

Dark Syntax Sugar



So Sweet!

Single arg method calls allow {}

Single abstract method

Methods with ':' are special

- right-associative if ending in ':'
- left-associative otherwise

```
val result = singleParamMethod {  
  // ...  
  42  
}
```

```
trait Action {  
  def act(x: Int): Unit  
}
```

```
val action: Action = (x: Int) => println(x)
```

```
class MyStream[T] {  
  def -->:(value: T): MyStream[T] = ???  
}
```

```
val stream = 1 -->: 2 -->: new MyStream[Int]
```

So Sweet!

Method naming, level++

```
class Teen {  
  def `and then said`(message: String) = ???  
}  
  
mike `and then said` "funny name!"
```

Infix generic types

```
trait <[A, B]  
val lessThan: A < B = ???
```

Update()

```
val anArray = Array(1,2,3)  
anArray(2) = 7 // <=> anArray.update(2, 7)
```

Setters for mutable types

```
trait MutableContainer {  
  def member: Int  
  def member_=(value: Int): Unit  
}
```

```
mContainer.member = 42
```

Scala rocks

