

Tech Talk

関数プログラミング入門

masatora 2021/05/27

このテーマを扱う動機

ターゲット

ゴール





参考文献

- 関数プログラミング実践入門
- 関数プログラミングはなぜ重要か
- Scala研修テキスト by ドワンゴ
- Scala関数型デザイン&プログラミング

関数プログラミングとはなにか

関数プログラミングの何が嬉しいか

目次

-  **関数プログラミングのデモ** - focus on the content with Markdown, and then style them later
-  **関数プログラミングとはなにか** - theme can be shared and used with npm packages
-  **関数型言語とはなにか** - code highlighting, live coding with autocompletion
-  **関数プログラミングの何が嬉しいか** - embedding Vue components to enhance your expressions
-  **いろんな言語に見られる関数プログラミングの影響** - built-in recording and camera view
-  **関数プログラミングをやってみる** - export into PDF, PNGs, or even a hostable SPA

関数プログラミングのデモ

うんこ

unko

関数プログラミングとはなにか

(再掲)関数プログラミングとは

プログラミングパラダイムの1つ

プログラミングのパラダイム

有名なプログラミングパラダイムとして、以下の3つがある。

1. 命令型プログラミングのパラダイム
2. オブジェクト指向プログラミングのパラダイム
3. 関数プログラミングのパラダイム

1. 命令型プログラミングのパラダイム

プログラムとは、**計算機が行うべき
命令の列**であるとするパラダイム。

2. オブジェクト指向プログラミングのパラダイム

プログラムとは、オブジェクトとそのメッセージングであるとするパラダイム。

3. 関数プログラミングのパラダイム

プログラムとは、**関数**であるとする
パラダイム。値に**関数**を適用してい
くことで計算をすすめる。

関数とはなにか

関数プログラミングにおける関数とは、数学的な意味での関数。

すなわち、与えられた入力の値のみから出力となる値をただ 1 つ決める規則のこと。

言い換えると、プログラミングが対象とする処理内容のうち、副作用のないもの。

副作用とは

副作用とは、状態を参照し、あるいは状態に変化を与えることで、次回以降の結果にまで影響を与える効果のこと。

関数型言語とはなにか

関数型言語とは、関数が第一級の対象である言語のこと。

関数型言語とはなにか

第一級の対象であるとは

1. リテラルがある

2. 実行時に生成できる

3. 変数に入れて扱える

4. 手続きや関数に引数として与えることができる

5. 手続きや関数の結果として返すことができる

命令型言語と関数型言語の違い

2. 代入 vs 束縛

関数型言語の例と分類

関数型言語の例

関数型言語の分類

1. 動的型付けと静的型付け

2. 型付けの強弱

3. 純粹

4. 型推論

5. 依存型

6. 評估戰略

関数型言語の歴史

関数型言語のこれから

関数型言語が採用されている分野/プロダクト

関数プログラミングの何が嬉しいか

関数プログラミングのメリット

1. コード量が少なくなる

2. 最適化がしやすい

3. 並行/並列化がしやすい

4. バグり/バグらせにくい

5. ドキュメントが少なくなる

関数型言語のメリット

1. 宣言的であることのメリット

2. 制約の充足をチェックしてくれるメリット

3. 型と型検査があることのメリット

4. 型推論のメリット

5. 束縛によるメリット

いろんな言語に見られる関数プ
ログラミングの影響

関数の扱いやすさ

1. ラムダ式

2. 部分適用

3. 演算子

データ型定義とパターンマッチ

Rust, Swift

型システムの強化

型推論

リスト内包表記

Python, C#

モナド

Java, Swift

コンパイル時計算

C++

関数プログラミングをやってみる

a

Unko