

Section 1

Introduction to Elixir



In this Section, we are going to take a look at...

- What is Elixir
- History and significance
- Functional programming paradigm

João Gonçalves

Video 1.2

What is Elixir?



In this video, we are going to take a look at...

- Exploring the history of Elixir
- Discussing language characteristics
- Observing notable use cases

History of Elixir



Erlang

Design Goals

Compatibility Productivity Extensibility

History of Elixir



2014

Characteristics

- General Purpose
 - Suitable for multiple application types
- Functional
 - Adheres mostly to the functional paradigm, but not only
- Actor Model
 - Processes are akin to Actors, encapsulating data (state), receiving, and sending messages

Characteristics

- Dynamically Typed
 - Types are inferred in compile and run-time
- Compiled
 - The compiler generates code that is executed by the BEAM (Erlang VM)

Use Cases



Use Cases



- General applications
- Large scale services
- Web applications



- Systems programming
- Scripting