

## # Appendix D - The Cairo Prelude

### ## Prelude

The Cairo prelude is a collection of commonly used modules, functions, data types, and traits that are automatically brought into scope of every module in a Cairo crate without needing explicit import statements. Cairo's prelude provides the basic building blocks developers need to start Cairo programs and writing smart contracts.

The core library prelude is defined in the `_[lib.cairo]`(<https://github.com/starkware-libs/cairo/blob/main/corelib/src/lib.cairo>)\_

file of the `corelib` crate and contains Cairo's primitive data types, traits, operators, and utility functions. This includes:

- Data types: integers, bools, arrays, dicts, etc.
- Traits: behaviors for arithmetic, comparison, and serialization operations
- Operators: arithmetic, logical, bitwise
- Utility functions - helpers for arrays, maps, boxing, etc.

The core library prelude delivers the fundamental programming constructs and operations needed for basic Cairo programs, without requiring the explicit import of elements. Since the core library prelude is automatically imported, its contents are available for use in any Cairo crate without explicit imports. This prevents repetition and provides a better devX. This is what allows you to use ``ArrayTrait::append()`` or the ``Default`` trait without bringing them explicitly into scope.

You can choose which prelude to use. For example, adding ``edition = "2024_07"`` in the `_Scarb.toml_` configuration file will load the prelude from July 2024. Note that when you create a new project using ``scarb new`` command, the `_Scarb.toml_` file will automatically include ``edition = "2024_07"``.

Different prelude versions will expose different functions and traits, so it is important to specify the correct edition in the `_Scarb.toml_` file. Generally, you want to start a new project using the latest edition, and migrate to newer editions as they are released.