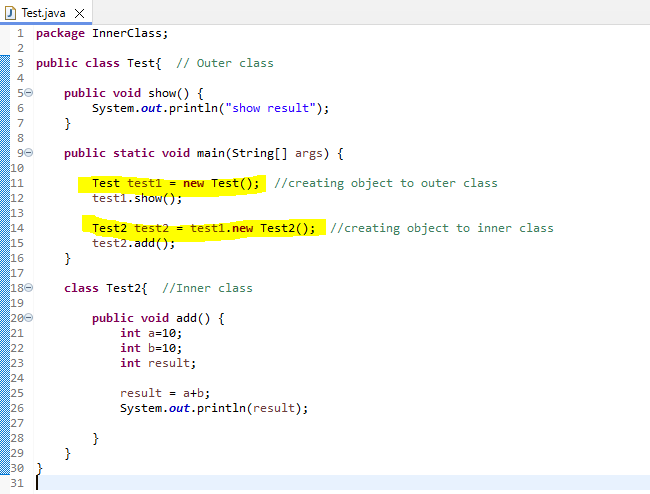
* after executing program in backend outer class will generate with direct class names but inner class will generate with outer class name $ inner class names
* E.g.: TestClass1$TestClass4.class
* We can create private, public, protected inner classes. If we won't create by default it creates as default access modifier.



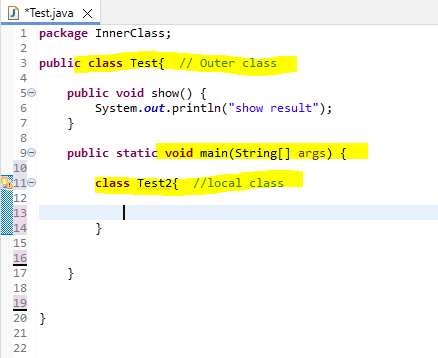
* We can create inner classes with static, final, abstract keywords also.



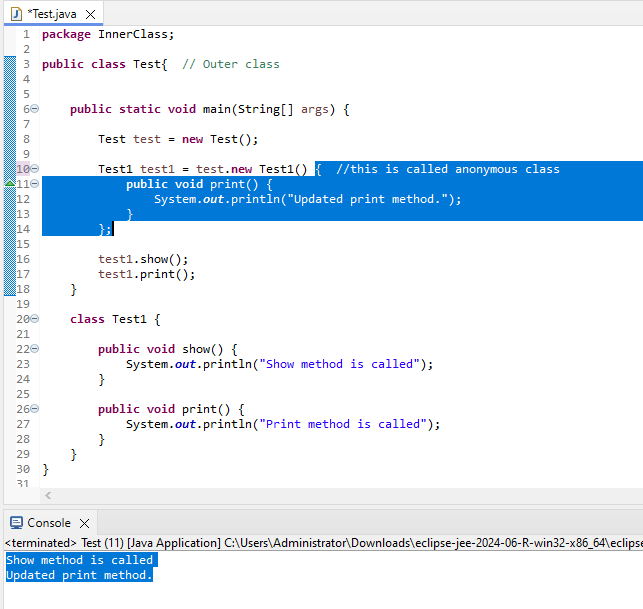
* We can't write main method in inner classes.
* Inside inner class directly we can create object. But when we creating inner class object at outer class we need to follow syntax.
* TestCase1 test1 = new TestCase1 (); --> outer class object
* TestCase2 test2 = test1.new TestCase2 (); -->Inner class object creation.



* Both abstract class and non-abstract class we can extend in inner classes. And we can access the methods and variables.
* In inner class we can implement interface also.
* We can extend one inner with another inner class.
* Local classes
* However we are accessing local variables in the same way we are accessing local classes also.
* We can access them only in particular block of code. We can’t access them outside the block.
* We can create then in constructor, in a block, in a method.
* If we want to use a class particularly in this method then we use local class concept.



* For local class we can’t use public, private, protected and static.
* Only final and abstract keywords we can use.
* We can create N number of local class.
* For local class can access entire class variables but outer class can’t access local class.
* Anonymous inner classes/anonymous classes



* Here we are not showing class name.
* In backed it save like Test$1.class