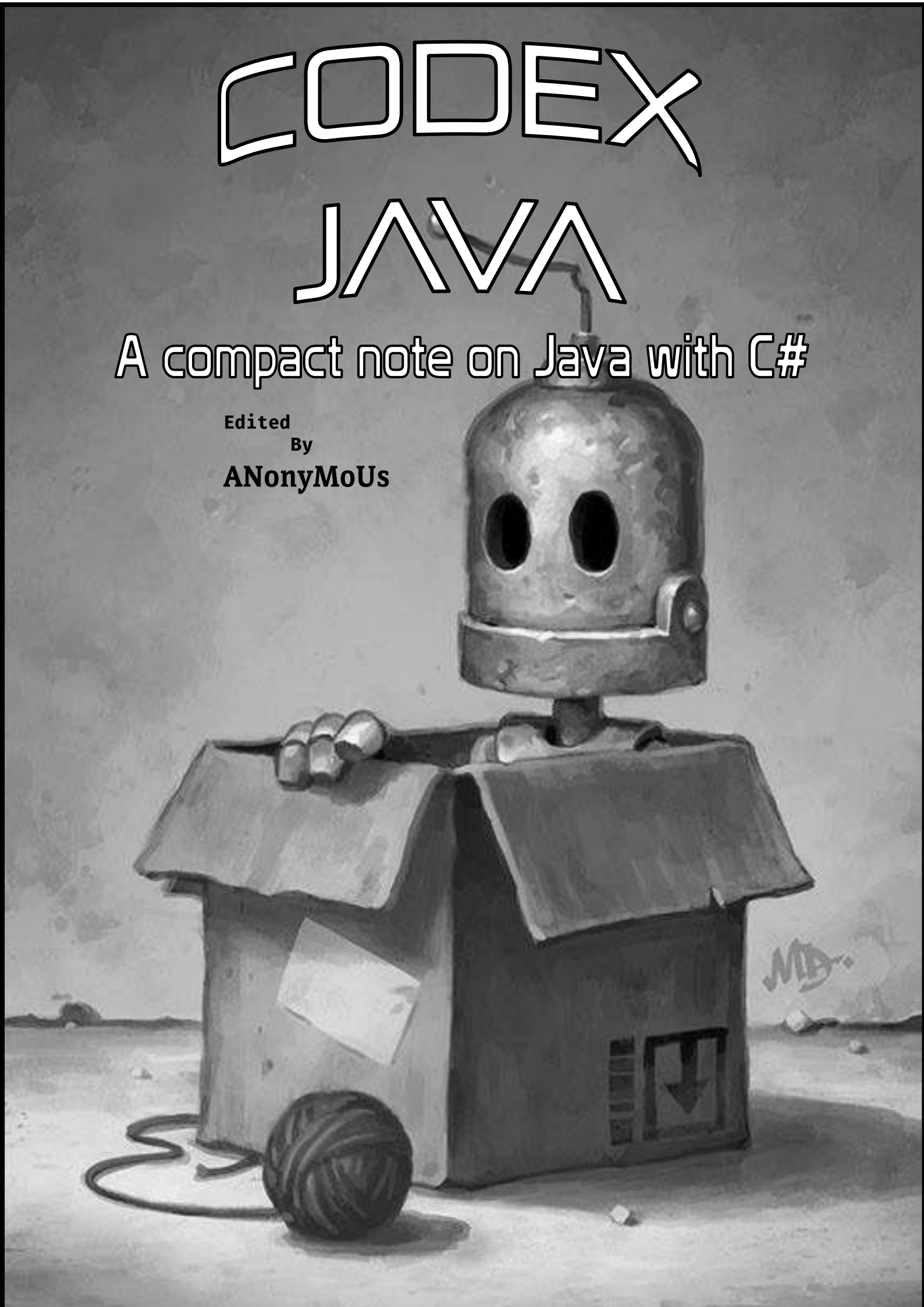
CODEX JAVA

A compact note on Java with C#

Edited
By
ANonyMoUs



CODEX JAVA

A compact note on Java with C#

BASICS OF OOP

PHASE 1 : *Introduction, Data structure, Flow-Control*

Step 1: *INTRODUCTION*

[1] History, compiling/running/Interpretive process
[2] Installing
[3] Compiling a program: Explain steps
[4] Basic Console I/O for First time
[5] Variable declaration and Basic Operators
[6] Keywords
[7] Standard Library

Step 2: *DATA STRUCTURE*

[1] Data types: Primitive/Reference, Implicit
[2] Literals, Escape sequence
[3] Variable initialization
[4] Lifetime & Scope of variables
[5] Operators
[6] Operator Precedence
[7] Type Conversions & casts
[8] Type promotion rules

Step 3: *FLOW-CONTROL*

[1] The if Statement, Nested ifs, The if-else-if Ladder
[2] SWITCH, NESTED SWITCH
[3] For-loop and its variations
[4] While, Do-While & Nested-loops
[5] Continue and Break
[6] goto-lebel Jump

PHASE 2 : *Encapsulation, Inheritance-Polymorphism Abstractions*

Step 4: *ENCAPSULATION*

[1] Class, objects
[2] Reference Variables and Assignment
[3] Methods: returning, parameter, constructor
[4] this reference
[5] Arrays
[6] For-each-loop
[7] Strings
[8] Bitwise Operators
[9] ? ternary Operator
[10] Access Modifiers
[11] Pass Objects to Methods
[12] CALL-BY-VALUE and CALL-BY-REFERENCE
[13] Returning Objects
[14] Overloading : Method/Constructors
[15] Recursion
[16] Static keyword
[17] Nested and Inner Classes
[18] Variable-Length Arguments
[19] MAIN( ) : Returning Values, Passing Arguments
[20] Operator Method: Overloading
[21] Indexers/Properties/Accessor

Step 5: *INHERITANCE-POLYMORPHISM*

[1] Inheritance : base & derived
[2] Constructors and Inheritance
[3] Multilevel Hierarchy
[4] Superclass References and Subclass Objects
[5] Method Overriding, Virtual Method
[6] Final
[7] The Object Class
[8] Structures
[9] Enumerations
[10] Boxing and Unboxing, type wrappers, parsing methods

Step 6: *ABSTRACTIONS*

[1] Abstract Methods and Abstract Classes
[2] Packages (Encapsulation of classes)
[3] Access specifier
[4] Interfaces(more abstraction)
[5] Multiple inheritance and Interface
[6] static Methods in an Interface
[7] Generics
[8] Generic Methods, Constructors, Interfaces, Structures, Delegates
[9] Constrained Types in Generics
[10] Default Value of a Type Parameter
[11] MODULE (Another Abstraction)

PHASE 3 : *Exception-I/O, Advanced-Topics*

Step7: *EXCEPTION-I/O*

[1] Exception Handling basics
[2] try and catch
[3] Throw, Rethrow
[4] Built-in Exceptions
[5] Chained exceptions
[6] User defined Exception
[7] I/O System
[8] Byte Streams, Character Streams, Binary Streams and Pre-defined Streams
[9] Console I/O using BYTE Streams
[10] File I/O using BYTE Streams
[11] Reading from a File
[12] Writing to a File
[13] Automatically Closing a File
[14] Reading and Writing Binary Data
[15] Random-Access Files
[16] Console-based I/O
[17] Reading Characters
[18] Reading Strings
[19] Console Output/writing Using Character Streams
[20] File I/O : FileStream and Byte-Oriented File I/O
[21] File I/O : Character-Based File I/O
[22] Redirecting the Standard Streams
[23] Reading and Writing Binary Data

Step8: *ADVANCED-TOPICS*

[1] Delegates, Events
[2] Anonymous Methods
[3] Events
[4] Namespaces
[5] USING directive

[6] Multithreading
[7] Priorities of Threads
[8] Synchronization
[9] Thread Communication
[10] Suspending, Resuming, and Stopping Threads
[11] Using the Main Thread

[12] Lambda Expression: Parameterized, Block
[13] Functional Interfaces ("FI")
[14] Generic Functional Interfaces
[15] Pass an LE as an Argument
[16] Method References (MRf) and Constructor References

[17] GUI
[18] QUERY: LINQ
[19] Pre-processors, RTTI
[20] STL/Standard Library/ Collections
[21] API