

# Index

## A

### Abstract class

- abstract members, 211
- default member implementation, 211

### Active pattern, 137, 149

- & operator, 143
- complete, 142
- default match, 141
- efficiency, 144
- multi case, 139
- parameterized, 141
- partial, 140
- single case, 137

### ALGOL W, 29

### Alignment, 364, 374

### AllowNullLiteral attribute, 216

### append, 65, 71

### ArgumentException, 80

### Array

- clamps `[[[]]]`, 124, 320
- indexed access, 315
- vs.* list, 317
- module, 63
- resize, 107
- vs.* sequence, 318, 327
- slicing, 67

### Array clamps (`[[]...[]]`), 320

### Array2D module, 87

### Array3D module, 87

### Array4D module, 87

### Array module, 61, 63

- average, 84
- contains, 343
- filter, 63
- find, 343
- head, 80
- init, 63
- iter, 64
- length, 320
- map, 346
- Parallel.map, 66, 340, 342, 346
- sort, 63
- sortInPlaceWith, 66
- tryAverage, 84
- tryFind, 343
- zip, 81

### Asterisk (`* int`), 235

### Asymmetric representation, 183

### Asynchronous programming, 251, 259

- batching, 267
- cancellation token, 275
- imperative operation, 261
- lock expression, 266
- recommendations for design, 276
- testing downloads, 267
- throttling, 271

### async computation expression, 260

## INDEX

Async module, 261  
    AwaitTask, 261  
    Parallel, 264, 269–270, 272, 274  
    ParallelWithThrottle, 271–272  
    RunSynchronously, 263, 270, 274  
Auto-implemented property, 189  
average, 65, 68, 81  
averageBy, 68

**B**

Bad practices, 9  
Banana clips (|...|), 137  
Benchmark attribute, 311  
BenchmarkDotNet, 310–312, 315, 346–347  
    creating a project, 310  
    diagnostic output, 312  
    inappropriate collection type, 312  
        Array module, 317  
        avoiding collection functions, 320  
        avoiding loops having skips, 321  
        harness module, 314  
        improvements, 324  
        sequences instead of arrays, 318  
    initialization, 315  
    naive string building  
        concatenation, 338  
        using Array.Parallel.map, 341  
ShortTermObjects  
    \*\* operator, 332  
    harness module, 326  
    Point3d's methods, 328  
    results, 327  
    withinRadius function, 326  
BillingDetails type, 44–46  
Block selection (editor), 370  
Bounds checking, 15

break, 109  
Bug, 15, 29, 374  
Business rule, 44  
By reference parameter, 52, 125

## C

C#, 6, 16, 30–31, 47, 51–52, 62–63, 135, 162, 168, 183, 288  
    8.0, 54  
    Task, 261  
        *vs.* F# Async, 274  
Cardinality (of collection functions), 65  
Case (of Discriminated Union), 32  
Casting, 23  
Chain (measurement), 18, 25  
choose, 65, 67, 89–90  
Classes, 11, 183, 380  
    abstract, 210  
    additional constructors, 189  
    asymmetric representation, console  
        prompt declaration, 185  
    backing store, 192, 200  
    BeepOnError property, 189  
    collection property, 198  
    comparison, 213  
    constructor body, 185–186, 196  
        binding values, 186  
        imperative operations, 186  
    constructor values as  
        members, 188  
    equality, 213  
    generic, 193, 195  
    getters, 188, 190  
    hierarchy, 183  
    implement IComparable,  
        LatLon class, 218, 221

- implement IEquatable
  - Equals() method, 216
  - GetHashCode(), 216
  - LatLon class, 214
- implementing interfaces, 202–204
- indexed property, 198, 200
- internal mutable state, 192
- member, recursive, 185
- mutable property, 188
- named argument syntax, 198
- named parameters, 196
- object initialization syntax,
  - construction styles, 196, 197
- property validation, 190
- recommendations for design, 221
- vs.* record, 183
- self identifier, 185
- setters, 188, 190
- syntax, 185
- thread safety, 193
- values as members, 187
- Code duplication, 8
- Coding standards, 1
- Cold task, 274
- collect, 65, 70, 89
- Collection functions, 61, 62
  - Array module, 63
  - Array.ofSeq/Seq.toArray, 88–89
  - Collection.toOtherCollection*, 88
  - commonly used, 65
  - list module, 63
  - Many-of-Many to Many operations, 70
  - 2-of-Many to Many operations, 71
  - Many to 2-of-Many operations, 71
  - Many-to-Boolean operations, 69
  - Many-to-Equally Many operations, 66
  - Many-to-Fewer operations, 66–68
  - Many-to-Groupings operations, 71
  - Many-to-Nothing operations, 70
  - Many-to-One function, 68
  - Microsoft.FSharp.Collections
    - namespace, 87
  - missing, 88
  - mistakes, 89–90, 93
  - Nothing-to-Many operations, 69
  - ordering, parameters, 63
  - partial (*see* Partial functions)
  - recommendations for design, 93
  - Set.map operation, 88
  - setup code, exercises, 72
  - single, 74
- Collection type, 312, 322
- Comma Separated Values (CSV), 336
- Comment, 367, 369, 374
- Communication (failure of), 2, 5, 8
- compare (built in function), 219
- Comparison, 218
- Compiler warnings, 120, 121
- Complexity explosion, 8, 374
- Complex numbers, 131
- Comprehensions, 69, 324
  - array, 320
  - list, 313, 320
- concat, 65, 70
- Conciseness, 22
- Console, 183, 252, 310
- Console program
  - DU, 253
  - getLinks function, 254
  - printing colored messages, 252
  - tryDownload function, 255–256
- Cons operator (::), 343
- Constructor (of Discriminated Union), 21
- Content equality. Structural equality
- Cost-benefit analysis, 14
- CPU, 261

## INDEX

- Cumulative computation, 114
- Curried arguments, 231, 243
- Currying, 128, 201, 231, 233
  - code reuse, 232
  - type signature, 235

## D

- DateTime, 343, 346
- DateTimeOffset, 343, 346
- Debugging, 364
- Decomposition, 123
  - arrays, 123
  - discriminated unions, 129
  - lists, 123
  - records, 126
- Default case, 147
- default (keyword), 211
- Delivery address, 35, 37, 40–41,  
43–45, 47
- Deserialization, 21
- Designing functions
  - fromMilesPointYards function, 21
  - miles terminology, 12
  - Object Oriented (OO)  
programming, 11
- Dictionary, 343
- Discriminated Union (DU), 16, 18, 32, 44,  
128, 202, 253, 289, 296, 362, 373
  - error-types, 298
  - generic, 289
  - labels, 129, 131, 133, 148, 363
  - layout, 363
  - map function, 299
  - mapError, 299
  - payload, 47, 128
- distinct, 65, 67
- distinctBy, 67

- do (keyword), 186
- do! (keyword), 261
- Don't Repeat Yourself (DRY), 100
- \_\_ (double underscore) naming  
convention, 185, 363

## E

- Efficiency, 309
- Elixir, 19
- Ergonomics, 32
- Errors, 5, 13–14, 283, 295, 296
- Exception, 15, 30, 283, 288
- exists, 65, 69
- Explicit pattern matching, 120

## F

- %f format specifier, 336
- Filtering, 62, 65, 67, 323
- find, 65, 68, 81
- Find extreme value, 111
- Flag value, 104
- Floating point, 214
- fold, 65, 68
- foldBack, 68
- Fold operation, 114
- forall, 65, 69
- For loop, 313, 343, 346
  - skipping, 320, 321
- Forward piping, 64, 90
- FSharp.Data, 254
- FSharpX, 271
- Function, 11, 229, 253
  - body, 13
  - composition, 240, 243, 294, 361
  - concealing mutable state, 239
  - as first class value, 229

- injection, 194
- in module, 359–360, 363
- naming, 373
- as a parameter, 236
- recommendations for design, 24, 243
- returning functions, 238
- signature, 13–14
- type hint, 236
- type inference, 236
- type signature, 194, 235, 238, 289

fun (keyword), 238

## G

- Garbage collection, 317, 319, 324, 327, 338
- Generic Discriminated Union, 32
- Generic records, 169
- Geographic Information
  - System (GIS), 12, 15, 17
- getLinks function, 254
- git, 30
- GitHub, 31
- Google n-grams, 267
- Greek characters (in code), 369
- groupBy, 65, 71
- Guarded linear search, 105–106

## H

- Handling, *see* Exception
- HasFlag (discriminated union
  - member), 136
- hash (built in function), 217
- Hash code, 216
- Haversine formula, 173
- head, 65, 68, 80–81, 124
- head::tail, 323
- Higher order function, 63, 87

- Hoare, Tony, 29, 374
- Hot task, 274
- HtmlDocument, 259
  - AsyncLoad, 259
  - Load, 254

## I, J, K

- IComparable, 213, 218
  - CompareTo() method, 219
  - generic *vs.* non-generic, 219
- IDisposable, 207, 213
- IEnumerable, 62–63
- IEquatable, 213–214
  - Equals method, 214
  - implementing, 214
- Immutability and mutation, 99
  - acc and elem, 114
  - coding, 64
  - coding patterns, 103
  - cumulative computation, 114
  - processing, repeat until, 110
  - process items imperatively, 106
  - programming, 114
  - recommendations for design, 115
  - Seq.max function, 101
- Indentation, 371
- init, 65, 69
- Initialization value, 104
- initInfinite, 69
- inline, 84
- Instance methods, 173
- Int32.TryParse, 125
- Intellisense, 169
- intent, 5
- Interface, 202
  - abstract member definition, 203
  - accessing members, 204

## INDEX

### Interface (*cont.*)

- definition, 202
  - explicit implementation  
(C# *vs.* F#), 207
  - implementation, 203
- Interop, 47, 137
- IsCase (option type property), 51
- isEmpty, 65, 69
- IsSome (option type property), 43
- Item (option type property), 51
- iter, 65, 70
- iteri, 65, 70

## L

- Lambda function, length of, 63, 89, 91
- Latitude and longitude, 162, 213
- Layout and naming
- alignment, 364–365
  - antipattern, 354
  - infelicitous code, 356
  - recommendations for design, 373
- length, 65, 68
- Let binding, 132
- let! (keyword), 260–261
- Linear search, 103–104
- Linked list, *see* List
- LINQ, 62–63, 115
- List, 315
- vs.* Array, 317
  - concatenation, 343
  - indexed access, 315, 323
- List module, 63
- length, 320
  - map, 316
- Locking, 265
- lock (keyword), 265
- Loop, 100

## M

- Magnitude (astronomical), 358
- Maintainability, 5, 309
- Making illegal state
- unrepresentable, 15, 21, 47
- map function, 65–66, 108, 299
- mapi, 65–66
- Map module, 87
- match, 120, 147
- Match expression, 18, 22, 35–36,  
46, 373
- match! (keyword), 262
- max, 65, 68, 81
- maxBy, 65, 68, 81
- Mechanical sympathy
- (principle of), 7, 8, 347, 384
- Memory allocation, 338
- MemoryDiagnoser attribute, 311
- Mental overhead, 7, 364–365, 373
- Microsoft Azure, 341
- Microsoft.FSharp.Collections
- namespace, 87
- Miles, 11
- MilesYards module, 175
- min, 65, 68, 81
- minBy, 65, 68, 81
- Minor planet, 257, 352, 366
- type
    - code demonstration, 372
    - tidier version, 368–369
- Minor Planet Center Orbit Database  
(MPCORB), 352
- Missing data
- BillingDetails, 46
  - Delivery DU, 47
  - delivery address, 45
  - motivational transparency, 47

- recommendations for design, 55
- tryDeliveryLabel, 46
- ML (language), 19
- Modeling, 30
- Module, 19, 373
- Motivational transparency (principle of),
  - 6, 8, 15, 18, 36, 47, 85, 101, 130, 150,
  - 167, 202, 243, 302, 319, 321, 347,
  - 359, 363, 384
- MSDN F# Style Guide, 169
- Mutation, 99

## N

- Naming, 19
- .NET Core, 310
- NewCase, 51
- NoComparison attribute, 165
- NoEquality attribute, 165
- None (option case), 33–34, 102
- Non-null value, 30
- Nuget, 254, 271, 310
- Nullable integer, 53
- Nullable reference types, 54
- Nullable strings, 50
- Null coalescing operator (??), 30, 32
- Null-conditional operator
  - (?. and ?[]), 30, 32
- Null reference errors, 29–30
  - ALGOL W code, 29–30
  - .NET framework, 54
  - and non-nullable types, 54
  - Option.ofObj, 50–51, 53
  - Option.toNullable, 53
  - vs.* option types, 32–34
  - Visual Studio 2017 Team
    - Explorer, 31
- NullReferenceException, 34, 47

## O

- O(1) operation, 315
- Objects, 162
  - Equals, 162
  - GetHashCode, 162
  - short-term, 324
- Object expression, 207
- Object initializer syntax, 167
- Object Oriented (OO)
  - programming, 11, 29
- OCaml, 15
- %O format specifier, 175
- ofSeq, 88
- O(n) operation, 315
- op\_Equality, 218
- Operator
  - creation, 23
  - ::, 124, 343
  - ?:, 145
  - @, 343, 346
  - \*\*, 332, 336, 346
  - =, 217
  - ==, 218
  - performance, 332, 336, 346
- Opposition (astronomical), 354, 361
- Optimization, 321, 324, 346
- Option.bind function, 40, 43
- option keyword, 33–34
- Option.map, 39–40
- Option module, 36, 147
  - bind and map
    - compared, 38–41, 103, 296
  - count, 38
  - defaultValue, 37, 103, 146
  - defaultWith, 195
  - iter, 38, 40
  - map, 38–40, 46, 103, 296

## INDEX

- Option module (*cont.*)
  - map (diagrammed), 40
  - ofNullable, 50
  - ofObj, 50, 110, 146
  - toArray, 38
  - toList, 38
  - toNullable, 53
  - toObj, 51
- Option.ofObj, 53
- Option types, 32
  - delivery address, 34
  - myOrder case, 35
  - vs.* null, 32, 34
  - Option.bind, 39–41, 43
  - Option.defaultValue function, 37
  - pattern matching, 35, 44
  - ValueOption, 55
  - viewed as a DU, 33
- Orbital elements, 352
- Overriding method, 174–175
  
- P**
- Paket, 254, 271, 310
- Parallel.mapi, 66
- Parallel programming,
  - throttling, 272–273
- Parameter declaration and pattern matching, 22
- Parameters (ordering), 37
- Partial application, 169, 231–232, 252
- Partial functions, 80, 102, 105
  - coding around, 82
  - errors, 80
  - head function, 80
  - list of, 81
  - partial application, 80
  - try... version (*see* try... functions)
- Partition, 65, 71, 88
- Pass-through adapter, 290
- Pattern matching, 18, 36, 119, 146
  - & operator, 143
  - arrays, 122, 148
  - in C#, 119
  - on discriminated unions in function parameters, 130, 132
  - DU, 128, 131, 148
  - enums, 135, 136, 149
  - in F#, 120
  - features, 149
  - at function call boundary, 23, 127
  - in lambdas, 133
  - in let bindings, 132, 133
  - list of features, 147
  - lists, 122–124, 148
  - in loops, 133–134, 149
  - matched value, 121
  - multiple match cases, 121
  - on null, 146, 149
  - recommendations for design, 147
  - record types, 126, 128, 148
  - tuples, 125, 148
  - types, 144, 149, 216, 219
  - When Guards, 122
  - with '&', 144
  - without match
    - expression, 126
- Payload, 32
  - of DU, 32
  - of option type, 35
- Performance, 310, 323, 382
  - antipatterns, 312
  - F# 4.5 and Span Support, 344
- Petricek, Tomas, 275
- pick, 65, 68, 81
- Pinker, Steven, 2



- Pipeline, 302, 318
    - of collection functions, 323
    - layout, 364, 373
    - pipeline friendly coding, 64
  - Postal code, 41, 141
  - pown function, 332, 346
  - printf, 184
  - printfn, 184, 230
  - private (keyword), 21, 363
  - Process items imperatively, 106
  - Programming by coincidence, 9
  - Programming with functions, 380
    - mixed tupled and curried
      - styles, 233–234
    - partial application, 231
- Q**
- Quote marks, 240
- R**
- Railroad, 11
  - Railway, 11
  - Railway Oriented Programming (ROP), 40, 43, 283, 382
    - building production line
      - tiny naming changes, 296
      - validateAndSave function, 293, 294
    - definition, 283
    - exception, 294
    - widget factory, adapter housings, 286
  - Readability, 309
  - ReasonML, 19
  - Record type
    - accessing fields, 158
    - vs.* class, 161, 162, 168
    - CLIMutable attribute, 161
    - constructor, 174
    - copy-and-update, 160
    - declaring, 157
    - disambiguating, 158
    - of functions, 168
    - generic, 169–171
      - equality, 170
      - type inference, 170
    - and immutability, 158
    - instantiation, 157–158, 167, 173
    - as interface, 168
    - layout, 175–177, 366, 374
    - methods
      - design considerations, 175
      - instance, 172–173
      - override, 174
      - static, 173–174
    - module, 175
    - mutable fields, 159, 160
    - recommendations for design, 177
    - recursive, circular, 171, 172
    - serialization, 160
    - static method, 174
    - struct, 165
      - mutability, 166
      - mutually referential, 166
    - structural equality, 162
    - ToString(), 174
    - when not to use, 168
  - Recursion, 110
  - Recursive records, 171
  - reduce, 65, 68, 81
  - reduceBack, 68
  - Refactoring, 5, 9
  - ReferenceEquality attribute, 164, 165
  - Referential equality, 162, 213
    - record fields, 163
    - records, 164

## INDEX

Regular expression, 141, 254, 257  
Repeat until, 108–110  
ResizeArray, 107  
Result module, 295  
    bind, 296  
    map, 296  
    mapError, 298  
return (keyword), 260–261  
rev, 65–66  
Revisability (principle of), 5, 15, 18, 47,  
    101, 383  
Ring buffer, 198–199  
Root mean square (RMS), 112  
ROP, *see* Railway Oriented  
    Programming (ROP)

## S

SafeString type, 49  
Select (LINQ), 61  
Self identifier, *see* Classes  
Semantic focus (principle of), 4, 8, 18, 36,  
    44, 46, 84–85, 167, 196, 243, 288,  
    347, 358, 383  
Separation of concerns, 17  
Seq module, 63, 324  
    average, 112  
    averageBy, 112  
    choose, 46  
    chunkBySize, 270, 274  
    find, 104  
    fold, 114  
    iter, 107  
    map, 101, 108, 264, 346  
    max, 101  
    min, 101  
    ofSeq, 319  
    sum, 112  
    sumBy, 112  
    tryFind, 106  
    tryMax, 102, 111  
    tryMaxBy, 111  
    zip, 81  
Sequence, 63  
    *vs.* Array, 318, 327  
    expression, 110  
Serialization, 21  
Set module, 87, 88  
Shadowing, 186  
Shape Discriminated Union, 32  
Shared resources, 265  
Side effects, 70  
Signature review, immutable  
    class, C#, 16  
Single Case Discriminated  
    Union, 16, 23, 131  
    construction, 17  
    with field names, 16  
snd function, 316  
Some (option case), 33–34, 102  
sortBy, 65–66  
sortByDescending, 66  
Sorting, 62, 65–66  
sortWith, 66  
Software engineering, 8  
Span, 344  
sprintf function, 233, 338  
SQL, 62  
State, 7  
String, 336  
    building, 336, 342, 346  
    Format, 341  
    Join, 339, 342, 346  
    Replace, 240  
    Substring, 361  
    Trim, 361

- StringBuilder, 338, 342, 346
    - Append(), 338
    - AppendLine(), 338
  - String options, 35, 37
  - Struct attribute, 165
  - struct (keyword), 54, 330
  - Structural equality, 163, 213
    - methods, 172
    - record fields, 163
  - Style guide, 2
    - authorial intent, 6
    - checkRate and checkAmount, 4
    - experimentation, 7
    - F#, 301
    - The Sense of Style, 2
  - sub, 65, 67, 81
  - sum, 65, 68
  - sumBy, 68
  - Summarize a collection, 112–113
  - Syme, Don, 222
  - Synchronous programming, 252
  - System.Collections.Generic.List, 107
  - System.Diagnostics.Stopwatch, 256
  - System.Double.TryParse(), 52
  - System.IEquatable, 162
  - System.Nullable, 50
  - System.Object, 215
    - Equals() method, 215
    - GetHashCode() method, 216
  - System.Random, 239
  - System.Windows.Forms, 144
  - Sytem.Flags, 135
- T**
- Tag (option type property), 51
  - tail, 65, 67, 81, 124
  - take, 65, 67
  - takeWhile, 67
  - Testability, 8
  - Test Driven Design, 8
  - Tests, 344
  - Thread, 258
    - ID, 252
    - usage, 259, 266
  - Throttling, *see* Asynchronous programming
  - tidy() function, 292
  - Time complexity, *see* O(1) operation
  - T (naming convention), 19
  - toArray, 88
  - Tools, 1
    - Resharper, 1
    - StyleCop, 1
  - Tool tip, 367
  - Total function, 82
  - Transforming, 62
  - truncate, 65, 67
  - tryDownload function, 255
  - tryFind, 68, 82
  - try... function
    - consuming values, 86
    - filtering and averaging, 86
  - tryHead, 81
  - Try idiom, 84, 359, 373
  - tryLast, 81
  - tryPick, 68
  - Tuple, 90, 219
    - arguments, 233
    - length of, 90
    - performance, 329
    - vs.* records, 92
    - vs.* record type, 90
    - struct, 330, 335
  - Tupled arguments, *see* Currying
  - Twitter, 20

## INDEX

- Type, naming, 11, 373
- Type alias, 202, 326
- Type class, 183
- Type hint, 236
- Type parameter, 169, 193

## U

- Underscore (`_`), 62, 359
- unfold, 65, 69
- unit (keyword), 203
- Unit tests, 13
- Unix, 135
- Upcast operator, 204
- URL, 254
- use (keyword), 207
- USGS, 139

## V

- Validation, 290
- val (keyword), 188
- ValueNone (ValueOption case), 54
- ValueOption type, 54, 55
- Value (option type property), 43, 89
- ValueSome (ValueOption case), 54
- Variable declaration, 99
- Visual Studio 2017, 30

- Visual Studio Code, 370
- voption keyword, 54

## W, X

- Warnings as errors, 120
- WebClient.DownloadFile, 255, 261
- WebClient.DownloadFileAsync, 261
- WebClient.DownloadFileTaskAsync, 261
- When guard, 122, 147
- Where (LINQ), 61, 63
- Wildcard, 147, 373
- Wildcard pattern, 145
- Wind turbine, 139
- Wlaschin, Scott, 283

## Y

- Yards, 11
- Yaron Minsky, 15
- yield!, 110
- yield (keyword), 313

## Z

- zeroCreate, 69
- zip, 65, 71, 81
- zip3, 71