Inner Classes

Inner classes are non-static classes, declared on an enclosing class, at the member level.

Inner classes can have any of the four valid access modifiers.

An inner class has access to instance members, including private members, of the enclosing class.

Instantiating an inner class from external code, is a bit tricky, and I'll cover that shortly.

As of JDK16, static members of all types are supported on inner classes.

Inner Classes

To create an instance of an inner class, you first must have an instance of the Enclosing Class.

From that instance you call .new, followed by the inner class name and the parentheses, taking any constructor arguments.

This definitely looks strange the first time you see it.

```
EnclosingClass outerClass = new EnclosingClass();
EnclosingClass.InnerClass innerClass = outerClass.new InnerClass();
```