

Scala Programlama Dili Eğitimi

Süre: 5 Gün

Açıklama

Scala Programlama Dili Eğitimi, Scala sözdizimi, nesne yönelimli ve fonksiyonel programlama konularını ele alan 5 günlük bir eğitim programıdır.

Sınıflar, metotlar, fonksiyonlar ve koleksiyonlar etkileşimli soru-cevap ve alıştırmalarla işlenir; iki programlama yaklaşımını birlikte kullanma üzerine çalışılır.

Scala, işlevsel programlama topluluğunda uzun süredir kullanılan gelişmiş teknikleri nesneye dayalı en iyi uygulamalarla bir araya getirerek, geliştiricilerin ekiplerine ve karşılaştıkları sorunlara en iyi şekilde uyan tarz ve yaklaşımı seçmelerine olanak tanır.

Katılımcılar bu eğitimin sonunda, nesneye dayalı ve işlevsel programlamanın yanı sıra özellikle bu programlama tarzlarının nasıl kullanılacağı ve birleştirileceği ile ilgili tüm önemli özellikler hakkında bilgi sahibi olurlar.

15 modüle bölünmüş beş gün süren bu eğitim, çeşitli alıştırmalar ve katılımcıların eğitime daha iyi katılabilmelerini sağlamak üzere etkileşimli soru/cevap yaklaşımı kullanılarak verilir.

Bu eğitimde neler öğreneceksiniz?

- Değişken ve sabit veriler arasındaki farklar
- Scala tarafından kullanılan temel sözdizimi
- Scala kullanarak nesneler ve sınıflar oluşturma
- Değişken kapsamalarını belirleme ve örtülü işlev çağrıları
- Özellikler ve devralma
- İşlevsel programlama paradigması ve nesneye dayalı yaklaşımdan nasıl farklılaştığı
- “Currying” ve “Lambda” işlevleri ile ilgili kavramlar
- İşlevsel programlama ile koleksiyonları ve genel türleri kullanabilme
- Düzen eşleme hakkında bilgiler

Eğitim İçeriği

Introduction

- Why scala?
- Libraries

- Static typing
- Compositional syntax
- OO
- Functional
- Powerful
- Paradigms
- OO
- Functional
- Running scala
- Anatomy of scala programs
- Try the repl
- Scalac
- Scala interpreter
- Sbt

Fundamentals

- Language
- Objects
- Calling methods
- Operators as methods
- Values and variables
- Types
- Basic types
- Boolean
- Numeric types
- Unit
- Strings
- String methods
- Introduction to collections
- Type arguments
- Tuples, Lists, Maps

Flow

- Branching, matching & selecting
- Conditionals
- Intro to pattern matching
- Destructuring
- For comprehensions
- Yield vs Unit

- Comprehensions over lists
- Comprehensions over maps
- Comprehensions over ranges
- Option
- Multiple extraction
- Guards
- Ranges
- Let expressions
- While loops

Methods

- Code blocks
- Methods
- Method bodies
- Returning unit
- Passing arguments
- Variadics
- Def vs val
- Lazy vals
- Lazy arguments
- Recursion

Functions

- What Is Functional Programming?
- Pure Functions
- Functions
- The Function Type
- 'Function' vs Method
- Higher Order Functions
- Functions as Data
- Currying
- Aside: Type Aliases

Collections

- Review: types
- Collections heirachy
- Creating collections
- Range
- Array & array buffer

- List & list buffer
- Vector
- Maps
- Sets
- Empty
- Idioms
- Pattern matching
- For comprehensions review
- Zipping
- Traits

Transformation

- Streams
- Combinators
- .map & .flatMap
- Folding & Reducing
- .exists and .forall
- Option
- For comprehensions
- Foreach & Flatmap
- Pattern matching
- Review
- Casting
- Sequences
- For-yield-match
- Regex
- Guards
- Option

OO

- Classes
- Reading class definitions
- Defining classes
- Constructors
- Properties

Ön Koşullar

Temel Java Programlama ve OOP bilgisine sahip olmak