

# Panic

Code in Cairo 1.0 may panic - which means it may fail with an unrecoverable error - meaning it is impossible to catch and handle. When the program panics, using the [linear type system](#) all living variables on the stack would be dropped or otherwise destructed, which makes sure the run remains provable and valid.

## panic function

The basic function that all panic stems from is the `panic` function. It is defined as: `extern fn panic(data: Array) -> never;` The `panic` function takes a single argument, which is a `felt252` array. This array is the data that is passed as the reason the run panicked. The `panic` function never returns, and is marked as such with the [never type](#).

## nopanic notation

Functions may be marked with the `nopanic` notation. This means that the function will never panic. This can be useful for writing code that may never fail. Only `nopanic` functions may be called from an `nopanic` function.

## nopanic and traits

If a trait function is marked with `nopanic`, all implementations of that trait must also be marked with `nopanic`, as the trait function may be called from an `nopanic` function. An example for such a function is `Destruct` trait `destruct` function, which is `nopanic` as it is called during panic handling, see [linear type system](#) for more info. If a trait function is not marked with `nopanic`, all implementations of that trait may be marked with `nopanic` or not. An example for such a function is `Add` trait `add` function, which is `nopanic` for `felt252` addition, but isn't so for integer addition.

## panic\_with macro

A function returning an `Option` or `Result` may be marked with the `panic_with` macro. This macro takes two arguments, which is the data that is passed as the panic reason as well as the name for a wrapping function. If the function returns `None` or `Err`, `panic` function will be called with the given data.

# [panic\_with('got none value', unwrap)]

`fn identity(value: Option) -> Option { a }` Some `fn unwrap(value: Option) -> u128` that internally may panic may be created.

[9.4 Inference Memory model](#) §