Programming Languages

Janyl Jumadinova

January 18-20, 2023

What this course is NOT about

- Learning to program proficiently in lots of different languages
  - but this course should make it easier to learn new languages AND we will focus on Java and you should become proficient in it

What this course is NOT about

- Learning to program proficiently in lots of different languages
  - but this course should make it easier to learn new languages AND we will focus on Java and you should become proficient in it
- Learning tiny bits about lots of different languages
  - but we will use examples from many languages to examine more general principles

What this course is NOT about

- Learning to program proficiently in lots of different languages
  - but this course should make it easier to learn new languages AND we will focus on Java and you should become proficient in it
- Learning tiny bits about lots of different languages
  - but we will use examples from many languages to examine more general principles
- "Language wars" ("Which is better, Python, Java or C++?")
  - but you will learn about criteria that can be used to compare different languages

What this course is about

- How are languages designed and implemented?
  - Specifying syntax and semantics, compiled vs. interpreted, etc.

What this course is about

- How are languages designed and implemented?
  - Specifying syntax and semantics, compiled vs. interpreted, etc.
- How do such choices affect ease of use, efficiency, scalability, and other criteria?
  - Example: How should recursive calls be implemented? How does "garbage collection" work?

What this course is about

- How are languages designed and implemented?
  - Specifying syntax and semantics, compiled vs. interpreted, etc.
- How do such choices affect ease of use, efficiency, scalability, and other criteria?
  - Example: How should recursive calls be implemented? How does "garbage collection" work?
- What are the different programming paradigms?
  - Example: Why would anyone ever use a language like ML rather than C++ or Java?

Why Study Programming Languages?

Why Study Programming Languages?

- Help you choose a language.

Why Study Programming Languages?

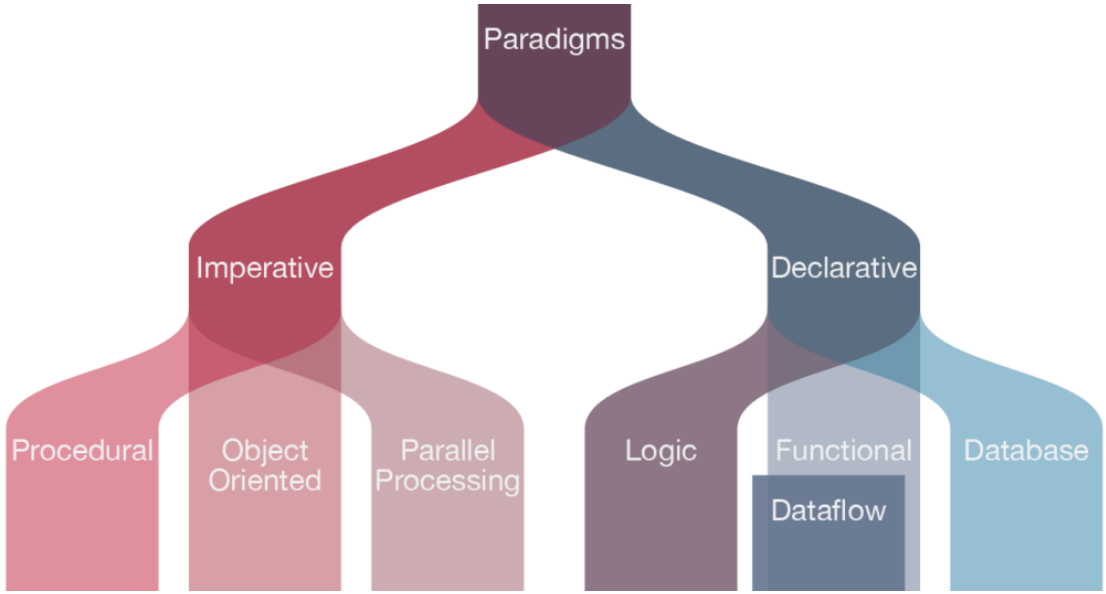
- Help you choose a language.
- Make it easier to learn new languages.

Why Study Programming Languages?

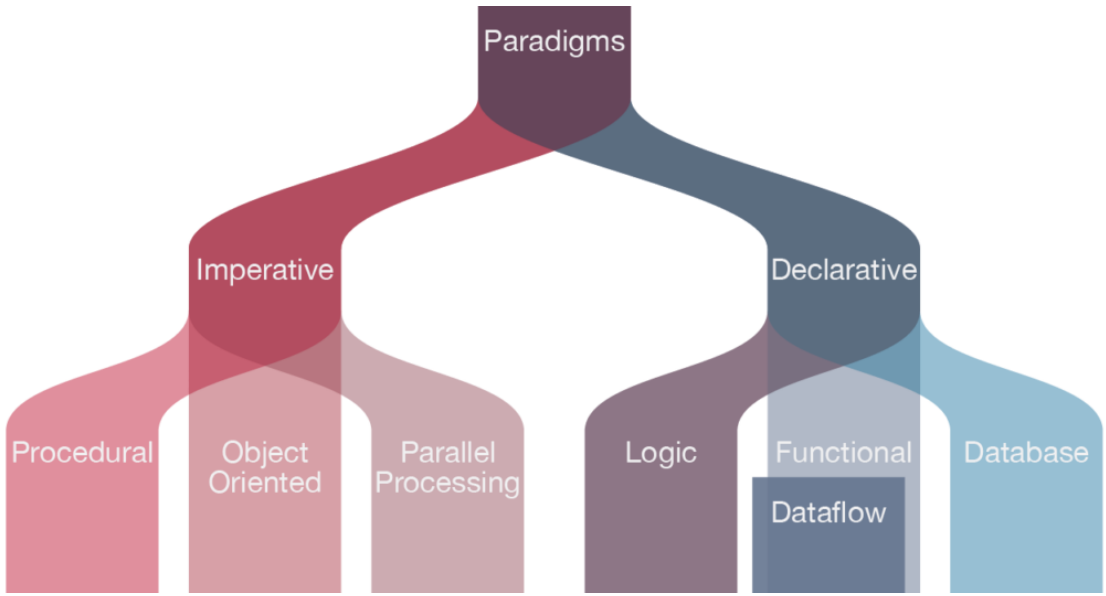
- Help you choose a language.
- Make it easier to learn new languages.
- Help you make better use of whatever language you use.

Image credit: https://pixabay.com

Programming Language Paradigms

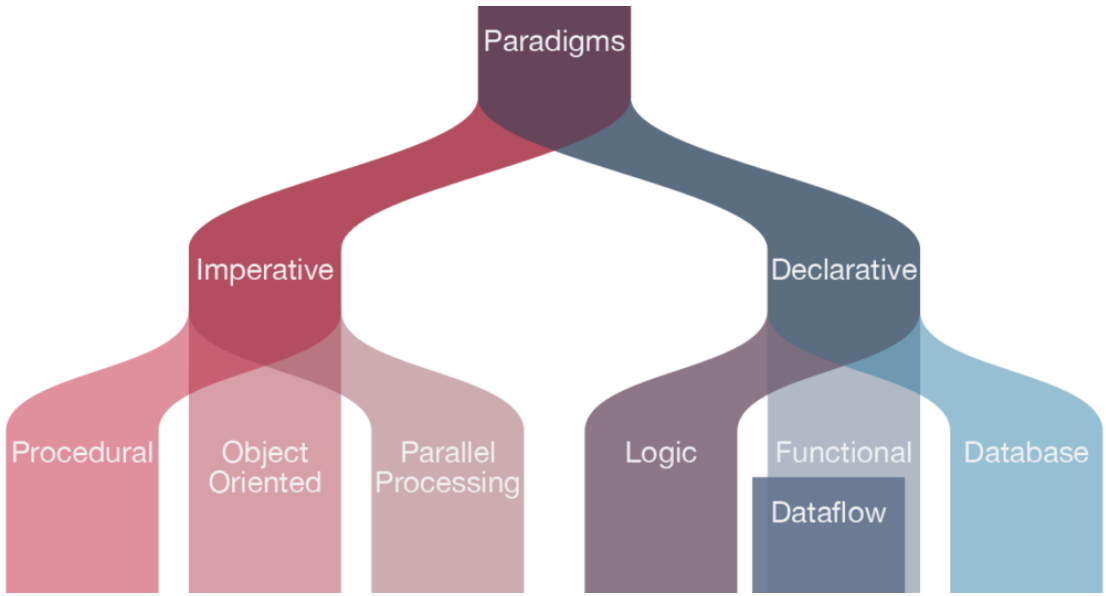


Programming Language Paradigms



declarative	
functional	Lisp/Scheme, ML, Haskell
dataflow	Id, Val
logic, constraint-based	Prolog, spreadsheets
template-based	XSLT
imperative	
von Neumann	C, Ada, Fortran, . . .
scripting	Perl, Python, PHP, . . .
object-oriented	Smalltalk, Eiffel, Java, . . .

Programming Language Paradigms



declarative	
functional	Lisp/Scheme, ML, Haskell
dataflow	Id, Val
logic, constraint-based	Prolog, spreadsheets
template-based	XSLT
imperative	
von Neumann	C, Ada, Fortran, . . .
scripting	Perl, Python, PHP, . . .
object-oriented	Smalltalk, Eiffel, Java, . . .

Image credits: https://thepafhelper.blogspot.com and PLP book

What makes a language successful?

Language Evaluation Criteria

- **Readability**: the ease with which programs can be read and understood

Language Evaluation Criteria

- **Readability**: the ease with which programs can be read and understood
- **Writability**: the ease with which a language can be used to create programs

Language Evaluation Criteria

- **Readability**: the ease with which programs can be read and understood
- **Writability**: the ease with which a language can be used to create programs
- **Reliability**: conformance to specifications

Language Evaluation Criteria

- **Readability**: the ease with which programs can be read and understood
- **Writability**: the ease with which a language can be used to create programs
- **Reliability**: conformance to specifications
- **Cost**: the ultimate total cost

Art of Programming Language Design Activity

"...design [is] the intentional creation of plans for a new kind of thing". Parsons

Art of Programming Language Design Activity

"...design [is] the intentional creation of plans for a new kind of thing". Parsons

- Choose one programming language
- Working in a group, investigate the following:
  1. Key characteristics of the language **design**.
  2. Historical context (how it started, when, by whom, the first language, etc.).
  3. The purpose/usage (why is it there).
- Prepare to share your findings.

Image credit: https://pixabay.com

Why So Many Programming Languages?

Why So Many Programming Languages?

- Evolution

Why So Many Programming Languages?

- Evolution
- Socio-economic factors

Why So Many Programming Languages?

- Evolution
- Socio-economic factors
- Special purposes

Why So Many Programming Languages?

- Evolution
- Socio-economic factors
- Special purposes
- Special hardware

Why So Many Programming Languages?

- Evolution
- Socio-economic factors
- Special purposes
- Special hardware
- Range of ideas