Structure of a contract

A contract is a collection of persistentstate variables , andfunctions which may manipulate these variables. Functions and state variables within a contract's scope are said to belong to that contract. A contract can only access and modify its own state. If a contract wishes to access or modify another contract's state, it must make a call to an external function of the other contract. For anything to happen on the Aztec network, an external function of a contract needs to be called.

Contract

A contract may be declared and given a name using thecontract keyword (see snippet below). By convention, contracts are named inPascalCase .

contract keyword contract MyContract

{

// Imports

// Storage

// Functions } A note for vanilla Noir devs There is nomain() function within a Noircontract scope. More than one function can be an entrypoint.

Directory structure

Here's a common layout for a basic Aztec.nr Contract project:

——— my_aztec_contract_project ├——— src | ├——— main.nr <-- your contract └——— Nargo.toml <-- package and dependency management * See the vanilla Noir docs formore info on packages * . * You can review the structure of a complete contract in the token contract tutorialhere * . Edit this page

Previous How to setup a new contract projectNext What a contract looks like