

A Featurette: Scala Features

In the following lesson, you will be introduced to some of Scala's main features.

We'll cover the following

- 1. Type Inference
- 2. Immutability
- 3. Lazy Computation
- 4. Case Classes & Pattern Matching
- 5. String Interpolation
- 6. Higher-Order Functions
- 7. Traits
- 8. Extensive Collection



1. Type Inference

Type inference is Scala's ability to infer types without them being explicitly mentioned. The language is *smart* enough to deduce data types based on the context in which the object has been written.

2. Immutability

Immutability is the quality of being unchangeable. In Scala, variables are immutable by default; once assigned a value they cannot be modified by you or any other programmer. This might not be apparent now, but default immutability

is one of Scala's strongest features. It is primarily used for concurrency control

which manages concurrent (simultaneous) operations so that they do not conflict with each other.

You can also create immutable variables by explicitly mentioning it in your code.

3. Lazy Computation

Scala only evaluates an expression when required. This increases performance by reducing compile time.

4. Case Classes & Pattern Matching

In Scala, case classes are regular classes with the added feature of being immutable. This makes them great for modeling immutable data.

Case classes are useful for pattern matching, which is used for checking a value against a pattern and deconstructing it into its constituent parts.

5. String Interpolation

Scala's string interpolation methods allow you to embed variables directly inside a string literal allowing the creation of strings through data.

6. Higher-Order Functions

Higher-order functions are functions which take other functions as arguments. They can also return functions along with simple data types.

7. Traits

Traits are a collection of abstract and non-abstract methods. They are similar to Java's interface. Simply put, while classes and objects can only inherit from a single class, those same classes and objects can inherit from multiple traits.

8. Extensive Collection

Scala provides an extensive collection of classes and traits which are used for collecting data. They are divided into two broad categories of mutable and immutable collections and are all contained in Scala's package `Scala.collection`.

In the next lesson, we will look at some of the big names in the industry who are using Scala.