

NAME

Type::Parser – parse type constraint strings

SYNOPSIS

```
use v5.10;
use strict;
use warnings;

use Type::Parser qw( eval_type );
use Type::Registry;

my $reg = Type::Registry->for_me;
$reg->add_types("Types::Standard");

my $type = eval_type("Int | ArrayRef[Int]", $reg);

$type->check(10);          # true
$type->check([1..4]);      # true
$type->check({foo=>1});    # false
```

STATUS

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

DESCRIPTION

Generally speaking, you probably don't want to be using this module directly. Instead use the `lookup` method from `Type::Registry` which wraps it.

Functions

`parse($string)`

Parse the type constraint string into something like an AST.

If called in list context, also returns any “tail” found on the original string.

`extract_type($string, $registry)`

Compile a type constraint string into a `Type::Tiny` object.

If called in list context, also returns any “tail” found on the original string.

`eval_type($string, $registry)`

Compile a type constraint string into a `Type::Tiny` object.

Throws an error if the “tail” contains any non-whitespace character.

Constants

The following constants correspond to values returned by `$token->type`.

```
TYPE
QUOTELIKE
STRING
CLASS
L_BRACKET
R_BRACKET
COMMA
SLURPY
UNION
INTERSECT
NOT
L_PAREN
R_PAREN
```

MYSTERY

BUGS

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

SEE ALSO

Type::Registry.

AUTHOR

Toby Inkster <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.