

The Cairo Book

[The Cairo Book](title-page.md)

[Foreword](ch00-01-foreword.md)

[Introduction](ch00-00-introduction.md)

The Cairo Programming Language

Getting Started

- [Getting Started](ch01-00-getting-started.md)

- [Installation](ch01-01-installation.md)

- [Hello, World!](ch01-02-hello-world.md)

Common Programming Concepts

- [Common Programming Concepts](ch02-00-common-programming-concepts.md)

- [Variables and Mutability](ch02-01-variables-and-mutability.md)

- [Data Types](ch02-02-data-types.md)

- [Functions](ch02-03-functions.md)

- [Comments](ch02-04-comments.md)

- [Control Flow](ch02-05-control-flow.md)

Common Collections

- [Common Collections](ch03-00-common-collections.md)

- [Arrays](ch03-01-arrays.md)

- [Dictionaries](ch03-02-dictionaries.md)

Understanding Ownership

- [Understanding Ownership](ch04-00-understanding-ownership.md)

- [What is Ownership?](ch04-01-what-is-ownership.md)

- [References and Snapshots](ch04-02-references-and-snapshots.md)

Using Structs to Structure Related Data

- [Using Structs to Structure Related Data](ch05-00-using-structs-to-structure-related-data.md)

- [Defining and Instantiating Structs](ch05-01-defining-and-instantiating-structs.md)

- [An Example Program Using Structs](ch05-02-an-example-program-using-structs.md)

- [Method Syntax](ch05-03-method-syntax.md)

Enums and Pattern Matching

- [Enums and Pattern Matching](ch06-00-enums-and-pattern-matching.md)

- [Enums](ch06-01-enums.md)

- [The Match Control Flow Construct](ch06-02-the-match-control-flow-construct.md)

- [Concise Control Flow with `if let` and `while let`](ch06-03-concise-control-flow-with-if-let-and-while-let.md)

Managing Cairo Projects with Packages, Crates and Modules

- [Managing Cairo Projects with Packages, Crates and Modules](ch07-00-managing-cairo-projects-with-packages-crates-and-modules.md)

- [Packages and Crates](ch07-01-packages-and-crates.md)

- [Defining Modules to Control Scope](ch07-02-defining-modules-to-control-scope.md)

- [Paths for Referring to an Item in the Module Tree](ch07-03-paths-for-referring-to-an-item-in-the-module-tree.md)

- [Bringing Paths into Scope with the `use` Keyword](ch07-04-bringing-paths-into-scope-with-the-use-keyword.md)

- [Separating Modules into Different Files](ch07-05-separating-modules-into-different-

files.md)

Generic Data Types

- [Generic Types and Traits](ch08-00-generic-types-and-traits.md)
- [Generic Data Types](ch08-01-generic-data-types.md)
- [Traits in Cairo](ch08-02-traits-in-cairo.md)

Error Handling

- [Error Handling](ch09-00-error-handling.md)
- [Unrecoverable Errors with panic](ch09-01-unrecoverable-errors-with-panic.md)
- [Recoverable Errors with Result](ch09-02-recoverable-errors.md)

Testing Cairo Programs

- [Testing Cairo Programs](ch10-00-testing-cairo-programs.md)
- [How To Write Tests](ch10-01-how-to-write-tests.md)
- [Testing Organization](ch10-02-test-organization.md)

Advanced Cairo Features

- [Advanced Cairo Features](ch11-00-advanced-features.md)
- [Custom Data Structures](ch11-01-custom-data-structures.md)
- [Smart Pointers](ch11-02-smart-pointers.md)
- [Deref Coercion](ch11-09-deref-coercion.md)
- [Operator Overloading](ch11-03-operator-overloading.md)
- [Working with Hashes](ch11-04-hash.md)
- [Macros](ch11-05-macros.md)
- [Inlining in Cairo](ch11-06-inlining-in-cairo.md)
- [Printing](ch11-08-printing.md)
- [Appendix (Cairo)](appendix-00.md)
- [A - Keywords](appendix-01-keywords.md)
- [B - Operators and Symbols](appendix-02-operators-and-symbols.md)
- [C - Derivable Traits](appendix-03-derivable-traits.md)
- [D - The Cairo Prelude](appendix-04-cairo-prelude.md)
- [E - Common Error Messages](appendix-05-common-error-messages.md)
- [F - Useful Development Tools](appendix-06-useful-development-tools.md)

Smart Contracts in Cairo

Introduction to Starknet Smart Contracts

- [Introduction to Starknet Smart Contracts](./ch13-00-introduction-to-starknet-smart-contracts.md)
- [General Introduction to Smart Contracts](./ch13-01-general-introduction-to-smart-contracts.md)
- [Anatomy of a Simple Contract](./ch13-02-anatomy-of-a-simple-contract.md)

Building Starknet Smart Contracts

- [Building Starknet Smart Contracts](./ch14-00-building-starknet-smart-contracts.md)
- [Contract Storage](./ch14-01-00-contract-storage.md)
- [Storage Mappings](./ch14-01-01-storage-mappings.md)
- [Storage Vecs](./ch14-01-02-storage-vecs.md)
- [Contract Functions](./ch14-02-contract-functions.md)
- [Contract Events](./ch14-03-contract-events.md)

Starknet Cross-Contract Interactions

- [Starknet Contract Interactions](./ch15-00-starknet-contract-interactions.md)
- [Contract Class ABI](./ch15-01-contract-class-abi.md)
- [Interacting with Another Contract](./ch15-02-interacting-with-another-contract.md)
- [Executing Code from Another Class](./ch15-03-executing-code-from-another-class.md)

Building Advanced Starknet Smart Contracts

- [Building Advanced Starknet Smart Contracts](./ch16-00-building-advanced-starknet-smart-contracts.md)
- [Optimizing Storage Costs](./ch16-01-optimizing-storage-costs.md)
- [Composability and Components](./ch16-02-00-composability-and-components.md)
 - [Under the Hood](./ch16-02-01-under-the-hood.md)
 - [Component Dependencies](./ch16-02-02-component-dependencies.md)
 - [Testing Components](./ch16-02-03-testing-components.md)
- [Upgradeability](./ch16-03-upgradeability.md)
- [L1 <> L2 Messaging](./ch16-04-L1-L2-messaging.md)
- [Oracle Interactions](./ch16-05-oracle-interactions.md)
 - [Price Feeds](./ch16-05-01-price-feeds.md)
 - [Randomness](./ch16-05-02-randomness.md)
- [Other Examples](./ch16-06-00-other-examples.md)
 - [Deploying and Interacting with a Voting Contract](./ch16-06-01-deploying-and-interacting-with-a-voting-contract.md)

Starknet Smart Contracts Security

- [Starknet Smart Contracts Security](./ch17-00-starknet-smart-contracts-security.md)
- [General Recommendations](./ch17-01-general-recommendations.md)
- [Testing Smart Contracts](./ch17-02-testing-smart-contracts.md)
- [Static Analysis Tools](./ch17-03-static-analysis-tools.md)
- [Formal Verification]()

Appendix

- [Appendix (Starknet)](appendix-000.md)
- [A - System Calls](appendix-08-system-calls.md)

Cairo VM

- [Introduction](ch200-introduction.md)
- [Architecture]()

Memory

- [Memory]()
 - [Non-Deterministic Read-only Memory]()
 - [Segments]()
 - [Segment Value]()
 - [Relocation]()
 - [Layout]()

Execution Model

- [Execution Model]()
 - [Registers]()
 - [Instructions]()
 - [Cairo Assembly (CASM)]()

- [State transition]()
- ## Builtins
 - [Builtins](ch204-00-builtins.md)
 - [How Builtins Work](ch204-01-how-builtins-work.md)
 - [Builtins List](ch204-02-builtins-list.md)
 - [Output]()
 - [Pedersen](ch204-02-01-pedersen.md)
 - [Range Check]()
 - [ECDSA]()
 - [Bitwise]()
 - [EC OP]()
 - [Keccak]()
 - [Poseidon]()
 - [Range Check96]()
 - [AddMod]()
 - [MulMod]()
 - [Segment Arena]()
 - [Gas]()
 - [System]()
- ## Hints
 - [Hints]()
 - [Structure]()
 - [Hint runner]()
 - [List of hints]()
- ## Runner
 - [Runner]()
 - [Program]()
 - [Program Artifacts]()
 - [Program Parsing]()
 - [Runner Mode]()
 - [Execution Mode]()
 - [Proof Mode]()
 - [Output]()
 - [Cairo PIE]()
 - [Memory File]()
 - [Trace file]()
 - [AIR public input]()
 - [AIR private input]()
 - [Tracer]()
 - [Implementations]()
 - [Resources]()