

Table of Contents

Scala

Hello World

3

5

Scala

- <https://www.scala-lang.org/>

Scala (<i>programminglanguage</i>)	
Full Name	Scala
Short Name	Scala
Description	general-purpose programming language providing support for functional programming and a strong static type system
Company	Unkown
Web	No
Mobile	No
Enterprise	Yes
Embedded	Yes

What is Scala?

Scala is a modern programming language that combines object-oriented and functional programming paradigms. It is designed to be concise, expressive, and type-safe, and is often used for web development, data analysis, and other applications.

What are some advantages of using Scala?

Some advantages of using Scala include its support for functional programming concepts such as immutability and higher-order functions, its ability to seamlessly integrate with Java and other JVM languages, and its strong type system and support for type inference.

What are some popular frameworks in Scala?

Some popular frameworks in Scala include Akka for concurrent and distributed applications, Play Framework for web development, and Spark for big data processing.

What are traits in Scala?

Traits in Scala are similar to interfaces in other programming languages, and define a set of methods and fields that can be mixed into a class to provide additional functionality. Traits can be used to achieve code reuse and modularity, and are a key feature of the language.

What is pattern matching in Scala?

Pattern matching in Scala is a powerful feature that allows developers to match values against a set

of patterns, and execute different code depending on the pattern that matches. It is commonly used for tasks such as parsing input data, implementing algorithms, and handling exceptions.

What is a Future in Scala?

A Future in Scala is a container object that represents a value that may not yet be available. It is commonly used in concurrent and asynchronous programming, and allows developers to write code that can execute while waiting for a potentially long-running operation to complete.

What is the Scala REPL?

The Scala REPL (Read-Eval-Print Loop) is an interactive shell that allows developers to quickly experiment with Scala code, and see the results of their code in real time. It is a useful tool for learning the language, and for quickly prototyping and testing code snippets.

Snippet from [Wikipedia](#): [Scala \(programming language\)](#)

Scala (*SKAH-lah*) is a strong statically typed high-level general-purpose programming language that supports both object-oriented programming and functional programming. Designed to be concise, many of Scala's design decisions are intended to address criticisms of Java.

Scala source code can be compiled to Java bytecode and run on a Java virtual machine (JVM). Scala can also be compiled to JavaScript to run in a browser, or directly to a native executable. On the JVM Scala provides language interoperability with Java so that libraries written in either language may be referenced directly in Scala or Java code. Like Java, Scala is object-oriented, and uses a syntax termed *curly-brace* which is similar to the language C. Since Scala 3, there is also an option to use the off-side rule (indenting) to structure blocks, and its use is advised. Martin Odersky has said that this turned out to be the most productive change introduced in Scala 3.

Unlike Java, Scala has many features of functional programming languages (like Scheme, Standard ML, and Haskell), including currying, immutability, lazy evaluation, and pattern matching. It also has an advanced type system supporting algebraic data types, covariance and contravariance, higher-order types (but not higher-rank types), anonymous types, operator overloading, optional parameters, named parameters, raw strings, and an experimental exception-only version of algebraic effects that can be seen as a more powerful version of Java's checked exceptions.

The name Scala is a portmanteau of *scalable* and *language*, signifying that it is designed to grow with the demands of its users.

[Creative Commons Attribution-Share Alike 4.0](#)

Hello World

```
object HelloWorld extends App {  
  println("Hello, world!")  
}
```

Source: [Wikibooks](#)

See also: [Programming Languages](#)

[programminglanguage](#)

From:

<https://almbok.com/> - **ALMBoK.com**

Permanent link:

<https://almbok.com/dev/scala>

Last update: **2023/03/25 18:02**

