Functional Programming

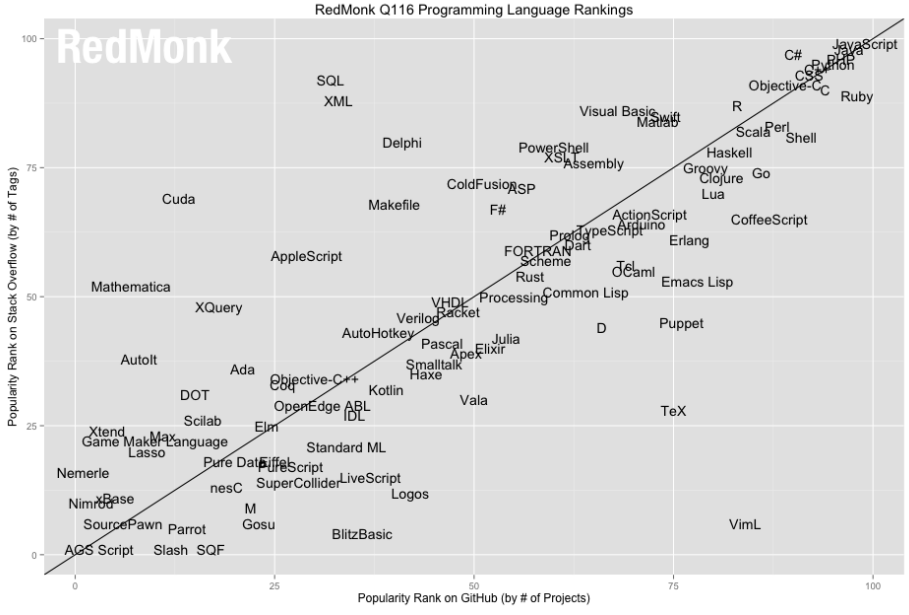
in Racket

Peter Campora

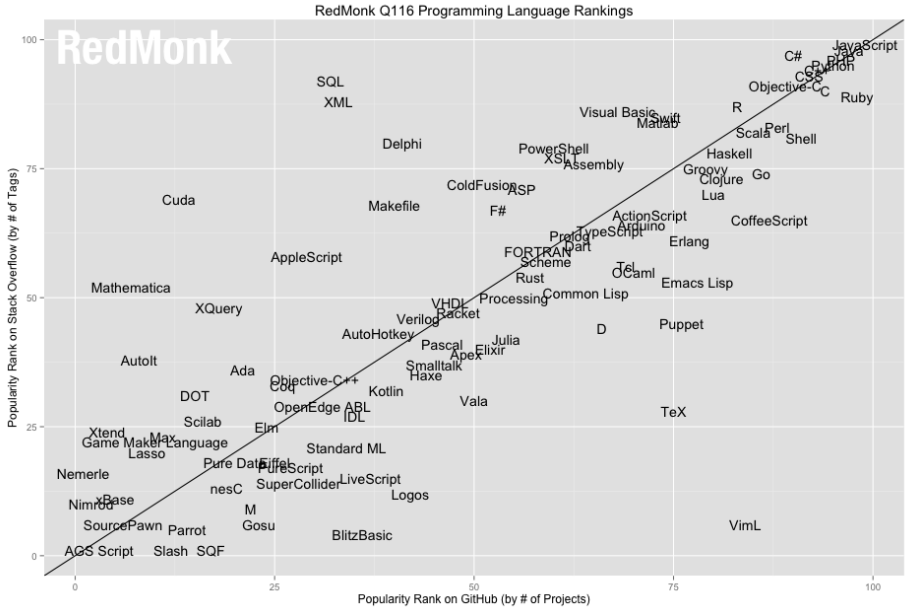
ULL

August 25, 2019

Programming Language Hot Takes

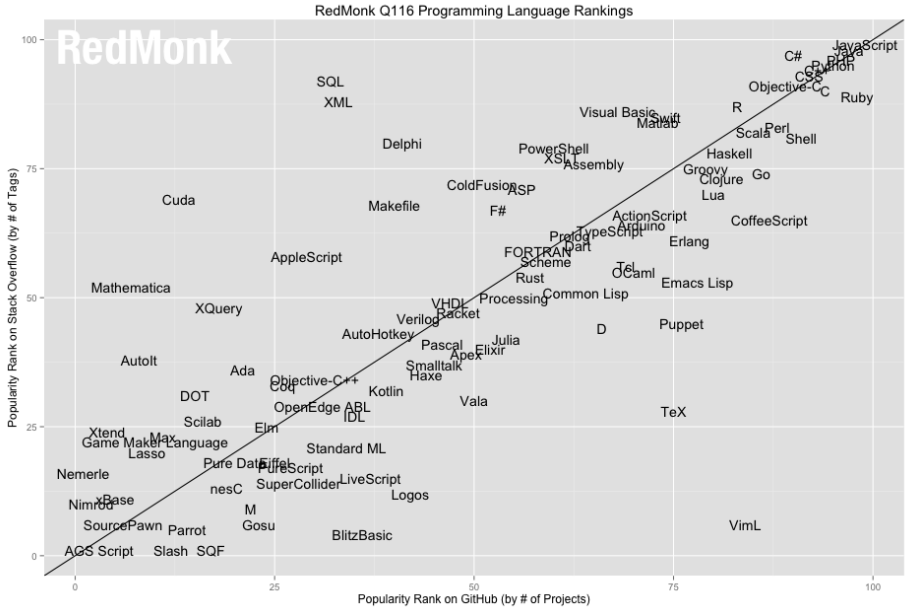


Programming Language Hot Takes



There are numerous programming languages and even more opinions about them.

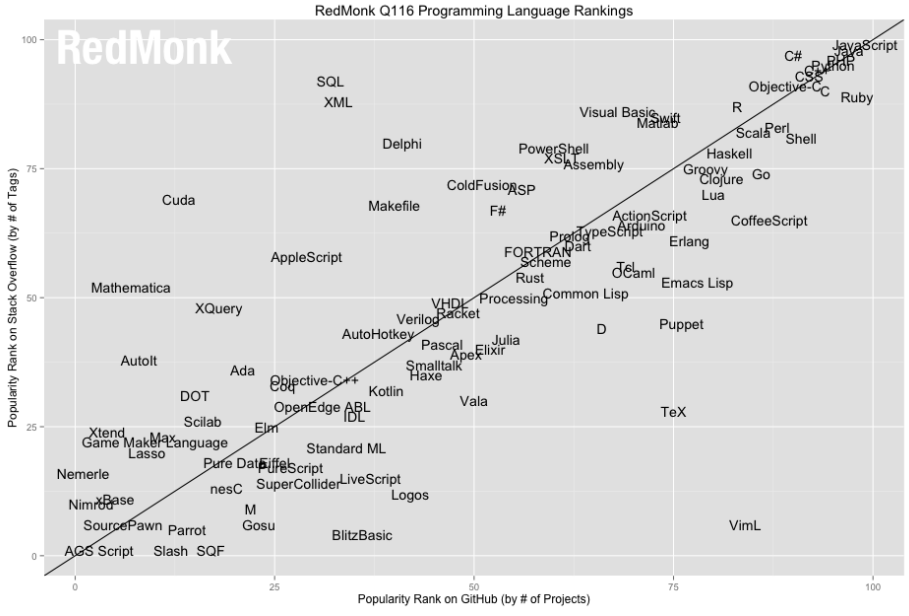
Programming Language Hot Takes



There are numerous programming languages and even more opinions about them.

So, why so many languages and how do they differ?

Programming Language Hot Takes

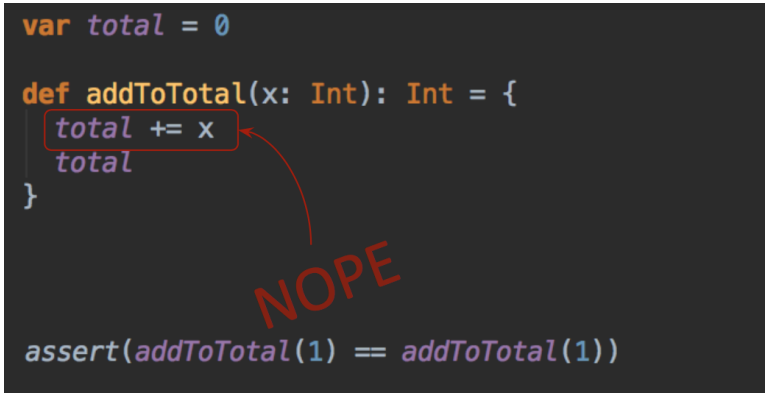


There are numerous programming languages and even more opinions about them.

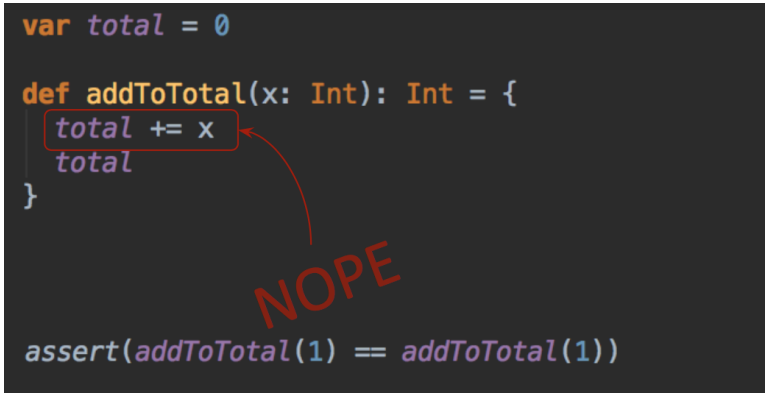
So, why so many languages and how do they differ?

And why is *functional programming* becoming such a buzzword?

Why Functional Programming?

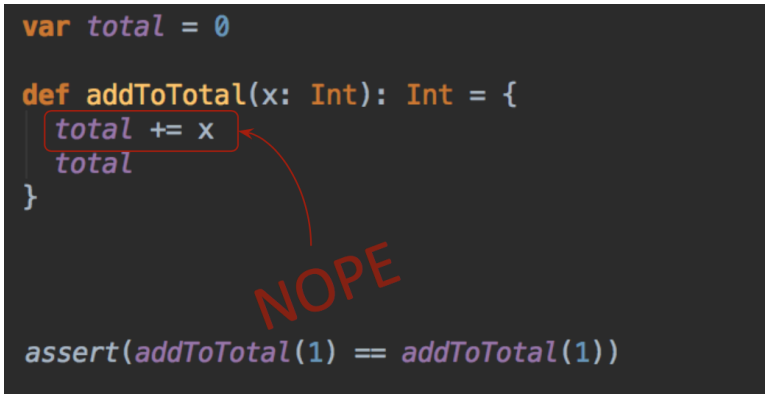


Why Functional Programming?



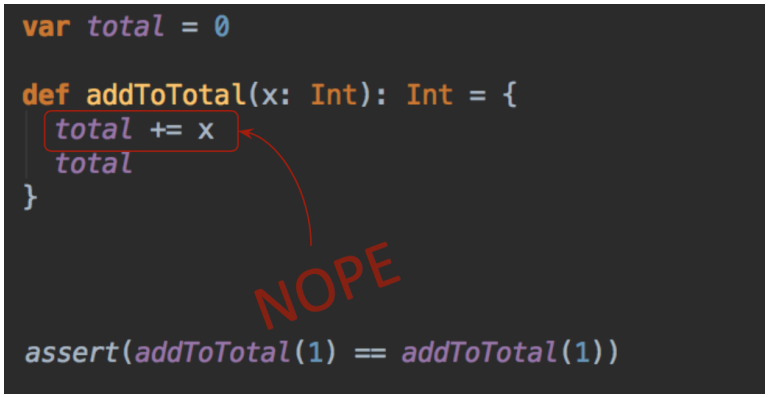
- The function above isn't *referentially transparent*

Why Functional Programming?



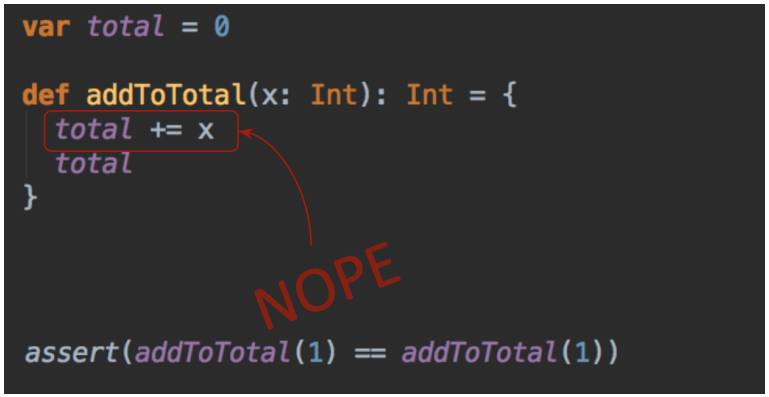
- The function above isn't *referentially transparent*
- Which means it really isn't even a function in the *mathematical* sense.

Why Functional Programming?



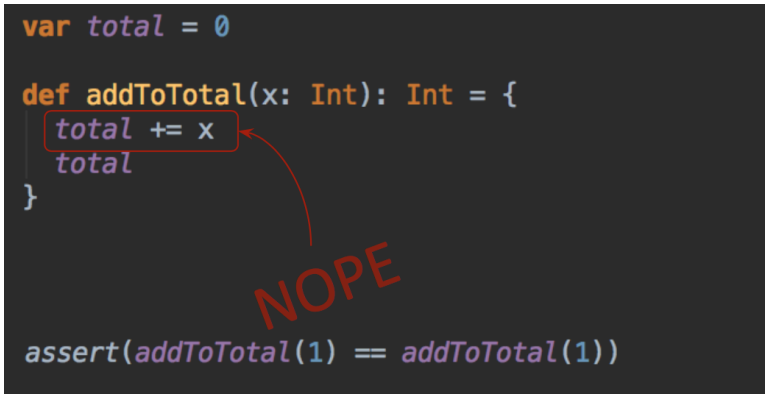
- The function above isn't *referentially transparent*
- Which means it really isn't even a function in the *mathematical* sense.
- This makes it hard to create reproducible behavior or prove code correctness

Why Functional Programming?



- The function above isn't *referentially transparent*
- Which means it really isn't even a function in the *mathematical* sense.
- This makes it hard to create reproducible behavior or prove code correctness
- Concurrent programming is made difficult since order of evaluation matters.

Why Functional Programming?

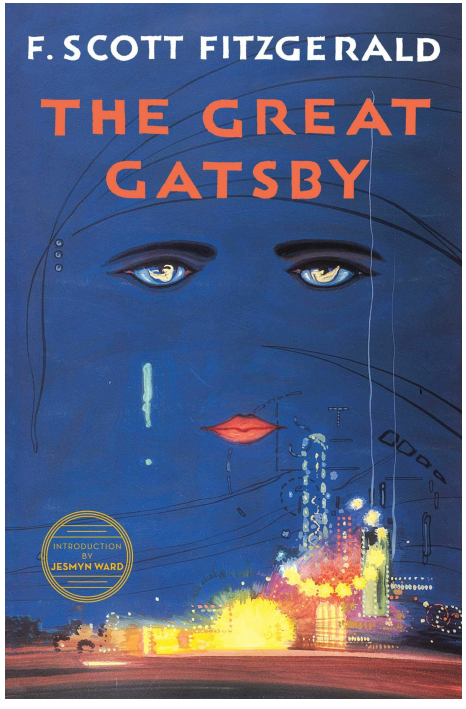


- The function above isn't *referentially transparent*
- Which means it really isn't even a function in the *mathematical* sense.
- This makes it hard to create reproducible behavior or prove code correctness
- Concurrent programming is made difficult since order of evaluation matters.
- Functional idioms creep into imperative languages. Lambda and streams were added to Java, and JavaScript code becomes increasingly functional.

What Things are Beautiful?

What Things are Beautiful?

What Things are Beautiful?



Can Code be Beautiful?

Write a program for summing the squares of numbers from 1 to n.

Can Code be Beautiful?

Write a program for summing the squares of numbers from 1 to n.

```
public class SumSquares
{
   public static void main(final int[] nums)
   {
      int sum = 0;
      for (int i : nums)
        sum += i * i;
      System.out.println("The sum of the squares is:
↪   " + sum);
    }
}
```

An Equivalent Program

Let's get our first look at a functional program written in Racket.

An Equivalent Program

Let's get our first look at a functional program written in Racket.

```
;;Number -> Number
(define (square x) (* x x))

;;Number List -> Number List
(define (sum-of-squares lst)
  (sum (map square lst)))
```

An Equivalent Program

Let's get our first look at a functional program written in Racket.

```
;;Number -> Number
(define (square x) (* x x))

;;Number List -> Number List
(define (sum-of-squares lst)
  (sum (map square lst)))
```

Don't worry about the meaning and syntax of the program, yet. Given time, the parens will become beautiful!

What *Is* Code Beauty?

First let's hear from you!

What *Is* Code Beauty?

First let's hear from you! Let's consider some other views.

- Utilitarian Program Beauty

What *Is* Code Beauty?

First let's hear from you! Let's consider some other views.

- Utilitarian Program Beauty
  - The programming language that maximizes the utility of writing my program is the best. Minimize bugs, execution time, and time until production.

What *Is* Code Beauty?

First let's hear from you! Let's consider some other views.

- Utilitarian Program Beauty
  - The programming language that maximizes the utility of writing my program is the best. Minimize bugs, execution time, and time until production.
  - Critique: Hard to measure which features objectively help the process of writing programs.

What *Is* Code Beauty?

First let's hear from you! Let's consider some other views.

- Utilitarian Program Beauty
  - The programming language that maximizes the utility of writing my program is the best. Minimize bugs, execution time, and time until production.
  - Critique: Hard to measure which features objectively help the process of writing programs.
- Egoist Program Beauty

What *Is* Code Beauty?

First let's hear from you! Let's consider some other views.

- Utilitarian Program Beauty
  - The programming language that maximizes the utility of writing my program is the best. Minimize bugs, execution time, and time until production.
  - Critique: Hard to measure which features objectively help the process of writing programs.
- Egoist Program Beauty
  - The programming language that allows me to write clever, expressive programs is best.

What *Is* Code Beauty?

First let's hear from you! Let's consider some other views.

- Utilitarian Program Beauty
  - The programming language that maximizes the utility of writing my program is the best. Minimize bugs, execution time, and time until production.
  - Critique: Hard to measure which features objectively help the process of writing programs.
- Egoist Program Beauty
  - The programming language that allows me to write clever, expressive programs is best.
  - Critique: Engineering is largely a social endeavor. Other people need to be able to read your programs and understand abstractions.

What *Is* Code Beauty?

First let's hear from you! Let's consider some other views.

- Utilitarian Program Beauty
  - The programming language that maximizes the utility of writing my program is the best. Minimize bugs, execution time, and time until production.
  - Critique: Hard to measure which features objectively help the process of writing programs.
- Egoist Program Beauty
  - The programming language that allows me to write clever, expressive programs is best.
  - Critique: Engineering is largely a social endeavor. Other people need to be able to read your programs and understand abstractions.
- Aesthetic Program Beauty

What *Is* Code Beauty?

First let's hear from you! Let's consider some other views.

- Utilitarian Program Beauty
  - The programming language that maximizes the utility of writing my program is the best. Minimize bugs, execution time, and time until production.
  - Critique: Hard to measure which features objectively help the process of writing programs.
- Egoist Program Beauty
  - The programming language that allows me to write clever, expressive programs is best.
  - Critique: Engineering is largely a social endeavor. Other people need to be able to read your programs and understand abstractions.
- Aesthetic Program Beauty
  - Beauty is subjective according to appreciation of the idioms of some paradigm (i.e. cubism in art).

What *Is* Code Beauty?

First let's hear from you! Let's consider some other views.

- Utilitarian Program Beauty
  - The programming language that maximizes the utility of writing my program is the best. Minimize bugs, execution time, and time until production.
  - Critique: Hard to measure which features objectively help the process of writing programs.
- Egoist Program Beauty
  - The programming language that allows me to write clever, expressive programs is best.
  - Critique: Engineering is largely a social endeavor. Other people need to be able to read your programs and understand abstractions.
- Aesthetic Program Beauty
  - Beauty is subjective according to appreciation of the idioms of some paradigm (i.e. cubism in art).
  - Critique: Disconnected from the empirical or mathematical viewpoints we typically employ to measure the value of scientific contributions.

Looks Like We Don't Know

So, Why Ask?

- Universities are a place for the creation and acquisition of knowledge.

So, Why Ask?

- Universities are a place for the creation and acquisition of knowledge.
- We should put our opinions under scrutiny.

So, Why Ask?

- Universities are a place for the creation and acquisition of knowledge.
- We should put our opinions under scrutiny.
- Computer scientists are critical thinkers.

So, Why Ask?

- Universities are a place for the creation and acquisition of knowledge.
- We should put our opinions under scrutiny.
- Computer scientists are critical thinkers.
- Computer science continues on even though many of the first major results were about undecidability.

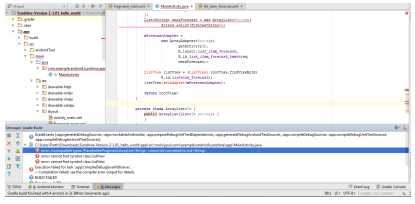
How Do We Write Programs?

How Do We Write Programs?

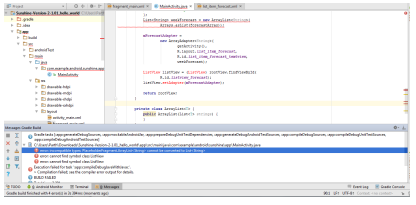
Let's hear from you.

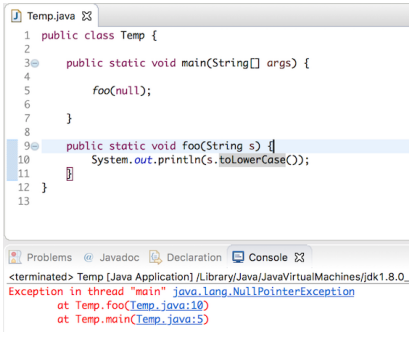
Is Programming Hard?

Is Programming Hard?

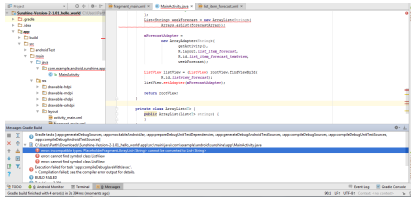


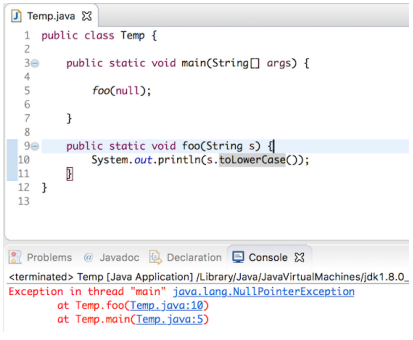
Is Programming Hard?

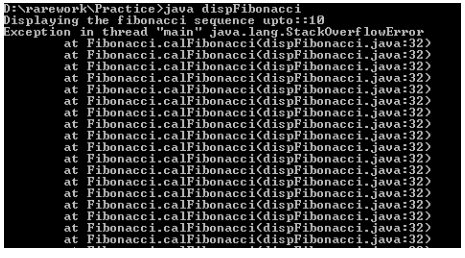




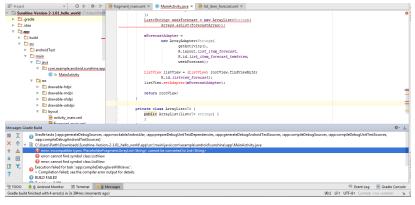
Is Programming Hard?

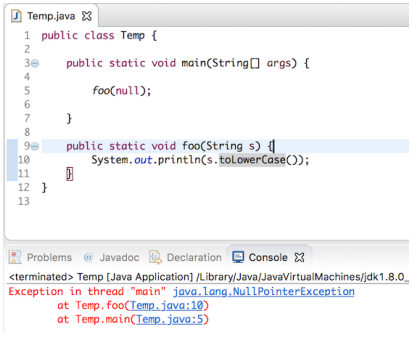




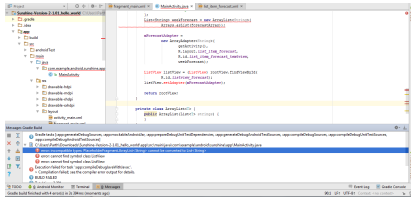


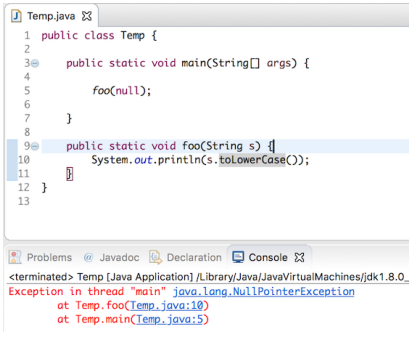
Is Programming Hard?





Is Programming Hard?





How Programming Is Taught

How Programming Is Taught

- Programming courses typically start by motivating new features with concrete programs.

How Programming Is Taught

- Programming courses typically start by motivating new features with concrete programs.
- Given enough examples, students begin to match problem descriptions to using features.

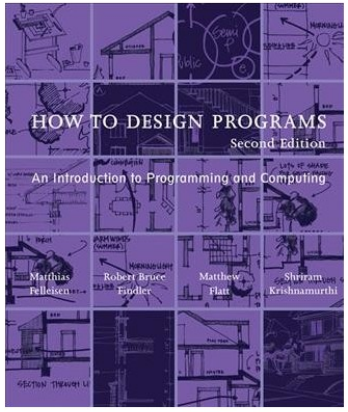
How Programming Is Taught

- Programming courses typically start by motivating new features with concrete programs.
- Given enough examples, students begin to match problem descriptions to using features.
- From there, programming becomes gluing together recognized patterns.

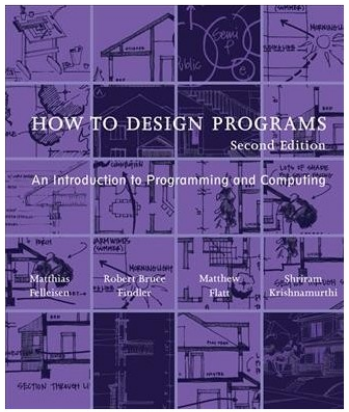
How Programming Is Taught

- Programming courses typically start by motivating new features with concrete programs.
- Given enough examples, students begin to match problem descriptions to using features.
- From there, programming becomes gluing together recognized patterns.
- **There has to be a better way!**

The Better Way

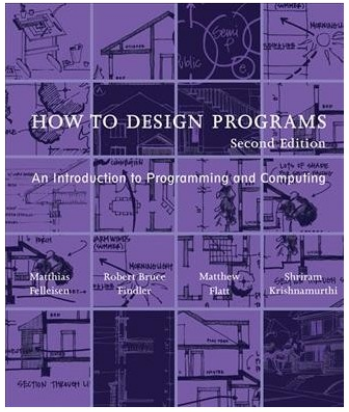


The Better Way



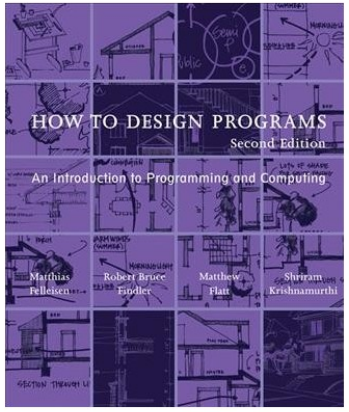
- This book (and course) will teach systematic program design.

The Better Way



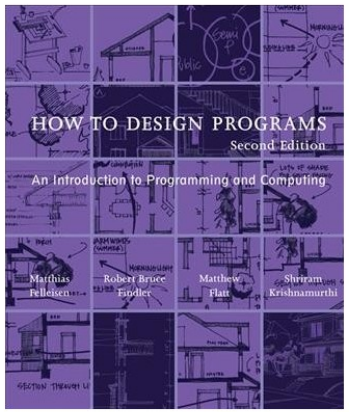
- This book (and course) will teach systematic program design.
- It presents recipes for designing two different types of programs.

The Better Way



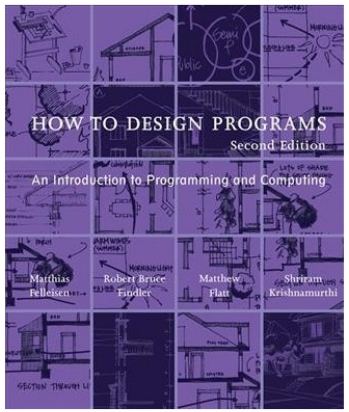
- This book (and course) will teach systematic program design.
- It presents recipes for designing two different types of programs.
    - Big bang (event driven) programs

The Better Way



- This book (and course) will teach systematic program design.
- It presents recipes for designing two different types of programs.
  - Big bang (event driven) programs
  - Batch programs (effectful scripts)

The Better Way



- This book (and course) will teach systematic program design.
- It presents recipes for designing two different types of programs.
  - Big bang (event driven) programs
  - Batch programs (effectful scripts)
- Isn't this a course on functional programming?

Learning Functional Programming from the Ground Up

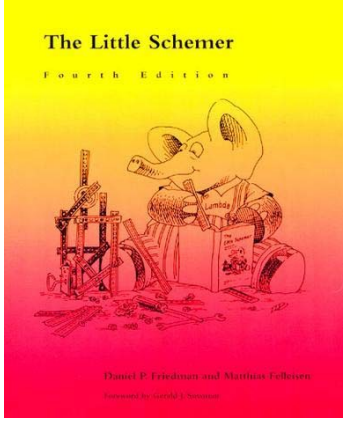
Learning Functional Programming from the Ground Up

- Why How to Design Programs?

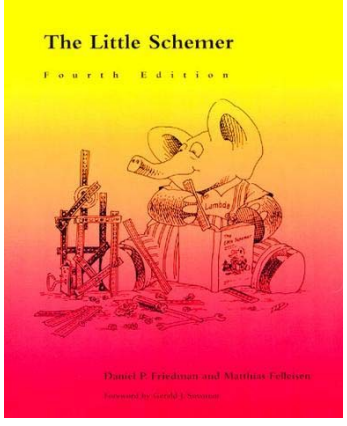
Learning Functional Programming from the Ground Up

- Why How to Design Programs?
- When I first tried to learn functional programming, I failed.

Learning Functional Programming from the Ground Up

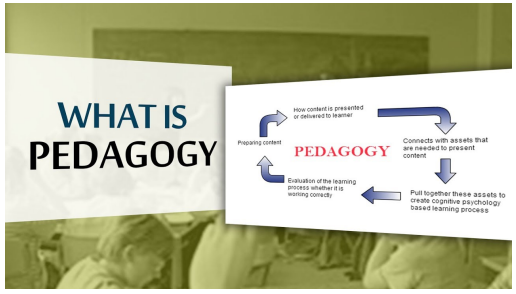
- Why How to Design Programs?
- When I first tried to learn functional programming, I failed.
- 

Learning Functional Programming from the Ground Up

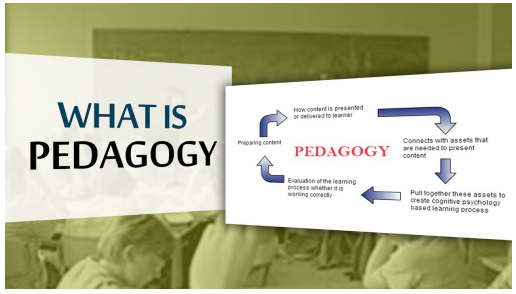
- Why How to Design Programs?
- When I first tried to learn functional programming, I failed.
- 

- We will learn functional programming and systematic design, together!

Why Racket and Not X?

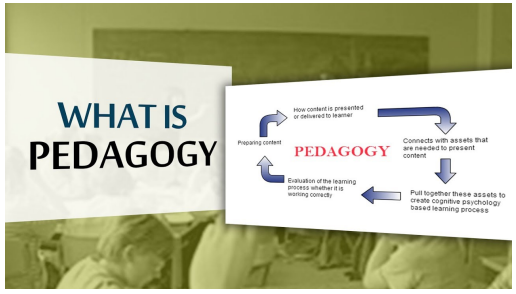


Why Racket and Not X?



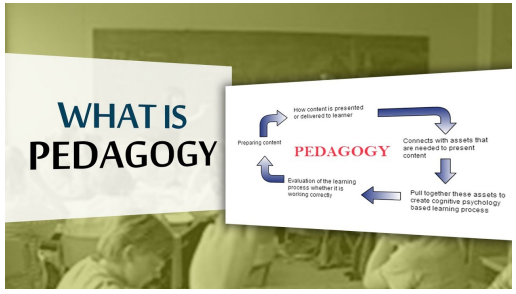
- Teaching functional programming in an imperative-first language is awkward

Why Racket and Not X?



- Teaching functional programming in an imperative-first language is awkward
- Other functional languages have additional complexity

Why Racket and Not X?



- Teaching functional programming in an imperative-first language is awkward
- Other functional languages have additional complexity
- It's focused on being a good language for education

Course Structure

Grades will come from:

- Programming assignments

Course Structure

Grades will come from:

- Programming assignments
- 4 or so projects

Course Structure

Grades will come from:

- Programming assignments
- 4 or so projects
- An easy test and final team project

Course Structure

Grades will come from:

- Programming assignments
- 4 or so projects
- An easy test and final team project
- *Try not to stress about grades and don't cheat*

Course Structure

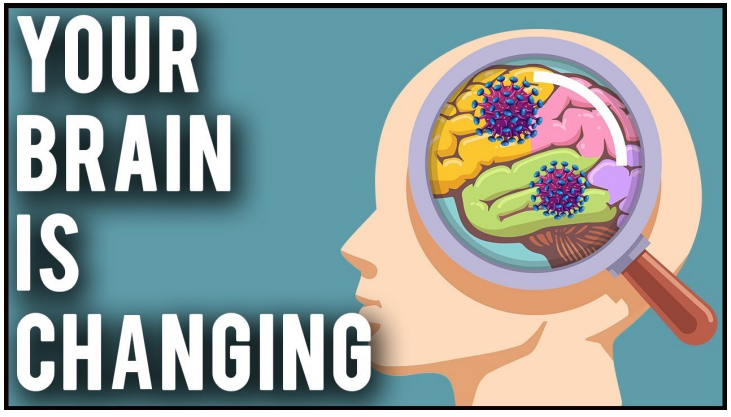
Grades will come from:

- Programming assignments
- 4 or so projects
- An easy test and final team project
- *Try not to stress about grades and don't cheat*
- I don't want to use this:

Moss

A System for Detecting Software Similarity

The Main Things I Want



What do you want?