```
(x
+ y)
```

or
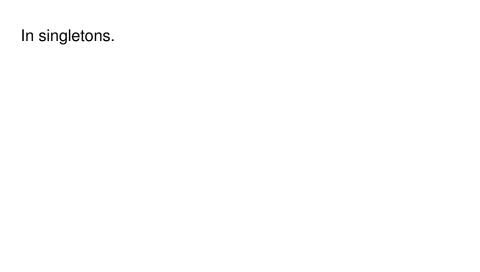
```
(x +
y)
```

The end of the line is treaded as a semicolon unless
- the line ends with a word that it's a legal end of a statement, like a period or infix operator.
- the next line starts with a word that can't start a statement.
- the line ends while in `(...)` or `[...]`.

In singletons.

- It allows no access to command line arguments.
- Treading limitations make it bad for multithreaded programs.
- HotSpot doesn't optimize an object executed by the `Application` trait.

Start and compile: `fsc`
Stop: `fsc -shutdown`

Each other's private members.

```
scala._
scala.Predef._
java.lang._
```

It cannot define a new Scala type, but it can still be an instance of a superclass, trait, or companion class.
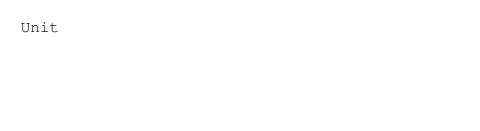
It's called a standalone object.

If it mutates a state outside a method or performs i/o.

It's a singleton object with the same name as the class it shares a source file with.

When it ends in a result expression.

They are `val`s.

Statics.

Unit

Public. Achieved by leaving off the visibility modifier.

A singleton object with `main` method that takes `Array[String]`.

The result type is irrelevant.

scala.collection.jcl.WeakHashMap

The result type automatically becomes `Unit`.