Anonymous Classes

An anonymous class is a local class that doesn't have a name.

All the nested classes we've looked at so far have been created with a class declaration.

The anonymous class is never created with a class declaration, but it's always instantiated as part of an expression.

Anonymous classes are used a lot less, since the introduction of Lambda Expressions in JDK 8.

But there are still some use cases where an anonymous class might be a good solution.

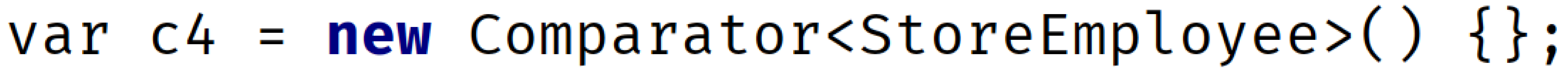
Anonymous class creation

An anonymous class is instantiated and assigned in a single statement.

The new keyword is used followed by any type.

This is **NOT** the type of the class being instantiated.

It's the super class of the anonymous class, or it's the interface this anonymous class will implement as I'm showing here.



```
var c4 = new Comparator<StoreEmployee>() {};
```

Anonymous class creation

In the first example on this slide, the anonymous unnamed class will implement the Comparator interface.

```
var c4 = new Comparator<StoreEmployee>() {};
```

In the second example on this slide, the anonymous class extends the Employee class, meaning it's a subclass of Employee.

```
var e1 = new Employee {};
```

In both cases, it's important to remember the semi-colon after the closing bracket, because this is an expression, not a declaration.