FREE eBook

LEARNING

F#

Free unaffiliated eBook created from
Stack Overflow contributors.

#f#

Table of Contents

About 1

Chapter 1: Getting started with F# 2

Remarks 2

Versions 2

Examples 2

Installation or Setup 2

Windows 2

OS X 2

Linux 3

Hello, World! 3

F# Interactive 3

Chapter 2: 1 : F# WPF Code Behind Application with FsXaml 5

Introduction 5

Examples 5

Create a new F# WPF Code Behind Application. 5

3 : Add an icon to a window 7

4 : Add icon to application 7

2 : Add a control 8

How to add controls from third party libraries 9

Chapter 3: Active Patterns 10

Examples 10

Simple Active Patterns 10

Active Patterns with parameters 10

Active Patterns can be used to validate and transform function arguments 10

Active Patterns as .NET API wrappers 12

Complete and Partial Active Patterns 13

Chapter 4: Classes 14

Examples 14

Declaring a class 14

Creating an instance 14

Chapter 5: Design pattern implementation in F# 15
Examples 15
Data-driven programming in F# 15
Chapter 6: Discriminated Unions 18
Examples 18
Naming elements of tuples within discriminated unions 18
Basic Discriminated Union Usage 18
Enum-style unions 18
Converting to and from strings with Reflection 19
Single case discriminated union 19
Using Single-case Discriminated Unions as Records 19
RequireQualifiedAccess 20
Recursive discriminated unions 20
Recursive type 20
Mutually dependent recursive types 21
Chapter 7: F# on .NET Core 22
Examples 22
Creating a new project via dotnet CLI 22
Initial project workflow 22
Chapter 8: F# Performance Tips and Tricks 23
Examples 23
Using tail-recursion for efficient iteration 23
Measure and Verify your performance assumptions 24
Comparison of different F# data pipelines 33
Chapter 9: Folds 42
Examples 42
Intro to folds, with a handful of examples 42
Calculating the sum of all numbers 42
Counting elemets in a list (implementing count) 42
Finding the maximum of list 43
Finding the minimum of a list 43

Concatenating lists 43

Calculating the factorial of a number 44

Implementing forall, exists and contains 44

Implementing reverse: 45

Implementing map and filter 45

Calculating the sum of all elements of a list 45

Chapter 10: Functions 47

Examples 47

Functions of more than one parameter 47

Basics of functions 48

Curried vs Tupled Functions 48

Inlining 49

Pipe Forward and Backward 50

Chapter 11: Generics 52

Examples 52

Reversal of a list of any type 52

Mapping a list into a different type 52

Chapter 12: Introduction to WPF in F# 54

Introduction 54

Remarks 54

Examples 54

FSharp.ViewModule 54

Gjallarhorn 56

Chapter 13: Lazy Evaluation 59

Examples 59

Lazy Evaluation Introduction 59

Introduction to Lazy Evaluation in F# 59

Chapter 14: Lists 61

Syntax 61

Examples 61

Basic List Usage 61

Calculating the total sum of numbers in a list 61

Creating lists 62

Chapter 15: Mailbox Processor 65

Remarks 65

Examples 65

Basic Hello World 65

Mutable State Management 66

Concurrency 67

True mutable state 67

Return Values 68

Out-of-Order Message Processing 69

Chapter 16: Memoization 70

Examples 70

Simple memoization 70

Memoization in a recursive function 71

Chapter 17: Monads 73

Examples 73

Understanding Monads comes from practice 73

Computation Expressions provide an alternative syntax to chain Monads 81

Chapter 18: Operators 84

Examples 84

How to compose values and functions using common operators 84

Latebinding in F# using ? operator 85

Chapter 19: Option types 87

Examples 87

Definition of Option 87

Use Option<'T> over null values 87

Option Module enables Railway Oriented Programming 88

Using Option types from C# 89

Pre-F# 4.0 89

F# 4.0 89

Chapter 20: Pattern Matching 91

Click here to download full PDF material