

# Smart Contracts

This section is a collection of how-to guides and references for building smart contracts with Aztec.nr.

If you are looking for an overview of how smart contracts work, head to the [Concepts section](#).

## What is Aztec.nr?

Aztec.nr is a framework for writing Aztec smart contracts.

## Nomenclature

[Noir](#) is a domain specific language for creating and verifying proofs. Its design choices are influenced heavily by Rust.

A smart contract is just a collection of persistent state variables, and a collection of functions which may edit those state variables.

An Aztec smart contract is a smart contract with private state variables and private functions.

Aztec.nr is a framework for writing Aztec smart contracts, written in Noir.

Further Nomenclature can be found [here](#).

## Getting started

### Install aztec-nargo

To write an Aztec.nr contract, you need the compiler, aztec-nargo which is installed when you install the sandbox. See install instructions [here](#).

For those coming from vanilla Noir, the version used for aztec.nr is tracked separately to nargo for vanilla Noir. Be sure to use aztec-nargo to compile your contracts.

### Install Noir LSP (recommended)

Install the [Noir Language Support extension](#) to get syntax highlighting, syntax error detection and go-to definitions for your Aztec contracts.

Once the extension is installed, check your nargo binary by hovering over Nargo in the status bar on the bottom right of the application window. Click to choose the path to aztec-nargo (or regular nargo, if you have that installed).

You can print the path of your aztec-nargo executable by running:

which aztec-nargo To specify a custom nargo executable, go to the VSCode settings and search for "noir", or click extension settings on the noir-lang LSP plugin. Update the Noir: Nargo Path field to point to your desired aztec-nargo executable.

### Install Noir tooling

There are a number of tools to make writing Aztec.nr contracts in Noir more pleasant. See [here](#).

## Tutorials

See the [Private Voting tutorial](#) for more info on getting set up to write contracts.

## Learn more

[Edit this page](#)

[Previous Private Execution Environment \(PXE\)](#) [Next How to setup a new contract project](#)