

Features of Scala

There are following features of scala:

Type inference

Singleton object

Immutability

Lazy computation

Case classes and Pattern matching

Concurrency control

String interpolation

Higher order function

Traits

Rich collection set

Type Inference

In Scala, you don't require to mention data type and function return type explicitly. Scala is enough smart to deduce the type of data. The return type of function is determined by the type of last expression present in the function.

Singleton object

In Scala, there are no static variables or methods. Scala uses singleton object, which is essentially class with only one object in the source file. Singleton object is declared by using object instead of class keyword.

Immutability

Scala uses immutability concept. Each declared variable is immutable by default. Immutable means you can't modify its value. You can also create mutable variables which can be changed.

Immutable data helps to manage concurrency control which requires managing data.

Lazy Computation

In Scala, computation is lazy by default. Scala evaluates expressions only when they are required. You can declare a lazy variable by using lazy keyword. It is used to increase performance.

Case classes and Pattern matching

Scala case classes are just regular classes which are immutable by default and decomposable through pattern matching.

All the parameters listed in the case class are public and immutable by default.

Case classes support pattern matching. So, you can write more logical code.

Concurrency control

Scala provides standard library which includes the actor model. You can write concurrency code by using actor. Scala provides one more platform and tool to deal with concurrency known as Akka. Akka is a separate open source framework that provides actor-based concurrency. Akka actors may be distributed or combined with software transactional memory.

String Interpolation

Since Scala 2.10.0, Scala offers a new mechanism to create strings from your data. It is called string interpolation. String interpolation allows users to embed variable references directly in processed string literals. Scala provides three string interpolation methods: s, f and raw.

Higher Order Functions

Higher order function is a function that either takes a function as argument or returns a

function. In other words, we can say a function which works with another function is called higher order function.

Higher order function allows you to create function composition, lambda function or anonymous function etc.

Traits

A trait is like an interface with a partial implementation. In Scala, trait is a collection of abstract and non-abstract methods. You can create trait that can have all abstract methods or some abstract and some non-abstract methods.

Traits are compiled into Java interfaces with corresponding implementation classes that hold any methods implemented in the traits.

Rich Set of Collection

Scala provides rich set of collection library. It contains classes and traits to collect data. These collections can be mutable or immutable. You can use it according to your requirement. `Scala.collection.mutable` package contains all the mutable collections. You can add, remove and update data while using this package.

`Scala.collection.immutable` package contains all the immutable collections. It does not allow you to modify data.