

**NAME**

Error::TypeTiny – exceptions for Type::Tiny and friends

**SYNOPSIS**

```
use Data::Dumper;
use Try::Tiny;
use Types::Standard qw(Str);

try {
    Str->assert_valid(undef);
}
catch {
    my $exception = shift;
    warn "Encountered Error: $exception";
    warn Dumper($exception->explain)
        if $exception->isa("Error::TypeTiny::Assertion");
};
```

**STATUS**

This module is covered by the Type-Tiny stability policy.

**DESCRIPTION**

When Type::Tiny and its related modules encounter an error, they throw an exception object. These exception objects inherit from Error::TypeTiny.

**Constructors**

`new(%attributes)`

Moose-style constructor function.

`throw(%attributes)`

Constructs an exception and passes it to die.

Automatically populates `context` and `stack_trace` if appropriate.

**Attributes**

`message`

The error message.

`context`

Hashref containing the package, file and line that generated the error.

`stack_trace`

A more complete stack trace. This feature requires `Devel::StackTrace`; use the `$StackTrace` package variable to switch it on.

**Methods**

`to_string`

Returns the message, followed by the context if it is set.

**Functions**

`Error::TypeTiny::croak($format, @args)`

Functional-style shortcut to `throw` method. Takes an `sprintf`-style format string and optional arguments to construct the message.

**Overloading**

- Stringification is overloaded to call `to_string`.

**Package Variables**

`%Error::TypeTiny::CarpInternal`

Serves a similar purpose to `%Carp::CarpInternal`.

`$Error::TypeTiny::StackTrace`  
Boolean to toggle stack trace generation.

`$Error::TypeTiny::LastError`  
A reference to the last exception object thrown.

## CAVEATS

Although `Error::TypeTiny` objects are thrown for errors produced by `Type::Tiny`, that doesn't mean every time you use `Type::Tiny` you'll get `Error::TypeTinys` whenever you want.

For example, if you use a `Type::Tiny` type constraint in a Moose attribute, Moose will not call the constraint's `assert_valid` method (which throws an exception). Instead it will call `check` and `get_message` (which do not), and will confess an error message of its own. (The `$LastError` package variable may save your bacon.)

## BUGS

Please report any bugs to <http://rt.cpan.org/Dist/Display.html?Queue=Type-Tiny>.

## SEE ALSO

`Error::TypeTiny::Assertion`, `Error::TypeTiny::WrongNumberOfParameters`.

`Try::Tiny`, `Try::Tiny::ByClass`.

## AUTHOR

Toby Inkster <[tobyink@cpan.org](mailto:tobyink@cpan.org)>.

## COPYRIGHT AND LICENCE

This software is copyright (c) 2013–2014, 2017–2019 by Toby Inkster.

This is free software; you can redistribute it and/or modify it under the same terms as the Perl 5 programming language system itself.

## DISCLAIMER OF WARRANTIES

THIS PACKAGE IS PROVIDED “AS IS” AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.