Visibility

In Aztec there are multiple different types of visibility that can be applied to functions. Namely we havedata visibility andfunction visibility . This page explains these types of visibility.

For a practical guide of using multiple types of data and function visibility,follow thetoken tutorial .

Data Visibility

Data visibility is used to describe whether the data (or state) used in a function is generally accessible (public) or on a need to know basis (private).

Function visibility

This is the kind of visibility you are more used to seeing in Solidity and more traditional programming languages. It is used to describe whether a function is callable from other contracts, or only from within the same contract.

By default, all functions are callable from other contracts, similarly to the Soliditypublic visibility. To make them only callable from the contract itself, you can mark them asinternal . Contrary to solidity, we don't have theexternal norprivate keywords.external since it is limited usage when we don't support inheritance, andprivate since we don't support inheritance and it would also be confusing with multiple types ofprivate .

A good place to useinternal is when you want a private function to be able to alter public state. As mentioned above, private functions cannot do this directly. They are able to call public functions and by making these internal we can ensure that this state manipulating function is only callable from our private function.

danger Note that non-internal functions could be used directly as an entry-point, which currently means that themsg_sender would be0 , so for now, using address0 as a burn address is not recommended. You can learn more about this in theAccounts concept page . To understand how visibility works under the hood, check out theInner Workings page . Edit this page

Previous Public, Private, and Unconstrained Functions Next How to call functions from other functions