

Functional Programming - Go vs. F#

By Maximilian Gobbel.

Agenda

1. Functional Programming Concepts
2. Introduction to Go and F#
3. NerdDeck Project Overview
4. Demo
5. Learnings

“

**ALL RACE CONDITIONS,
DEADLOCK CONDITIONS,
AND CONCURRENT UPDATE
PROBLEMS ARE DUE TO
MUTABLE VARIABLES.**

- Robert C. Martin, Clean Architecture



Functional Programming Concepts



LAMBDA CALCULUS

The fundamental concept of functional programming which involves a mathematical representation of functions.



HIGHER-ORDER FUNCTIONS

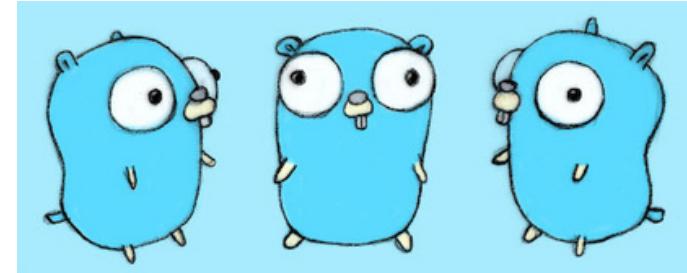
Utilization of functions as parameters and return values for enhanced modularity.



IMMUTABILITY

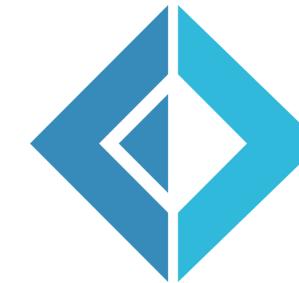
The idea that once data is created, it cannot be modified. Instead, new data is created based on the original data.

Introduction to Go and F#



GO(LANG)

- Developed by Google Developers in 2007, released in 2009
- Emphasizes for simplicity, readability, and efficiency
- Strong support for concurrency
- Widely used in backend development, cloud services, and containerization



F#

- Developed by Microsoft Research in 2005
- Functional-first programming language in the ML family
- Runs on the .NET platform
- Strong type inference and immutability
- Suitable for data manipulation, parallel programming, and web development



NerdDeck Project Overview

A functional programming project for every student.

NerdDeck Project Overview

Objective

Develop a flash card app resembling Anki but with streamlined functionality.

Implementation Details

- Sole reliance on a JSON file as the database.
- Exclusively Command Line Interface (CLI) without graphical user interface.
- Emphasis on the use of a single deck to mitigate system complexity.

ID	Requirement
1	As a student, I want to create a flashcard inside a deck.
2	As a student, I want to view my flashcards.
3	As a student, I would like to learn how to utilize the program.
4	As a student, I want each flashcard to have a unique combination of a question and an answer to avoid duplicates.
5	As a student, I would like to utilize a spaced repetition algorithm to enhance my learning with flashcards.

FlashCard	
PK	ID: string
	Question: string
	Answer: string
	Repetitions: int
	EasinessFactor: int
	NextReview: datetime



Demo

LEARNINGS



Learnings



LEARNING COMPLEXITY

Mastering two languages with non-trivial paradigms is demanding.



IMPURITY IN GO AND ALSO F#

Inherent impurity complicates adherence to programming principles. A pure language like Haskell would be a solution.



USE CASE DEPENDENCY

Adopting a functional language depends on the project's nature; mismatch can introduce unnecessary complexity.



Thank you for your attention



<https://github.com/maex0/nerddeck>

References

- .NET programming languages | C#, F#, and Visual Basic. <https://dotnet.microsoft.com/en-us/languages>. (Accessed on 12/20/2023)
- F# Software Foundation. <https://fsharp.org/>. (Accessed on 12/21/2023)
- The Go Programming Language. <https://go.dev/>. (Accessed on 11/27/2023)
- J. Hughes. Why Functional Programming Matters. *The Computer Journal*, 32(2):98–107, feb 1989
- What is F# - .NET | Microsoft Learn. <https://learn.microsoft.com/en-us/dotnet/fsharp/what-is-fsharp>. (Accessed on 12/21/2023)
- Application of a computer to improve the results obtained in working with the SuperMemo method - SuperMemo. <https://www.supermemo.com/en/blog/application-of-a-computer-to-improve-the-results-obtained-in-working-with-the-supermemo-method>. (Accessed on 12/01/2023)
- A. L. Tharp. Selecting the “Right” Programming Language. *SIGCSE Bull.*, 14(1):151–155, feb 1982
- TIOBE Index - TIOBE. <https://www.tiobe.com/tiobe-index/>. (Accessed on 11/27/2023)
- <https://www.goodreads.com/quotes/tag/functional-programming> (Accessed on 01/10/2024)