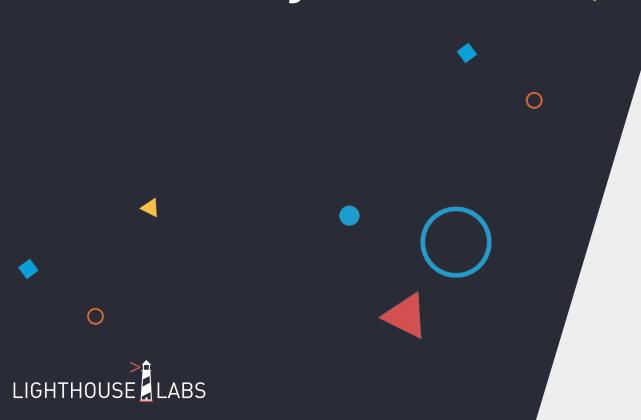
Intro To Ruby



AGENDA

Prog Paradigms

Feature Comparison

Code Demo



Programming Paradigms



What is a Programming Paradigm

- Programming paradigms are a way of organizing and describing languages based upon philosophical or ideological design decisions
- At the highest level, all programming languages fall into one of two umbrella categories
 - a. Imperative
 - b. Declarative
- In Imperative programming, our focus is in telling the computer "how" to do some work
- In Declarative programming, our focus is in telling the computer "what" a solution to a problem is,
 and allowing the computer to create potential solutions



Imperative Programming

- Includes
 - a. Procedural languages
 - b. Object Oriented languages
- i.e.
 - a. Java
 - b. Ruby
 - c. Python
 - d. C++
 - e. C#
 - f. Rust



Declarative Programming

- Includes
 - a. Functional
 - b. Logical
- i.e.
 - a. Lisp
 - b. Haskell
 - c. CSS
 - d. SQL
 - e. Prolog



Feature Comparison



Type Safety

- Type safety is the extent to which type errors are discouraged or prevented by the language
- Very few languages are strict about type safe
 - a. Strict type safety fails at the edges (libraries written in other languages for example)
 - b. Strict type safety is onerous for developers
- Type safety is largely a trade off between developer ease of use and correctness guarantees
- High level languages favour developer ease of use, and are thus more weakly typed
- Most low level languages are more strongly typed
- Most object oriented languages are somewhat strongly typed



Type Checking

- The process of verifying that the **types** of values stored in variables are consistent
- Type checking systems are one of the useful ways that we organize languages
 - a. Static Type Checking
 - Compile time type checking
 - b. Dynamic Type Checking
 - Run Time type checking
- As with Paradigms, some languages use a hybrid approach, statically type checking things that are easy to validate, while letting the remainder by dynamically checked



Some languages we know

- JavaScript is non type checked weakly typed language
- SQL is a statically type checked strongly typed language
- CSS is a statically typed checked strongly typed language
- Ruby is a dynamically type checked strongly typed language

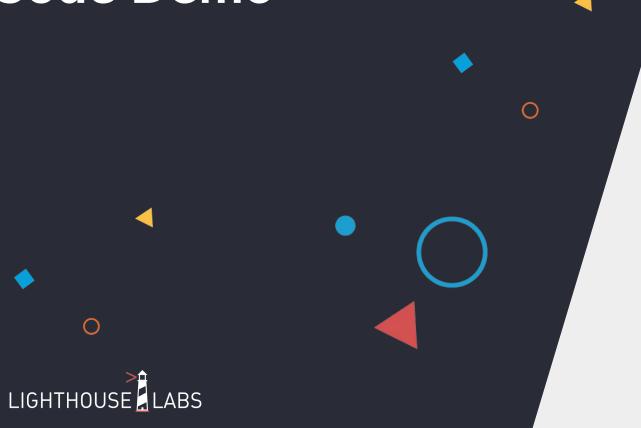


Some languages we know

- JavaScript is non type checked weakly typed language
- Ruby is a dynamically type checked strongly typed language
- JavaScript makes no effort to protect you from type errors, and generally we rarely think about types in JavaScript at all
- While Ruby looks like a lot like JavaScript, it actually cares about types, and will make an effort to show you type mistakes at compile time
 - Ruby variables are strongly typed, but can change types throughout your program
 - Little type checking is done statically in ruby



Code Demo



Questions?

