

# **Issue 318 Documentation**

**version 0.0**

**Roberto Alsina**

April 29, 2018



# Contents

<b>Welcome to Issue 318's documentation!</b>	<b>1</b>
<b>Indices and tables</b>	<b>1</b>
<b>Index</b>	<b>3</b>
<b>Python Module Index</b>	<b>5</b>



# Welcome to Issue 318's documentation!

Contents:

`bool namespaced::theclass::method(int arg1, std::string arg2)`

Describes a method with parameters and types.

`bool namespaced::theclass::method(arg1, arg2)`

Describes a method without types.

`template<>const T &array<T>::operator[]() const`

Describes the constant indexing operator of a templated array.

`operator bool() const`

Describe a casting operator here.

`std::string theclass::name`

`type theclass::const_iterator`

`format_exception(etype, value, tb[, limit=None])`

Format the exception with a traceback.

**Parameters:**

- **etype** – exception type
- **value** – exception value
- **tb** – traceback object
- **limit** (*integer or None*) – maximum number of stack frames to show

**Return type:** list of strings

`parrot.spam(eggs)`

`parrot.ham(eggs)`

Spam or ham the foo.

## Indices and tables

- [genindex](#)
- [modindex](#)
- [search](#)



# Index

## *A*

`array<T>::operator[]` (C++ function)

## *F*

`format_exception()` (built-in function)

## *H*

`ham()` (in module `parrot`)

## *N*

`namespaced::theclass::method` (C++ function) [1]

## *O*

`operator bool` (C++ function)

## *P*

`parrot` (module)

## *S*

`spam()` (in module `parrot`)

## *T*

`theclass::const_iterator` (C++ type)

`theclass::name` (C++ member)





# Python Module Index

## *p*

### **parrot** (Unix, Windows)

Analyze and reanimate dead parrots.