

## Lab 6

Generated by Doxygen 1.8.16



<b>1 Tina's Burger Bistro</b>	<b>1</b>
1.1 Overall Design Architecture	1
1.2 Testing	2
1.3 Building Tina's Burger Bistro	2
1.4 Notable Concepts Used	2
<b>2 Todo List</b>	<b>3</b>
<b>3 Hierarchical Index</b>	<b>5</b>
3.1 Class Hierarchy	5
<b>4 Class Index</b>	<b>9</b>
4.1 Class List	9
<b>5 Class Documentation</b>	<b>13</b>
5.1 Catch::Detail::Approx Class Reference	13
5.1.1 *	13
5.1.2 *	13
5.1.3 *	14
5.2 Catch::AssertionHandler Class Reference	14
5.2.1 *	14
5.3 Catch::AssertionInfo Struct Reference	15
5.3.1 *	15
5.4 Catch::AssertionReaction Struct Reference	15
5.4.1 *	15
5.5 Catch::AutoReg Struct Reference	15
5.5.1 *	15
5.6 Catch::BenchmarkLooper Class Reference	16
5.6.1 *	16
5.7 Catch::BinaryExpr< LhsT, Rhst > Class Template Reference	16
5.7.1 *	16
5.7.2 *	16
5.8 Catch::Matchers::StdString::CasedString Struct Reference	16
5.8.1 *	16
5.8.2 *	17
5.9 Catch::CaseSensitive Struct Reference	17
5.9.1 *	17
5.10 Cashier Class Reference	17
5.10.1 *	17
5.10.2 Detailed Description	17
5.10.3 Member Function Documentation	18
5.10.3.1 serve_customer()	18
5.11 Catch_global_namespace_dummy Struct Reference	18
5.12 Catch::Matchers::Vector::ContainsElementMatcher< T > Struct Template Reference	19

5.12.1 *	19
5.12.2 *	19
5.12.3 *	19
5.13 Catch::Matchers::StdString::ContainsMatcher Struct Reference	19
5.13.1 *	19
5.13.2 *	20
5.14 Catch::Matchers::Vector::ContainsMatcher< T > Struct Template Reference	20
5.14.1 *	20
5.14.2 *	20
5.14.3 *	20
5.15 Cook Class Reference	21
5.15.1 *	21
5.15.2 Detailed Description	21
5.15.3 Member Function Documentation	21
5.15.3.1 prepare_dish()	21
5.16 Catch::Counts Struct Reference	22
5.16.1 *	22
5.16.2 *	22
5.17 Customer Class Reference	22
5.17.1 *	22
5.17.2 *	23
5.17.3 Detailed Description	23
5.17.4 Member Function Documentation	23
5.17.4.1 charge_money()	23
5.17.4.2 expel()	23
5.17.4.3 get_order()	23
5.17.4.4 refund_money()	24
5.18 Catch::Decomposer Struct Reference	24
5.18.1 *	24
5.19 Catch::Matchers::StdString::EndsWithMatcher Struct Reference	24
5.19.1 *	24
5.19.2 *	25
5.20 Catch::Matchers::StdString::EqualsMatcher Struct Reference	25
5.20.1 *	25
5.20.2 *	25
5.21 Catch::Matchers::Vector::EqualsMatcher< T > Struct Template Reference	25
5.21.1 *	25
5.21.2 *	26
5.21.3 *	26
5.22 Catch::ExceptionTranslatorRegistrar Class Reference	26
5.22.1 *	26
5.23 Catch::ExprLhs< LhsT > Class Template Reference	26

5.23.1 *	26
5.24 Catch::IExceptionTranslator Struct Reference	27
5.24.1 *	27
5.25 Catch::IExceptionTranslatorRegistry Struct Reference	27
5.25.1 *	27
5.26 Catch::IMutableRegistryHub Struct Reference	27
5.26.1 *	27
5.27 Catch::IRegistryHub Struct Reference	28
5.27.1 *	28
5.28 Catch::IResultCapture Struct Reference	28
5.28.1 *	28
5.29 Catch::IRunner Struct Reference	29
5.29.1 *	29
5.30 Catch::is_range< T > Struct Template Reference	29
5.30.1 *	29
5.30.2 Member Data Documentation	29
5.30.2.1 value	29
5.31 Catch::Detail::IsStreamInsertable< T > Class Template Reference	29
5.31.1 *	29
5.32 Catch::IStream Struct Reference	30
5.32.1 *	30
5.33 Catch::ITestCaseRegistry Struct Reference	30
5.33.1 *	30
5.34 Catch::ITestInvoker Struct Reference	30
5.34.1 *	30
5.35 Catch::ITransientExpression Struct Reference	31
5.35.1 *	31
5.35.2 *	31
5.36 Kitchen Class Reference	31
5.36.1 *	31
5.36.2 Detailed Description	32
5.36.3 Member Function Documentation	32
5.36.3.1 prepare_dish()	32
5.37 Catch::LazyExpression Class Reference	32
5.37.1 *	32
5.37.2 *	33
5.38 Catch::Matchers::Impl::MatchAllOf< ArgT > Struct Template Reference	33
5.38.1 *	33
5.38.2 *	33
5.38.3 *	33
5.39 Catch::Matchers::Impl::MatchAnyOf< ArgT > Struct Template Reference	34
5.39.1 *	34

5.39.2 *	34
5.39.3 *	34
5.40 Catch::Matchers::Impl::MatcherBase< T > Struct Template Reference	34
5.40.1 *	35
5.40.2 *	35
5.41 Catch::Matchers::Impl::MatcherMethod< ObjectT > Struct Template Reference	35
5.41.1 *	35
5.42 Catch::Matchers::Impl::MatcherMethod< PtrT * > Struct Template Reference	35
5.42.1 *	35
5.43 Catch::Matchers::Impl::MatcherUntypedBase Class Reference	36
5.43.1 *	36
5.43.2 *	36
5.43.3 *	36
5.44 Catch::MatchExpr< ArgT, MatcherT > Class Template Reference	36
5.44.1 *	36
5.44.2 *	37
5.45 Catch::Matchers::Impl::MatchNotOf< ArgT > Struct Template Reference	37
5.45.1 *	37
5.45.2 *	37
5.45.3 *	37
5.46 Catch::MessageBuilder Struct Reference	38
5.46.1 *	38
5.46.2 *	38
5.47 Catch::MessageInfo Struct Reference	38
5.47.1 *	38
5.47.2 *	38
5.48 Catch::MessageStream Struct Reference	39
5.48.1 *	39
5.48.2 *	39
5.49 Catch::NameAndTags Struct Reference	39
5.49.1 *	39
5.49.2 *	39
5.50 Catch::NonCopyable Class Reference	40
5.51 Catch::not_this_one Struct Reference	40
5.52 Order Class Reference	40
5.52.1 *	40
5.52.2 *	40
5.52.3 Detailed Description	41
5.53 Catch::pluralise Struct Reference	41
5.53.1 *	41
5.53.2 *	41
5.53.3 *	41

5.54 <a href="#">Catch::Matchers::StdString::RegexMatcher Struct Reference</a>	41
5.54.1 *	41
5.54.2 *	42
5.55 <a href="#">Catch::RegistrarForTagAliases Struct Reference</a>	42
5.55.1 *	42
5.56 <a href="#">Catch::ResultDisposition Struct Reference</a>	42
5.56.1 *	42
5.57 <a href="#">Catch::ResultWas Struct Reference</a>	42
5.57.1 *	42
5.58 <a href="#">Catch::ReusableStringStream Class Reference</a>	43
5.58.1 *	43
5.58.2 *	43
5.59 <a href="#">Catch::ScopedMessage Class Reference</a>	43
5.59.1 *	43
5.59.2 *	43
5.60 <a href="#">Catch::Section Class Reference</a>	44
5.60.1 *	44
5.61 <a href="#">Catch::SectionEndInfo Struct Reference</a>	44
5.61.1 *	44
5.61.2 *	44
5.62 <a href="#">Catch::SectionInfo Struct Reference</a>	45
5.62.1 *	45
5.62.2 *	45
5.63 <a href="#">Catch::SourceLineInfo Struct Reference</a>	45
5.63.1 *	45
5.63.2 *	45
5.64 <a href="#">Catch::Matchers::StdString::StartsWithMatcher Struct Reference</a>	46
5.64.1 *	46
5.64.2 *	46
5.65 <a href="#">Catch::StreamEndStop Struct Reference</a>	46
5.65.1 *	46
5.66 <a href="#">Catch::StringMaker&lt; T, typename &gt; Struct Template Reference</a>	47
5.66.1 *	47
5.67 <a href="#">Catch::StringMaker&lt; bool &gt; Struct Template Reference</a>	47
5.67.1 *	47
5.68 <a href="#">Catch::StringMaker&lt; Catch::Detail::Approx &gt; Struct Template Reference</a>	47
5.68.1 *	47
5.69 <a href="#">Catch::StringMaker&lt; char * &gt; Struct Template Reference</a>	48
5.69.1 *	48
5.70 <a href="#">Catch::StringMaker&lt; char &gt; Struct Template Reference</a>	48
5.70.1 *	48
5.71 <a href="#">Catch::StringMaker&lt; char const * &gt; Struct Template Reference</a>	48

5.71.1 *	48
5.72 Catch::StringMaker< char[SZ]> Struct Template Reference	48
5.72.1 *	48
5.73 Catch::StringMaker< double > Struct Template Reference	49
5.73.1 *	49
5.74 Catch::StringMaker< float > Struct Template Reference	49
5.74.1 *	49
5.75 Catch::StringMaker< int > Struct Template Reference	49
5.75.1 *	49
5.76 Catch::StringMaker< long > Struct Template Reference	49
5.76.1 *	49
5.77 Catch::StringMaker< long long > Struct Template Reference	50
5.77.1 *	50
5.78 Catch::StringMaker< R C::* > Struct Template Reference	50
5.78.1 *	50
5.79 Catch::StringMaker< R, typename std::enable_if< is_range< R >::value &&!::Catch::Detail::IsStreamInsertable< R >::value >::type > Struct Template Reference	50
5.79.1 *	50
5.80 Catch::StringMaker< signed char > Struct Template Reference	51
5.80.1 *	51
5.81 Catch::StringMaker< signed char[SZ]> Struct Template Reference	51
5.81.1 *	51
5.82 Catch::StringMaker< std::nullptr_t > Struct Template Reference	51
5.82.1 *	51
5.83 Catch::StringMaker< std::string > Struct Template Reference	51
5.83.1 *	51
5.84 Catch::StringMaker< std::wstring > Struct Template Reference	52
5.84.1 *	52
5.85 Catch::StringMaker< T * > Struct Template Reference	52
5.85.1 *	52
5.86 Catch::StringMaker< T[SZ]> Struct Template Reference	52
5.86.1 *	52
5.87 Catch::StringMaker< unsigned char > Struct Template Reference	52
5.87.1 *	52
5.88 Catch::StringMaker< unsigned char[SZ]> Struct Template Reference	53
5.88.1 *	53
5.89 Catch::StringMaker< unsigned int > Struct Template Reference	53
5.89.1 *	53
5.90 Catch::StringMaker< unsigned long > Struct Template Reference	53
5.90.1 *	53
5.91 Catch::StringMaker< unsigned long long > Struct Template Reference	53
5.91.1 *	53



5.92 Catch::StringMaker< wchar_t * > Struct Template Reference	54
5.92.1 *	54
5.93 Catch::StringMaker< wchar_t const * > Struct Template Reference	54
5.93.1 *	54
5.94 Catch::Matchers::StdString::StringMatcherBase Struct Reference	54
5.94.1 *	54
5.94.2 *	54
5.94.3 *	55
5.95 Catch::StringRef Class Reference	55
5.95.1 *	55
5.95.2 *	55
5.95.3 *	55
5.95.4 Detailed Description	56
5.96 SupplyRunner Class Reference	56
5.96.1 *	56
5.96.2 Detailed Description	56
5.96.3 Member Function Documentation	56
5.96.3.1 get_ingredients()	56
5.97 Catch::TestCase Class Reference	57
5.97.1 *	57
5.97.2 *	57
5.98 Catch::TestCaseInfo Struct Reference	57
5.98.1 *	57
5.98.2 *	58
5.98.3 *	58
5.98.4 *	58
5.99 Catch::TestFailureException Struct Reference	58
5.100 Catch::TestInvokerAsMethod< C > Class Template Reference	59
5.100.1 *	59
5.101 Catch::Timer Class Reference	59
5.101.1 *	59
5.102 Catch::Totals Struct Reference	59
5.102.1 *	59
5.102.2 *	60
5.103 Catch::UnaryExpr< LhsT > Class Template Reference	60
5.103.1 *	60
5.103.2 *	60
5.104 Catch::Matchers::Vector::UnorderedEqualsMatcher< T > Struct Template Reference	60
5.104.1 *	60
5.104.2 *	61
5.105 Catch::Matchers::Floating::WithinAbsMatcher Struct Reference	61
5.105.1 *	61

5.105.2 *	61
5.106 Catch::Matchers::Floating::WithinUlpMatcher Struct Reference	61
5.106.1 *	61
5.106.2 *	62
<b>Index</b>	<b>63</b>

# Chapter 1

## Tina's Burger Bistro

This project is meant to simulate a fast food bistro. It is also meant to demonstrate what a fairly complete EE205 project should look like at the end.

...if you want to look at this file in a nicer format, look into "How to render Markdown to PDF on Linux". You'll probably end up using `pandoc` in order to do this.

Also, if you can get and use `xdg-open`, it can be useful for opening `.pdf` files from the terminal.

### 1.1 Overall Design Architecture

The bistro's design is formed by the following classes:

- `Customer`
- `Cashier`
- `Cook`
- `Kitchen`
- `Storeroom` (`std::map<std::string, std::map<Ingredient, std::size_t>>`)
- `SupplyRunner`
- `Order`
- `Dish` (`enum class : int`)
- `Ingredient` (`std::string`)

Customers are put onto a `std::queue<Customer>`. They are handled by a `Cashier`, who gives them a unique ID (starting from 0, counting up by 1 each time), takes their order, and charges them money. If they give invalid orders or cannot pay, they are expelled from the restaurant. After the `Cashier` takes a `Customer`'s order, the `Customer`'s ID is sent along with the ordered items in an `Order` object and put onto an `Order` stack (`std::stack<Order>`). Then, the `Cook` is meant to interface with the stack.

The `Cook` takes Orders off of the stack, and then looks it up in the `RecipeBook` (a map from `std::string` dish names to `std::map<Ingredient, std::size_t>`). Then, it asks the `SupplyRunner` to get the required ingredients, and uses the `Kitchen` to prepare the dish with the returned ingredients. After preparing the dish, the `Cook` will send it along with the correct customer ID as a `std::pair<std::size_t, Dish>` to the output

```
std::queue<std::pair<std::size_t, dish>. (Dish, by the way, is simply an enum class ↵  
: int).
```

The **SupplyRunner** is meant to be configured with a **Storeroom**, which holds ingredients and how many of that type of ingredient it currently has. When asked for a number of ingredients, the **SupplyRunner** will do its best and check whether the **Storeroom** has enough ingredients to supply. If it doesn't, the **SupplyRunner** will throw a `const char*` exception – otherwise, it will provide the correct ingredients as a `std::vector<Ingredient>`. Note that it is not required to return the ingredients in any specified order – just that the correct ingredients are returned in the vector in the correct overall amounts.

## 1.2 Testing

In this project, we use **Catch2** as our testing framework – a single-header library that is put into our `./dep/` directory. Thus, we do not need a linking phase to link the testing library in.

Our test cases live in the `./tst/` directory, with the test case suites prefixed by `test-`. We also have 3 examples files we created that are prefixed by `example-`.

## 1.3 Building Tina's Burger Bistro

Building the project is handled by our **Makefile** in the root directory of our project. One need only run `make` to build every executable, even the testing ones!

However, you can also run specialized rules to create the test case suites for only a certain class, such as `make runner` to create `./bin/test-SupplyRunner.out`. Inspect the **Makefile** manually if you wish for more detail.

## 1.4 Notable Concepts Used

**Doxygen** is a documentation tool that can be used to turn specially formatted comments into full-blown documentation for a project (one I like to use is HTML format so I can open it as a web page). It has support for C++. Every single header file in `./include/` is commented in the style required for Doxygen. Note that this is not required to build the project, but it is a nice addon if you wish to generate more readable documentaiton than just comments in your source files and have it unified in one place. It is a fairly common tool that you may see in the future on some larger projects.

**Dependency injection** is a concept where instead of you directly including objects in a class that are default constructed, instead, you take any objects you rely on in the constructor and save references, copies, or moved versions of them. The reason to do this is to allow yourself to use "mock" versions of dependency objects if the finished versions of other classes you are working on are not done yet. This is a relatively common OOP design pattern seen in production. Another related term is "inversion of control."

**Unit testing** is a form of program testing where you break a program into its part and test them individually in order to assume that the composition of small, verified parts created a safe, verified overall program. In OOP, the units of testing are usually classes – therefore, in this project, we provide the 3 test case suites that you need to pass in order to pass this lab.

## Chapter 2

## Todo List

### Member `Customer::expel ()`

Revise the design of expellation and the unit test cases so expellation can be unit tested without this global variable counter hack.



## Chapter 3

# Hierarchical Index

### 3.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Catch::Detail::Approx . . . . .	13
Catch::AssertionHandler . . . . .	14
Catch::AssertionInfo . . . . .	15
Catch::AssertionReaction . . . . .	15
Catch::BenchmarkLooper . . . . .	16
Catch::Matchers::StdString::CasedString . . . . .	16
Catch::CaseSensitive . . . . .	17
Cashier . . . . .	17
Catch_global_namespace_dummy . . . . .	18
Cook . . . . .	21
Catch::Counts . . . . .	22
Customer . . . . .	22
Catch::Decomposer . . . . .	24
Catch::ExceptionTranslatorRegistrar . . . . .	26
Catch::ExprLhs< LhsT > . . . . .	26
Catch::IExceptionTranslator . . . . .	27
Catch::IExceptionTranslatorRegistry . . . . .	27
Catch::IMutableRegistryHub . . . . .	27
Catch::IRegistryHub . . . . .	28
Catch::IResultCapture . . . . .	28
Catch::IRunner . . . . .	29
Catch::is_range< T > . . . . .	29
Catch::Detail::IsStreamInsertable< T > . . . . .	29
Catch::IStream . . . . .	30
Catch::ITestCaseRegistry . . . . .	30
Catch::ITestInvoker . . . . .	30
Catch::TestInvokerAsMethod< C > . . . . .	59
Catch::ITransientExpression . . . . .	31
Catch::BinaryExpr< LhsT, RhsT > . . . . .	16
Catch::MatchExpr< ArgT, MatcherT > . . . . .	36
Catch::UnaryExpr< LhsT > . . . . .	60
Kitchen . . . . .	31
Catch::LazyExpression . . . . .	32
Catch::Matchers::Impl::MatcherMethod< ObjectT > . . . . .	35
Catch::Matchers::Impl::MatcherMethod< ArgT > . . . . .	35

Catch::Matchers::Impl::MatcherBase< ArgT > . . . . .	34
Catch::Matchers::Impl::MatchAllOf< ArgT > . . . . .	33
Catch::Matchers::Impl::MatchAnyOf< ArgT > . . . . .	34
Catch::Matchers::Impl::MatchNotOf< ArgT > . . . . .	37
Catch::Matchers::Impl::MatcherMethod< double > . . . . .	35
Catch::Matchers::Impl::MatcherBase< double > . . . . .	34
Catch::Matchers::Impl::MatcherMethod< PtrT * > . . . . .	35
Catch::Matchers::Impl::MatcherMethod< std::string > . . . . .	35
Catch::Matchers::Impl::MatcherBase< std::string > . . . . .	34
Catch::Matchers::Impl::MatcherMethod< std::vector< T > > . . . . .	35
Catch::Matchers::Impl::MatcherBase< std::vector< T > > . . . . .	34
Catch::Matchers::Impl::MatcherMethod< T > . . . . .	35
Catch::Matchers::Impl::MatcherBase< T > . . . . .	34
Catch::Matchers::Floating::WithinAbsMatcher . . . . .	61
Catch::Matchers::Floating::WithinUlpMatcher . . . . .	61
Catch::Matchers::StdString::RegexMatcher . . . . .	41
Catch::Matchers::StdString::StringMatcherBase . . . . .	54
Catch::Matchers::StdString::ContainsMatcher . . . . .	19
Catch::Matchers::StdString::EndsWithMatcher . . . . .	24
Catch::Matchers::StdString::EqualsMatcher . . . . .	25
Catch::Matchers::StdString::StartsWithMatcher . . . . .	46
Catch::Matchers::Vector::ContainsElementMatcher< T > . . . . .	19
Catch::Matchers::Vector::ContainsMatcher< T > . . . . .	20
Catch::Matchers::Vector::EqualsMatcher< T > . . . . .	25
Catch::Matchers::Vector::UnorderedEqualsMatcher< T > . . . . .	60
Catch::Matchers::Impl::MatcherUntypedBase . . . . .	36
Catch::Matchers::Impl::MatcherBase< T > . . . . .	34
Catch::Matchers::Impl::MatcherBase< ArgT > . . . . .	34
Catch::Matchers::Impl::MatcherBase< double > . . . . .	34
Catch::Matchers::Impl::MatcherBase< std::string > . . . . .	34
Catch::Matchers::Impl::MatcherBase< std::vector< T > > . . . . .	34
Catch::MessageInfo . . . . .	38
Catch::MessageStream . . . . .	39
Catch::MessageBuilder . . . . .	38
Catch::NameAndTags . . . . .	39
Catch::NonCopyable . . . . .	40
Catch::AutoReg . . . . .	15
Catch::Section . . . . .	44
Catch::not_this_one . . . . .	40
Order . . . . .	40
Catch::pluralise . . . . .	41
Catch::RegistrarForTagAliases . . . . .	42
Catch::ResultDisposition . . . . .	42
Catch::ResultWas . . . . .	42
Catch::ReusableStringStream . . . . .	43
Catch::ScopedMessage . . . . .	43
Catch::SectionEndInfo . . . . .	44
Catch::SectionInfo . . . . .	45
Catch::SourceLineInfo . . . . .	45
Catch::StreamEndStop . . . . .	46
Catch::StringMaker< T, typename > . . . . .	47
Catch::StringMaker< bool > . . . . .	47
Catch::StringMaker< Catch::Detail::Approx > . . . . .	47
Catch::StringMaker< char * > . . . . .	48
Catch::StringMaker< char > . . . . .	48
Catch::StringMaker< char const * > . . . . .	48



Catch::StringMaker< char[SZ]> . . . . .	48
Catch::StringMaker< double > . . . . .	49
Catch::StringMaker< float > . . . . .	49
Catch::StringMaker< int > . . . . .	49
Catch::StringMaker< long > . . . . .	49
Catch::StringMaker< long long > . . . . .	50
Catch::StringMaker< R C::* > . . . . .	50
Catch::StringMaker< R, typename std::enable_if< is_range< R >::value &&!::Catch::Detail::IsStream← Insertable< R >::value >::type > . . . . .	50
Catch::StringMaker< signed char > . . . . .	51
Catch::StringMaker< signed char[SZ]> . . . . .	51
Catch::StringMaker< std::nullptr_t > . . . . .	51
Catch::StringMaker< std::string > . . . . .	51
Catch::StringMaker< std::wstring > . . . . .	52
Catch::StringMaker< T * > . . . . .	52
Catch::StringMaker< T[SZ]> . . . . .	52
Catch::StringMaker< unsigned char > . . . . .	52
Catch::StringMaker< unsigned char[SZ]> . . . . .	53
Catch::StringMaker< unsigned int > . . . . .	53
Catch::StringMaker< unsigned long > . . . . .	53
Catch::StringMaker< unsigned long long > . . . . .	53
Catch::StringMaker< wchar_t * > . . . . .	54
Catch::StringMaker< wchar_t const * > . . . . .	54
Catch::StringRef . . . . .	55
SupplyRunner . . . . .	56
Catch::TestCaseInfo . . . . .	57
Catch::TestCase . . . . .	57
Catch::TestFailureException . . . . .	58
Catch::Timer . . . . .	59
Catch::Totals . . . . .	59



## Chapter 4

# Class Index

### 4.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Catch::Detail::Approx	13
Catch::AssertionHandler	14
Catch::AssertionInfo	15
Catch::AssertionReaction	15
Catch::AutoReg	15
Catch::BenchmarkLooper	16
Catch::BinaryExpr< LhsT, RhsT >	16
Catch::Matchers::StdString::CasedString	16
Catch::CaseSensitive	17
Cashier	
Interacts with Customers, charges them money, and parses their orders	17
Catch_global_namespace_dummy	18
Catch::Matchers::Vector::ContainsElementMatcher< T >	19
Catch::Matchers::StdString::ContainsMatcher	19
Catch::Matchers::Vector::ContainsMatcher< T >	20
Cook	
Prepares dishes using a <a href="#">SupplyRunner</a> and a <a href="#">Kitchen</a>	21
Catch::Counts	22
Customer	
A class to represent restaurant customers	22
Catch::Decomposer	24
Catch::Matchers::StdString::EndsWithMatcher	24
Catch::Matchers::StdString::EqualsMatcher	25
Catch::Matchers::Vector::EqualsMatcher< T >	25
Catch::ExceptionTranslatorRegistrar	26
Catch::ExprLhs< LhsT >	26
Catch::IExceptionTranslator	27
Catch::IExceptionTranslatorRegistry	27
Catch::IMutableRegistryHub	27
Catch::IRegistryHub	28
Catch::IResultCapture	28
Catch::IRunner	29
Catch::is_range< T >	29
Catch::Detail::IsStreamInsertable< T >	29
Catch::IStream	30

Catch::ITestCaseRegistry	30
Catch::ITestInvoker	30
Catch::ITransientExpression	31
Kitchen	
A <a href="#">Kitchen</a> that can be used to prepare Dish objects by providing them a moved-map of ingredients	31
Catch::LazyExpression	32
Catch::Matchers::Impl::MatchAllOf< ArgT >	33
Catch::Matchers::Impl::MatchAnyOf< ArgT >	34
Catch::Matchers::Impl::MatcherBase< T >	34
Catch::Matchers::Impl::MatcherMethod< ObjectT >	35
Catch::Matchers::Impl::MatcherMethod< PtrT * >	35
Catch::Matchers::Impl::MatcherUntypedBase	36
Catch::MatchExpr< ArgT, MatcherT >	36
Catch::Matchers::Impl::MatchNotOf< ArgT >	37
Catch::MessageBuilder	38
Catch::MessageInfo	38
Catch::MessageStream	39
Catch::NameAndTags	39
Catch::NonCopyable	40
Catch::not_this_one	40
Order	
Represents a single <a href="#">Order</a> for a list of items	40
Catch::pluralise	41
Catch::Matchers::StdString::RegexMatcher	41
Catch::RegistrarForTagAliases	42
Catch::ResultDisposition	42
Catch::ResultWas	42
Catch::ReusableStringStream	43
Catch::ScopedMessage	43
Catch::Section	44
Catch::SectionEndInfo	44
Catch::SectionInfo	45
Catch::SourceLineInfo	45
Catch::Matchers::StdString::StartsWithMatcher	46
Catch::StreamEndStop	46
Catch::StringMaker< T, typename >	47
Catch::StringMaker< bool >	47
Catch::StringMaker< Catch::Detail::Approx >	47
Catch::StringMaker< char * >	48
Catch::StringMaker< char >	48
Catch::StringMaker< char const * >	48
Catch::StringMaker< char[SZ]>	48
Catch::StringMaker< double >	49
Catch::StringMaker< float >	49
Catch::StringMaker< int >	49
Catch::StringMaker< long >	49
Catch::StringMaker< long long >	50
Catch::StringMaker< R C::* >	50
Catch::StringMaker< R, typename std::enable_if< is_range< R >::value &&!::Catch::Detail::IsStreamInsertable< R >::value >::value >	50
Catch::StringMaker< signed char >	51
Catch::StringMaker< signed char[SZ]>	51
Catch::StringMaker< std::nullptr_t >	51
Catch::StringMaker< std::string >	51
Catch::StringMaker< std::wstring >	52
Catch::StringMaker< T * >	52
Catch::StringMaker< T[SZ]>	52
Catch::StringMaker< unsigned char >	52

Catch::StringMaker< unsigned char[SZ]> . . . . .	53
Catch::StringMaker< unsigned int > . . . . .	53
Catch::StringMaker< unsigned long > . . . . .	53
Catch::StringMaker< unsigned long long > . . . . .	53
Catch::StringMaker< wchar_t * > . . . . .	54
Catch::StringMaker< wchar_t const * > . . . . .	54
Catch::Matchers::StdString::StringMatcherBase . . . . .	54
Catch::StringRef . . . . .	55
SupplyRunner	
Gets Ingredients from the Storeroom . . . . .	56
Catch::TestCase . . . . .	57
Catch::TestCaseInfo . . . . .	57
Catch::TestFailureException . . . . .	58
Catch::TestInvokerAsMethod< C > . . . . .	59
Catch::Timer . . . . .	59
Catch::Totals . . . . .	59
Catch::UnaryExpr< LhsT > . . . . .	60
Catch::Matchers::Vector::UnorderedEqualsMatcher< T > . . . . .	60
Catch::Matchers::Floating::WithinAbsMatcher . . . . .	61
Catch::Matchers::Floating::WithinUlpMatcher . . . . .	61



## Chapter 5

# Class Documentation

### 5.1 Catch::Detail::Approx Class Reference

#### 5.1.1 \*

#### Public Member Functions

- **Approx** (double value)
- template<typename T , typename = typename std::enable\_if<std::is\_constructible<double, T>::value>::type>  
[Approx](#) **operator()** (T const &value)
- template<typename T , typename = typename std::enable\_if<std::is\_constructible<double, T>::value>::type>  
**Approx** (T const &value)
- template<typename T , typename = typename std::enable\_if<std::is\_constructible<double, T>::value>::type>  
[Approx](#) & **epsilon** (T const &newEpsilon)
- template<typename T , typename = typename std::enable\_if<std::is\_constructible<double, T>::value>::type>  
[Approx](#) & **margin** (T const &newMargin)
- template<typename T , typename = typename std::enable\_if<std::is\_constructible<double, T>::value>::type>  
[Approx](#) & **scale** (T const &newScale)
- std::string **toString** () const

#### 5.1.2 \*

#### Static Public Member Functions

- static [Approx](#) **custom** ()

### 5.1.3 \*

#### Friends

- `template<typename T, typename = typename std::enable_if<std::is_constructible<double, T>::value>::type>`  
`bool operator== (const T &lhs, Approx const &rhs)`
- `template<typename T, typename = typename std::enable_if<std::is_constructible<double, T>::value>::type>`  
`bool operator== (Approx const &lhs, const T &rhs)`
- `template<typename T, typename = typename std::enable_if<std::is_constructible<double, T>::value>::type>`  
`bool operator!= (T const &lhs, Approx const &rhs)`
- `template<typename T, typename = typename std::enable_if<std::is_constructible<double, T>::value>::type>`  
`bool operator!= (Approx const &lhs, T const &rhs)`
- `template<typename T, typename = typename std::enable_if<std::is_constructible<double, T>::value>::type>`  
`bool operator<= (T const &lhs, Approx const &rhs)`
- `template<typename T, typename = typename std::enable_if<std::is_constructible<double, T>::value>::type>`  
`bool operator<= (Approx const &lhs, T const &rhs)`
- `template<typename T, typename = typename std::enable_if<std::is_constructible<double, T>::value>::type>`  
`bool operator>= (T const &lhs, Approx const &rhs)`
- `template<typename T, typename = typename std::enable_if<std::is_constructible<double, T>::value>::type>`  
`bool operator>= (Approx const &lhs, T const &rhs)`

The documentation for this class was generated from the following file:

- `dep/catch.hpp`

## 5.2 Catch::AssertionHandler Class Reference

### 5.2.1 \*

#### Public Member Functions

- **AssertionHandler** ([StringRef](#) macroName, [SourceLineInfo](#) const &lineInfo, [StringRef](#) capturedExpression, ResultDisposition::Flags resultDisposition)
- `template<typename T >`  
`void handleExpr (ExprLhs< T > const &expr)`
- `void handleExpr (ITransientExpression const &expr)`
- `void handleMessage (ResultWas::OfType resultType, StringRef const &message)`
- `void handleExceptionThrownAsExpected ()`
- `void handleUnexpectedExceptionNotThrown ()`
- `void handleExceptionNotThrownAsExpected ()`
- `void handleThrowingCallSkipped ()`
- `void handleUnexpectedInflightException ()`
- `void complete ()`
- `void setCompleted ()`
- `auto allowThrows () const -> bool`

The documentation for this class was generated from the following file:

- `dep/catch.hpp`



## 5.3 Catch::AssertionInfo Struct Reference

### 5.3.1 \*

Public Attributes

- [StringRef](#) **macroName**
- [SourceLineInfo](#) **lineInfo**
- [StringRef](#) **capturedExpression**
- ResultDisposition::Flags **resultDisposition**

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.4 Catch::AssertionReaction Struct Reference

### 5.4.1 \*

Public Attributes

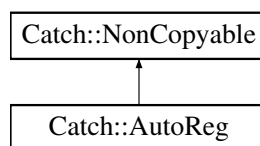
- bool **shouldDebugBreak** = false
- bool **shouldThrow** = false

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.5 Catch::AutoReg Struct Reference

Inheritance diagram for Catch::AutoReg:



### 5.5.1 \*

Public Member Functions

- **AutoReg** ([ITestInvoker](#) \*invoker, [SourceLineInfo](#) const &lineInfo, [StringRef](#) classOrMethod, [NameAndTags](#) const &nameAndTags) noexcept

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.6 Catch::BenchmarkLooper Class Reference

### 5.6.1 \*

Public Member Functions

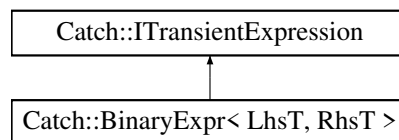
- **BenchmarkLooper** ([StringRef](#) name)
- **operator bool** ()
- void **increment** ()
- void **reportStart** ()
- auto **needsMoreIterations** () -> bool

The documentation for this class was generated from the following file:

- dep/catch.hpp

## 5.7 Catch::BinaryExpr< LhsT, RhsT > Class Template Reference

Inheritance diagram for Catch::BinaryExpr< LhsT, RhsT >:



### 5.7.1 \*

Public Member Functions

- **BinaryExpr** (bool comparisonResult, LhsT lhs, [StringRef](#) op, RhsT rhs)

### 5.7.2 \*

Additional Inherited Members

The documentation for this class was generated from the following file:

- dep/catch.hpp

## 5.8 Catch::Matchers::StdString::CasedString Struct Reference

### 5.8.1 \*

Public Member Functions

- **CasedString** (std::string const &str, CaseSensitive::Choice caseSensitivity)
- std::string **adjustString** (std::string const &str) const
- std::string **caseSensitivitySuffix** () const

### 5.8.2 \*

Public Attributes

- CaseSensitive::Choice **m\_caseSensitivity**
- std::string **m\_str**

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.9 Catch::CaseSensitive Struct Reference

### 5.9.1 \*

Public Types

- enum **Choice** { **Yes**, **No** }

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.10 Cashier Class Reference

Interacts with Customers, charges them money, and parses their orders.

```
#include <Cashier.hpp>
```

### 5.10.1 \*

Public Member Functions

- **Cashier** (double money\_)  
*Constructs the **Cashier** with a set amount of money.*
- double **get\_money** () const  
*Gets the amount of money that the cashier holds.*
- void **serve\_customer** (std::queue< **Customer** > &line, std::stack< **Order** > &orders)  
*Serves a customer by taking them off the queue, and possibly putting an order, if valid, onto the order stack.*

### 5.10.2 Detailed Description

Interacts with Customers, charges them money, and parses their orders.

Note

Must give each **Customer** upon interaction a unique ID number, counting up from 0.

### 5.10.3 Member Function Documentation

#### 5.10.3.1 `serve_customer()`

```
void Cashier::serve_customer (
    std::queue< Customer > & line,
    std::stack< Order > & orders )
```

Serves a customer by taking them off the queue, and possibly putting an order, if valid, onto the order stack.

- Retrieves the customer from the queue, popping them off.
- Receives the customer's desired order items in string format:  
<number-of-items> <name-of-item-with-no-spaces>
  - Multiple items are separated simply by a space
- If the [Cashier](#) detects an order for an invalid item that is not on the menu, it must `expel()` the customer.
- Must calculate the cost of all the items ordered, and charge the customer.
  - If the customer does not have enough money to pay, do not charge the customer, but instead, `expel()` them.
- Once paid for, the order items must be consolidated into an [Order](#), which is tagged with the unique customer ID generated at the beginning of this function, and then push it onto the stack.

See also

[Customer](#)

[Order](#)

The documentation for this class was generated from the following file:

- `include/Cashier.hpp`

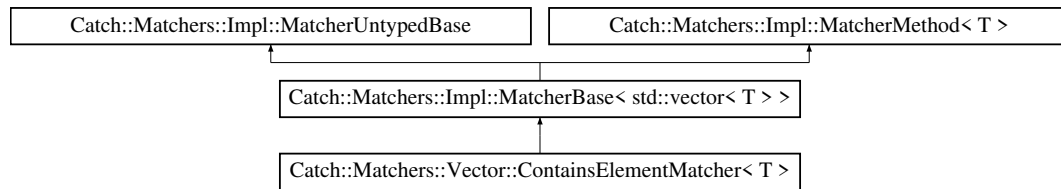
### 5.11 `Catch_global_namespace_dummy` Struct Reference

The documentation for this struct was generated from the following file:

- `dep/catch.hpp`

## 5.12 Catch::Matchers::Vector::ContainsElementMatcher< T > Struct Template Reference

Inheritance diagram for Catch::Matchers::Vector::ContainsElementMatcher< T >:



### 5.12.1 \*

Public Member Functions

- **ContainsElementMatcher** (T const &comparator)
- bool **match** (std::vector< T > const &v) const override
- std::string **describe** () const override

### 5.12.2 \*

Public Attributes

- T const & **m\_comparator**

### 5.12.3 \*

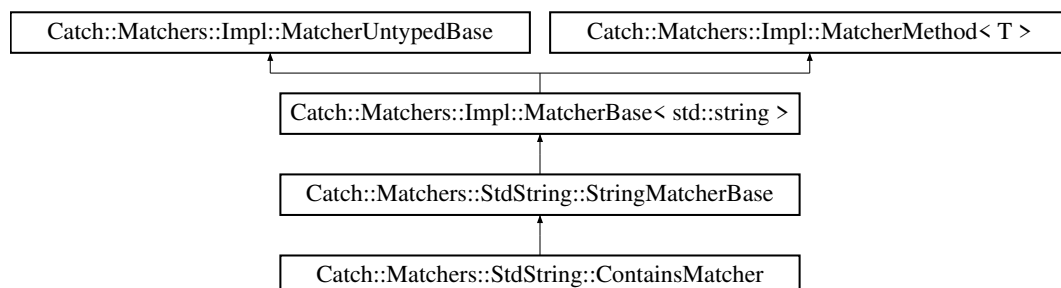
Additional Inherited Members

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.13 Catch::Matchers::StdString::ContainsMatcher Struct Reference

Inheritance diagram for Catch::Matchers::StdString::ContainsMatcher:



### 5.13.1 \*

Public Member Functions

- **ContainsMatcher** (CasedString const &comparator)
- bool **match** (std::string const &source) const override

### 5.13.2 \*

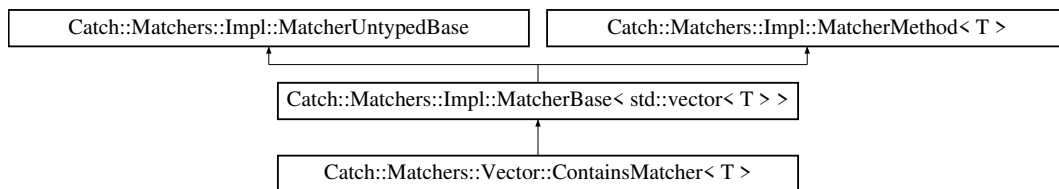
#### Additional Inherited Members

The documentation for this struct was generated from the following file:

- `dep/catch.hpp`

## 5.14 `Catch::Matchers::Vector::ContainsMatcher< T >` Struct Template Reference

Inheritance diagram for `Catch::Matchers::Vector::ContainsMatcher< T >`:



### 5.14.1 \*

#### Public Member Functions

- **`ContainsMatcher`** (`std::vector< T > const &comparator`)
- `bool match` (`std::vector< T > const &v`) `const` override
- `std::string describe` () `const` override

### 5.14.2 \*

#### Public Attributes

- `std::vector< T > const & m_comparator`

### 5.14.3 \*

#### Additional Inherited Members

The documentation for this struct was generated from the following file:

- `dep/catch.hpp`

## 5.15 Cook Class Reference

Prepares dishes using a [SupplyRunner](#) and a [Kitchen](#).

```
#include <Cook.hpp>
```

### 5.15.1 \*

Public Member Functions

- [Cook](#) ([SupplyRunner](#) &runner\_, [Kitchen](#) &kitchen\_)  
*Constructs a [Cook](#) with references to a [SupplyRunner](#) and a [Kitchen](#), which it must use in the process of cooking.*
- void [prepare\\_dish](#) (std::stack< [Order](#) > &orders, std::queue< std::pair< std::size\_t, Dish >> &finished\_dishes)  
*Prepares Dishes from a single order.*

### 5.15.2 Detailed Description

Prepares dishes using a [SupplyRunner](#) and a [Kitchen](#).

Note

Uses dependency injection with the constructor.

### 5.15.3 Member Function Documentation

#### 5.15.3.1 [prepare\\_dish\(\)](#)

```
void Cook::prepare_dish (
    std::stack< Order > & orders,
    std::queue< std::pair< std::size_t, Dish >> & finished_dishes )
```

Prepares Dishes from a single order.

- Take an [Order](#) from the stack.
- For each order item in the [Order](#),
  - Lookup its required ingredients in the RecipeBook.
  - Then, ask the [SupplyRunner](#) to get the correct amount of ingredients.
  - Put the vector of ingredients into an IngredientMap.
  - Send the IngredientMap to the [Kitchen](#) to have it turned into a dish.
  - Put the finished Dish onto the queue, embedding it in a pair that also contains the [Customer](#) ID of origin.

The documentation for this class was generated from the following file:

- include/Cook.hpp

## 5.16 Catch::Counts Struct Reference

### 5.16.1 \*

#### Public Member Functions

- [Counts](#) **operator-** ([Counts](#) const &other) const
- [Counts](#) & **operator+=** ([Counts](#) const &other)
- std::size\_t **total** () const
- bool **allPassed** () const
- bool **allOk** () const

### 5.16.2 \*

#### Public Attributes

- std::size\_t **passed** = 0
- std::size\_t **failed** = 0
- std::size\_t **failedButOk** = 0

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.17 Customer Class Reference

A class to represent restaurant customers.

```
#include <Customer.hpp>
```

### 5.17.1 \*

#### Public Member Functions

- [Customer](#) (double money\_, std::string order\_str\_)  
*Constructs a [Customer](#) with a set amount of money and a string to emit upon have its order taken as a string.*
- double [get\\_money](#) () const  
*Gets the current amount of money on the [Customer](#).*
- bool [is\\_expelled](#) () const  
*Returns whether the [Customer](#) was expelled.*
- std::string [take\\_order](#) ()  
*Returns the order\_str on the [Customer](#).*
- double [charge\\_money](#) (double amount)  
*Attempt to charge the amount of money from the [Customer](#).*
- void [refund\\_money](#) (double amount)  
*Refunds money back to the customer.*
- std::string [get\\_order](#) () const  
*Returns the order string.*
- void [expel](#) ()  
*Expels the [Customer](#) from the restaurant.*



## 5.17.2 \*

### Friends

- `std::ostream & operator<< (std::ostream &lhs, const Customer &rhs)`  
*Prints out a string representation of the [Customer](#)'s fields.*

## 5.17.3 Detailed Description

A class to represent restaurant customers.

## 5.17.4 Member Function Documentation

### 5.17.4.1 `charge_money()`

```
double Customer::charge_money (
    double amount )
```

Attempt to charge the amount of money from the [Customer](#).

#### Note

If the [Customer](#) doesn't have enough money, it will simply return its current money amount and drain its money to 0.

### 5.17.4.2 `expel()`

```
void Customer::expel ( )
```

Expels the [Customer](#) from the restaurant.

#### Note

For the purposes of testing, this is attached to the `extern std::size_t expelled_count` variable.

**Todo** Revise the design of expellation and the unit test cases so expellation can be unit tested without this global variable counter hack.

### 5.17.4.3 `get_order()`

```
std::string Customer::get_order ( ) const
```

Returns the order string.

#### See also

[take\\_order\(\)](#)

#### 5.17.4.4 refund\_money()

```
void Customer::refund_money (
    double amount )
```

Refunds money back to the customer.

##### Note

There are no bounds checking on this code, so you can refund negative money to charge money as well. Please do not do this; this is not how this member function is supposed to be used.

The documentation for this class was generated from the following files:

- include/Customer.hpp
- src/Customer.cpp

## 5.18 Catch::Decomposer Struct Reference

### 5.18.1 \*

#### Public Member Functions

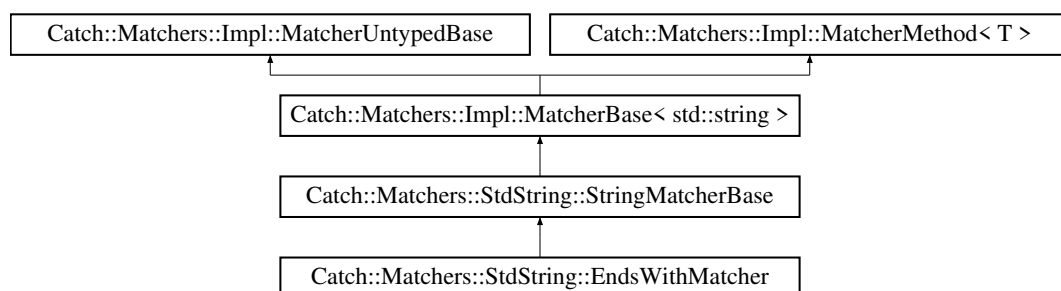
- `template<typename T >`  
`auto operator<= (T const &lhs) -> ExprLhs< T const & >`
- `auto operator<= (bool value) -> ExprLhs< bool >`

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.19 Catch::Matchers::StdString::EndsWithMatcher Struct Reference

Inheritance diagram for Catch::Matchers::StdString::EndsWithMatcher:



### 5.19.1 \*

#### Public Member Functions

- **EndsWithMatcher** (`CasedString` const &comparator)
- `bool match (std::string const &source) const` override

## 5.19.2 \*

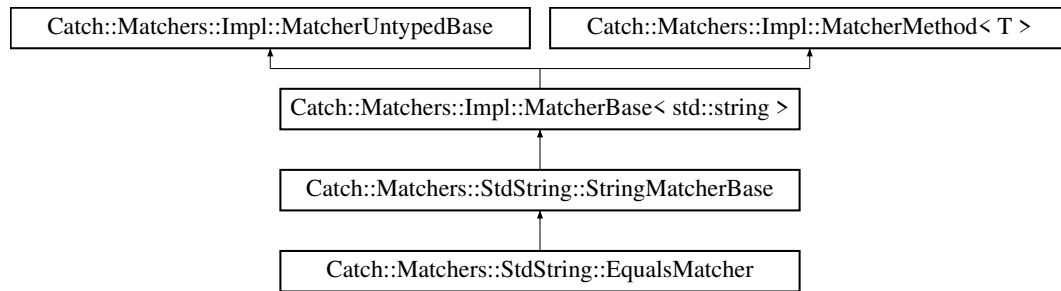
## Additional Inherited Members

The documentation for this struct was generated from the following file:

- `dep/catch.hpp`

## 5.20 Catch::Matchers::StdString::EqualsMatcher Struct Reference

Inheritance diagram for Catch::Matchers::StdString::EqualsMatcher:



## 5.20.1 \*

## Public Member Functions

- **EqualsMatcher** (`CasedString` const &comparator)
- `bool match` (`std::string` const &source) const override

## 5.20.2 \*

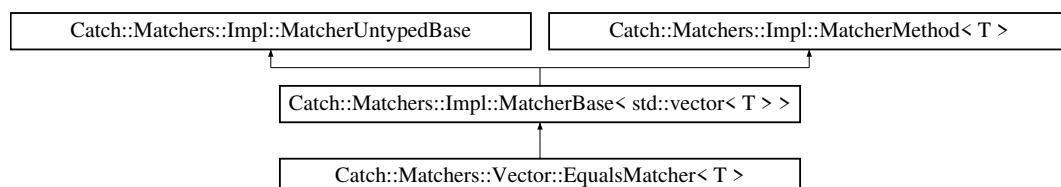
## Additional Inherited Members

The documentation for this struct was generated from the following file:

- `dep/catch.hpp`

## 5.21 Catch::Matchers::Vector::EqualsMatcher&lt; T &gt; Struct Template Reference

Inheritance diagram for Catch::Matchers::Vector::EqualsMatcher< T >:



## 5.21.1 \*

## Public Member Functions

- **EqualsMatcher** (`std::vector< T >` const &comparator)
- `bool match` (`std::vector< T >` const &v) const override
- `std::string describe` () const override

### 5.21.2 \*

Public Attributes

- `std::vector< T > const & m_comparator`

### 5.21.3 \*

Additional Inherited Members

The documentation for this struct was generated from the following file:

- `dep/catch.hpp`

## 5.22 Catch::ExceptionTranslatorRegistrar Class Reference

### 5.22.1 \*

Public Member Functions

- `template<typename T >`  
**ExceptionTranslatorRegistrar** (`std::string(*translateFunction)(T &)`)

The documentation for this class was generated from the following file:

- `dep/catch.hpp`

## 5.23 Catch::ExprLhs< LhsT > Class Template Reference

### 5.23.1 \*

Public Member Functions

- **ExprLhs** (`LhsT lhs`)
- `template<typename RhsT >`  
auto **operator==** (`RhsT const &rhs`) -> [BinaryExpr](#)< `LhsT`, `RhsT const &` > const
- auto **operator==** (`bool rhs`) -> [BinaryExpr](#)< `LhsT`, `bool` > const
- `template<typename RhsT >`  
auto **operator!=** (`RhsT const &rhs`) -> [BinaryExpr](#)< `LhsT`, `RhsT const &` > const
- auto **operator!=** (`bool rhs`) -> [BinaryExpr](#)< `LhsT`, `bool` > const
- `template<typename RhsT >`  
auto **operator>** (`RhsT const &rhs`) -> [BinaryExpr](#)< `LhsT`, `RhsT const &` > const
- `template<typename RhsT >`  
auto **operator<** (`RhsT const &rhs`) -> [BinaryExpr](#)< `LhsT`, `RhsT const &` > const
- `template<typename RhsT >`  
auto **operator>=** (`RhsT const &rhs`) -> [BinaryExpr](#)< `LhsT`, `RhsT const &` > const
- `template<typename RhsT >`  
auto **operator<=** (`RhsT const &rhs`) -> [BinaryExpr](#)< `LhsT`, `RhsT const &` > const
- auto **makeUnaryExpr** () const -> [UnaryExpr](#)< `LhsT` >

The documentation for this class was generated from the following file:

- `dep/catch.hpp`

## 5.24 Catch::IExceptionTranslator Struct Reference

### 5.24.1 \*

#### Public Member Functions

- virtual std::string **translate** (ExceptionTranslators::const\_iterator it, ExceptionTranslators::const\_iterator it↵End) const =0

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.25 Catch::IExceptionTranslatorRegistry Struct Reference

### 5.25.1 \*

#### Public Member Functions

- virtual std::string **translateActiveException** () const =0

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.26 Catch::IMutableRegistryHub Struct Reference

### 5.26.1 \*

#### Public Member Functions

- virtual void **registerReporter** (std::string const &name, IReporterFactoryPtr const &factory)=0
- virtual void **registerListener** (IReporterFactoryPtr const &factory)=0
- virtual void **registerTest** (TestCase const &testInfo)=0
- virtual void **registerTranslator** (const IExceptionTranslator \*translator)=0
- virtual void **registerTagAlias** (std::string const &alias, std::string const &tag, SourceLineInfo const &line↵Info)=0
- virtual void **registerStartupException** () noexcept=0

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.27 Catch::IRegistryHub Struct Reference

### 5.27.1 \*

#### Public Member Functions

- virtual IReporterRegistry const & **getReporterRegistry** () const =0
- virtual ITestCaseRegistry const & **getTestCaseRegistry** () const =0
- virtual ITagAliasRegistry const & **getTagAliasRegistry** () const =0
- virtual IExceptionTranslatorRegistry & **getExceptionTranslatorRegistry** ()=0
- virtual StartupExceptionRegistry const & **getStartupExceptionRegistry** () const =0

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.28 Catch::IResultCapture Struct Reference

### 5.28.1 \*

#### Public Member Functions

- virtual bool **sectionStarted** (SectionInfo const &sectionInfo, Counts &assertions)=0
- virtual void **sectionEnded** (SectionEndInfo const &endInfo)=0
- virtual void **sectionEndedEarly** (SectionEndInfo const &endInfo)=0
- virtual void **benchmarkStarting** (BenchmarkInfo const &info)=0
- virtual void **benchmarkEnded** (BenchmarkStats const &stats)=0
- virtual void **pushScopedMessage** (MessageInfo const &message)=0
- virtual void **popScopedMessage** (MessageInfo const &message)=0
- virtual void **handleFatalErrorCondition** (StringRef message)=0
- virtual void **handleExpr** (AssertionInfo const &info, ITransientExpression const &expr, AssertionReaction &reaction)=0
- virtual void **handleMessage** (AssertionInfo const &info, ResultWas::OfType resultType, StringRef const &message, AssertionReaction &reaction)=0
- virtual void **handleUnexpectedExceptionNotThrown** (AssertionInfo const &info, AssertionReaction &reaction)=0
- virtual void **handleUnexpectedInflightException** (AssertionInfo const &info, std::string const &message, AssertionReaction &reaction)=0
- virtual void **handleIncomplete** (AssertionInfo const &info)=0
- virtual void **handleNonExpr** (AssertionInfo const &info, ResultWas::OfType resultType, AssertionReaction &reaction)=0
- virtual bool **lastAssertionPassed** ()=0
- virtual void **assertionPassed** ()=0
- virtual std::string **getCurrentTestName** () const =0
- virtual const AssertionResult \* **getLastResult** () const =0
- virtual void **exceptionEarlyReported** ()=0

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.29 Catch::IRunner Struct Reference

### 5.29.1 \*

Public Member Functions

- virtual bool **aborting** () const =0

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.30 Catch::is\_range< T > Struct Template Reference

### 5.30.1 \*

Static Public Attributes

- static const bool **value**

### 5.30.2 Member Data Documentation

#### 5.30.2.1 value

```
template<typename T >
const bool Catch::is_range< T >::value [static]
```

**Initial value:**

```
=
    !std::is_same<decltype(begin(std::declval<T>())), not_this_one>::value &&
    !std::is_same<decltype(end(std::declval<T>())), not_this_one>::value
```

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.31 Catch::Detail::IsStreamInsertable< T > Class Template Reference

### 5.31.1 \*

Static Public Attributes

- static const bool **value** = decltype(test<std::ostream, const T&>(0))::value

The documentation for this class was generated from the following file:

- dep/catch.hpp

## 5.32 Catch::IStream Struct Reference

### 5.32.1 \*

Public Member Functions

- virtual std::ostream & **stream** () const =0

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.33 Catch::ITestCaseRegistry Struct Reference

### 5.33.1 \*

Public Member Functions

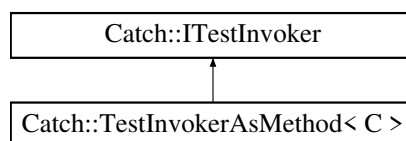
- virtual std::vector< [TestCase](#) > const & **getAllTests** () const =0
- virtual std::vector< [TestCase](#) > const & **getAllTestsSorted** (IConfig const &config) const =0

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.34 Catch::ITestInvoker Struct Reference

Inheritance diagram for Catch::ITestInvoker:



### 5.34.1 \*

Public Member Functions

- virtual void **invoke** () const =0

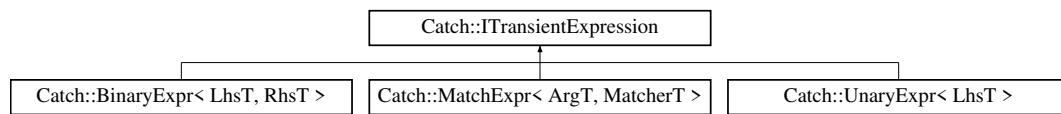
The documentation for this struct was generated from the following file:

- dep/catch.hpp



## 5.35 Catch::ITransientExpression Struct Reference

Inheritance diagram for Catch::ITransientExpression:



### 5.35.1 \*

Public Member Functions

- auto **isBinaryExpression** () const -> bool
- auto **getResult** () const -> bool
- virtual void **streamReconstructedExpression** (std::ostream &os) const =0
- **ITransientExpression** (bool isBinaryExpression, bool result)

### 5.35.2 \*

Public Attributes

- bool **m\_isBinaryExpression**
- bool **m\_result**

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.36 Kitchen Class Reference

A [Kitchen](#) that can be used to prepare Dish objects by providing them a moved-map of ingredients.

```
#include <Kitchen.hpp>
```

### 5.36.1 \*

Public Member Functions

- Dish [prepare\\_dish](#) (IngredientMap &&ingredients)  
*Consumes the map of ingredients (std::string) and returns the corresponding Dish.*

### 5.36.2 Detailed Description

A [Kitchen](#) that can be used to prepare Dish objects by providing them a moved-map of ingredients.

### 5.36.3 Member Function Documentation

#### 5.36.3.1 `prepare_dish()`

```
Dish Kitchen::prepare_dish (
    IngredientMap && ingredients )
```

Consumes the map of ingredients (`std::string`) and returns the corresponding Dish.

#### Note

This will return `Dish::INEDIBLE` if a map of ingredients that doesn't correspond to a proper dish is given.  
You must noticably **MOVE** the ingredients in or give it a map literal.

#### See also

[Ingredient.hpp](#)

#### Note

Look at the lab manual for more information about the item costs.

The documentation for this class was generated from the following files:

- `include/Kitchen.hpp`
- `src/Kitchen.cpp`

## 5.37 `Catch::LazyExpression` Class Reference

### 5.37.1 \*

#### Public Member Functions

- **LazyExpression** (`bool isNegated`)
- **LazyExpression** ([LazyExpression](#) const &other)
- [LazyExpression](#) & **operator=** ([LazyExpression](#) const &)=delete
- **operator bool** () const

## 5.37.2 \*

## Friends

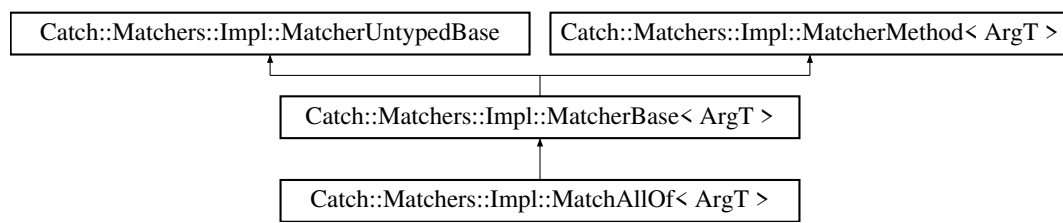
- class **AssertionHandler**
- struct **AssertionStats**
- class **RunContext**
- auto **operator**<< (std::ostream &os, [LazyExpression](#) const &lazyExpr) -> std::ostream &

The documentation for this class was generated from the following file:

- dep/catch.hpp

## 5.38 Catch::Matchers::Impl::MatchAllOf&lt; ArgT &gt; Struct Template Reference

Inheritance diagram for Catch::Matchers::Impl::MatchAllOf< ArgT >:



## 5.38.1 \*

## Public Member Functions

- bool **match** (ArgT const &arg) const override
- std::string **describe** () const override
- [MatchAllOf](#)< ArgT > & **operator&&** ([MatcherBase](#)< ArgT > const &other)

## 5.38.2 \*

## Public Attributes

- std::vector< [MatcherBase](#)< ArgT > const \* > **m\_matchers**

## 5.38.3 \*

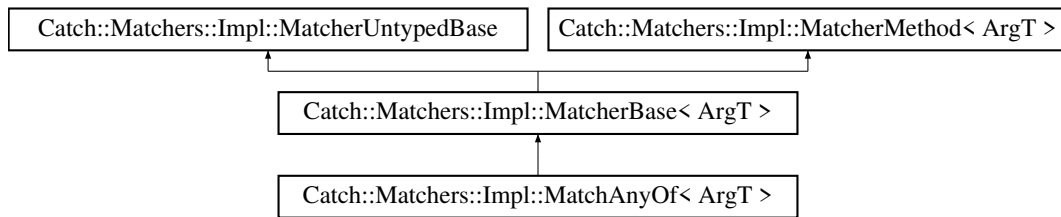
## Additional Inherited Members

The documentation for this struct was generated from the following file:

- dep/catch.hpp

### 5.39 Catch::Matchers::Impl::MatchAnyOf< ArgT > Struct Template Reference

Inheritance diagram for Catch::Matchers::Impl::MatchAnyOf< ArgT >:



#### 5.39.1 \*

Public Member Functions

- bool **match** (ArgT const &arg) const override
- std::string **describe** () const override
- [MatchAnyOf< ArgT >](#) & **operator||** ([MatcherBase< ArgT >](#) const &other)

#### 5.39.2 \*

Public Attributes

- std::vector< [MatcherBase< ArgT >](#) const \* > **m\_matchers**

#### 5.39.3 \*

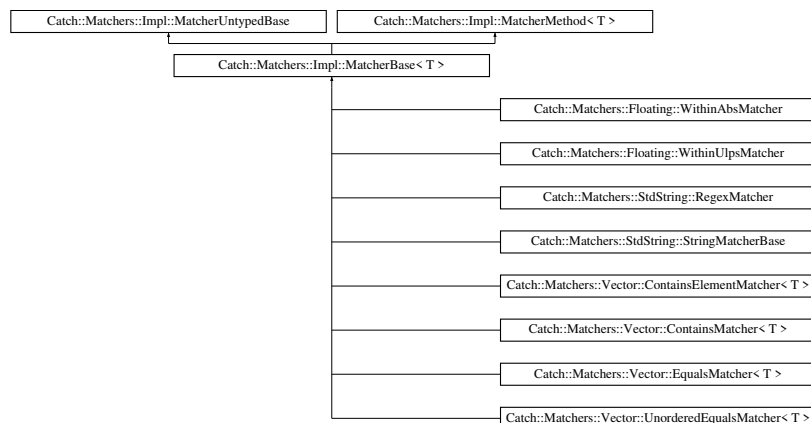
Additional Inherited Members

The documentation for this struct was generated from the following file:

- dep/catch.hpp

### 5.40 Catch::Matchers::Impl::MatcherBase< T > Struct Template Reference

Inheritance diagram for Catch::Matchers::Impl::MatcherBase< T >:



## 5.40.1 \*

Public Member Functions

- `MatchAllOf< T > operator&& (MatcherBase const &other) const`
- `MatchAnyOf< T > operator|| (MatcherBase const &other) const`
- `MatchNotOf< T > operator! () const`

## 5.40.2 \*

Additional Inherited Members

The documentation for this struct was generated from the following file:

- `dep/catch.hpp`

5.41 `Catch::Matchers::Impl::MatcherMethod< ObjectT >` Struct Template Reference

## 5.41.1 \*

Public Member Functions

- virtual bool **match** (ObjectT const &arg) const =0

The documentation for this struct was generated from the following file:

- `dep/catch.hpp`

5.42 `Catch::Matchers::Impl::MatcherMethod< PtrT * >` Struct Template Reference

## 5.42.1 \*

Public Member Functions

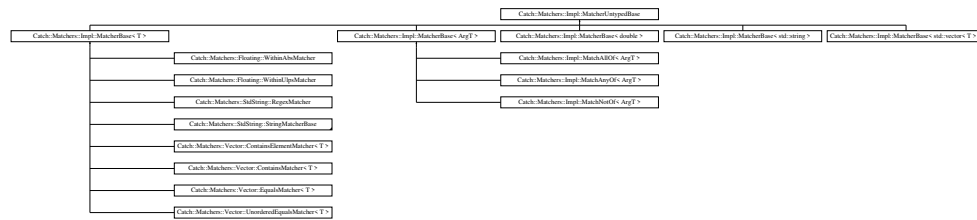
- virtual bool **match** (PtrT \*arg) const =0

The documentation for this struct was generated from the following file:

- `dep/catch.hpp`

## 5.43 Catch::Matchers::Impl::MatcherUntypedBase Class Reference

Inheritance diagram for Catch::Matchers::Impl::MatcherUntypedBase:



### 5.43.1 \*

Public Member Functions

- **MatcherUntypedBase** ([MatcherUntypedBase](#) const &)=default
- [MatcherUntypedBase](#) & **operator=** ([MatcherUntypedBase](#) const &)=delete
- std::string **toString** () const

### 5.43.2 \*

Protected Member Functions

- virtual std::string **describe** () const =0

### 5.43.3 \*

Protected Attributes

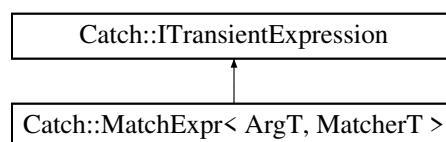
- std::string **m\_cachedToString**

The documentation for this class was generated from the following file:

- dep/catch.hpp

## 5.44 Catch::MatchExpr< ArgT, MatcherT > Class Template Reference

Inheritance diagram for Catch::MatchExpr< ArgT, MatcherT >:



### 5.44.1 \*

Public Member Functions

- **MatchExpr** (ArgT const &arg, MatcherT const &matcher, [StringRef](#) matcherString)
- void **streamReconstructedExpression** (std::ostream &os) const override

## 5.44.2 \*

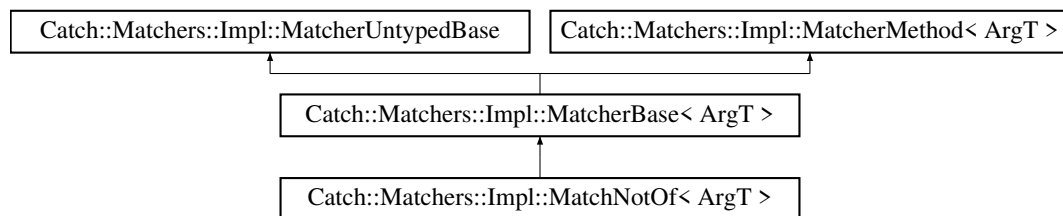
## Additional Inherited Members

The documentation for this class was generated from the following file:

- dep/catch.hpp

## 5.45 Catch::Matchers::Impl::MatchNotOf&lt; ArgT &gt; Struct Template Reference

Inheritance diagram for Catch::Matchers::Impl::MatchNotOf< ArgT >:



## 5.45.1 \*

## Public Member Functions

- **MatchNotOf** ([MatcherBase](#)< ArgT > const &underlyingMatcher)
- bool **match** (ArgT const &arg) const override
- std::string **describe** () const override

## 5.45.2 \*

## Public Attributes

- [MatcherBase](#)< ArgT > const & **m\_underlyingMatcher**

## 5.45.3 \*

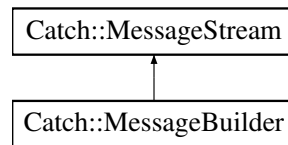
## Additional Inherited Members

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.46 Catch::MessageBuilder Struct Reference

Inheritance diagram for Catch::MessageBuilder:



### 5.46.1 \*

Public Member Functions

- **MessageBuilder** (std::string const &macroName, [SourceLineInfo](#) const &lineInfo, ResultWas::OfType type)
- template<typename T >  
[MessageBuilder](#) & **operator**<< (T const &value)

### 5.46.2 \*

Public Attributes

- [MessageInfo](#) **m\_info**

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.47 Catch::MessageInfo Struct Reference

### 5.47.1 \*

Public Member Functions

- **MessageInfo** (std::string const &\_macroName, [SourceLineInfo](#) const &\_lineInfo, ResultWas::OfType \_type)
- bool **operator**== ([MessageInfo](#) const &other) const
- bool **operator**< ([MessageInfo](#) const &other) const

### 5.47.2 \*

Public Attributes

- std::string **macroName**
- std::string **message**
- [SourceLineInfo](#) **lineInfo**
- ResultWas::OfType **type**
- unsigned int **sequence**

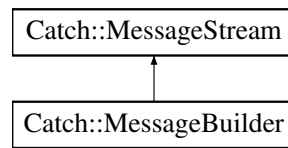
The documentation for this struct was generated from the following file:

- dep/catch.hpp



## 5.48 Catch::MessageStream Struct Reference

Inheritance diagram for Catch::MessageStream:



### 5.48.1 \*

Public Member Functions

- `template<typename T >`  
[MessageStream](#) & **operator**<< (T const &value)

### 5.48.2 \*

Public Attributes

- [ReusableStringStream](#) **m\_stream**

The documentation for this struct was generated from the following file:

- `dep/catch.hpp`

## 5.49 Catch::NameAndTags Struct Reference

### 5.49.1 \*

Public Member Functions

- **NameAndTags** ([StringRef](#) name\_<sub>=</sub>[StringRef](#)(), [StringRef](#) tags\_<sub>=</sub>[StringRef](#)()) noexcept

### 5.49.2 \*

Public Attributes

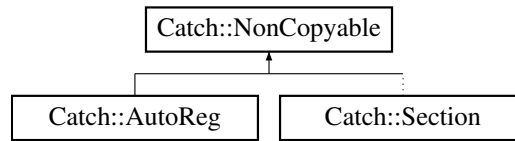
- [StringRef](#) **name**
- [StringRef](#) **tags**

The documentation for this struct was generated from the following file:

- `dep/catch.hpp`

## 5.50 Catch::NonCopyable Class Reference

Inheritance diagram for Catch::NonCopyable:



The documentation for this class was generated from the following file:

- dep/catch.hpp

## 5.51 Catch::not\_this\_one Struct Reference

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.52 Order Class Reference

Represents a single [Order](#) for a list of items.

```
#include <Order.hpp>
```

### 5.52.1 \*

#### Public Member Functions

- [Order](#) (std::size\_t id\_, std::vector< std::string > items\_)  
*Constructors an [Order](#) with a customer ID and the ordered items.*
- std::size\_t [get\\_id](#) () const  
*Gets the ID associated with the [Order](#).*
- std::vector< std::string > [get\\_items](#) () const  
*Gets the vector of order items (strings).*

### 5.52.2 \*

#### Friends

- std::ostream & [operator<<](#) (std::ostream &lhs, const [Order](#) &rhs)  
*Prints out a readable representation of an [Order](#) to an ostream&.*

### 5.52.3 Detailed Description

Represents a single [Order](#) for a list of items.

The documentation for this class was generated from the following files:

- include/Order.hpp
- src/Order.cpp

## 5.53 Catch::pluralise Struct Reference

### 5.53.1 \*

Public Member Functions

- **pluralise** (std::size\_t count, std::string const &label)

### 5.53.2 \*

Public Attributes

- std::size\_t **m\_count**
- std::string **m\_label**

### 5.53.3 \*

Friends

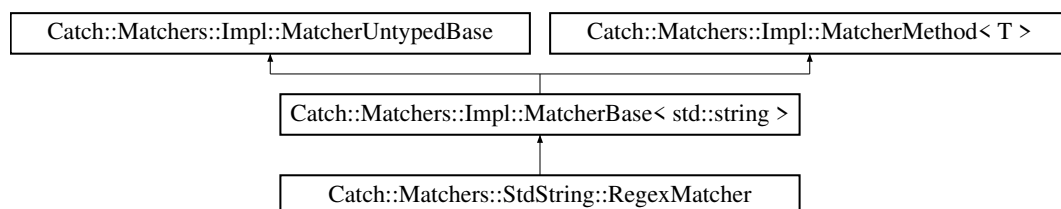
- std::ostream & **operator**<< (std::ostream &os, [pluralise](#) const &pluraliser)

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.54 Catch::Matchers::StdString::RegexMatcher Struct Reference

Inheritance diagram for Catch::Matchers::StdString::RegexMatcher:



### 5.54.1 \*

Public Member Functions

- **RegexMatcher** (std::string regex, CaseSensitive::Choice caseSensitivity)
- bool **match** (std::string const &matchee) const override
- std::string **describe** () const override

## 5.54.2 \*

## Additional Inherited Members

The documentation for this struct was generated from the following file:

- `dep/catch.hpp`

5.55 **Catch::RegistrarForTagAliases** Struct Reference

## 5.55.1 \*

## Public Member Functions

- **RegistrarForTagAliases** (char const \*alias, char const \*tag, [SourceLineInfo](#) const &lineInfo)

The documentation for this struct was generated from the following file:

- `dep/catch.hpp`

5.56 **Catch::ResultDisposition** Struct Reference

## 5.56.1 \*

## Public Types

- enum **Flags** { **Normal** = 0x01, **ContinueOnFailure** = 0x02, **FalseTest** = 0x04, **SuppressFail** = 0x08 }

The documentation for this struct was generated from the following file:

- `dep/catch.hpp`

5.57 **Catch::ResultWas** Struct Reference

## 5.57.1 \*

## Public Types

- enum **OfType** {  
**Unknown** = -1, **Ok** = 0, **Info** = 1, **Warning** = 2,  
**FailureBit** = 0x10, **ExpressionFailed** = FailureBit | 1, **ExplicitFailure** = FailureBit | 2, **Exception** = 0x100 |  
FailureBit,  
**ThrewException** = Exception | 1, **DidntThrowException** = Exception | 2, **FatalErrorCondition** = 0x200 |  
FailureBit }

The documentation for this struct was generated from the following file:

- `dep/catch.hpp`

## 5.58 Catch::ReusableStringStream Class Reference

### 5.58.1 \*

#### Public Member Functions

- auto **str** () const -> std::string
- template<typename T >  
auto **operator**<< (T const &value) -> [ReusableStringStream](#) &
- auto **get** () -> std::ostream &

### 5.58.2 \*

#### Static Public Member Functions

- static void **cleanup** ()

The documentation for this class was generated from the following file:

- dep/catch.hpp

## 5.59 Catch::ScopedMessage Class Reference

### 5.59.1 \*

#### Public Member Functions

- **ScopedMessage** ([MessageBuilder](#) const &builder)

### 5.59.2 \*

#### Public Attributes

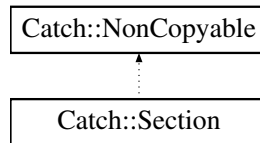
- [MessageInfo](#) **m\_info**

The documentation for this class was generated from the following file:

- dep/catch.hpp

## 5.60 Catch::Section Class Reference

Inheritance diagram for Catch::Section:



### 5.60.1 \*

Public Member Functions

- **Section** ([SectionInfo](#) const &info)
- **operator bool** () const

The documentation for this class was generated from the following file:

- `dep/catch.hpp`

## 5.61 Catch::SectionEndInfo Struct Reference

### 5.61.1 \*

Public Member Functions

- **SectionEndInfo** ([SectionInfo](#) const &\_sectionInfo, [Counts](#) const &\_prevAssertions, double \_durationIn↵  
Seconds)

### 5.61.2 \*

Public Attributes

- [SectionInfo](#) **sectionInfo**
- [Counts](#) **prevAssertions**
- double **durationInSeconds**

The documentation for this struct was generated from the following file:

- `dep/catch.hpp`

## 5.62 Catch::SectionInfo Struct Reference

### 5.62.1 \*

#### Public Member Functions

- **SectionInfo** ([SourceLineInfo](#) const &\_lineInfo, std::string const &\_name, std::string const &\_↵ description=std::string())

### 5.62.2 \*

#### Public Attributes

- std::string **name**
- std::string **description**
- [SourceLineInfo](#) **lineInfo**

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.63 Catch::SourceLineInfo Struct Reference

### 5.63.1 \*

#### Public Member Functions

- **SourceLineInfo** (char const \*\_file, std::size\_t \_line) noexcept
- **SourceLineInfo** ([SourceLineInfo](#) const &other)=default
- **SourceLineInfo** ([SourceLineInfo](#) &&)=default
- [SourceLineInfo](#) & **operator=** ([SourceLineInfo](#) const &)=default
- [SourceLineInfo](#) & **operator=** ([SourceLineInfo](#) &&)=default
- bool **empty** () const noexcept
- bool **operator==** ([SourceLineInfo](#) const &other) const noexcept
- bool **operator<** ([SourceLineInfo](#) const &other) const noexcept

### 5.63.2 \*

#### Public Attributes

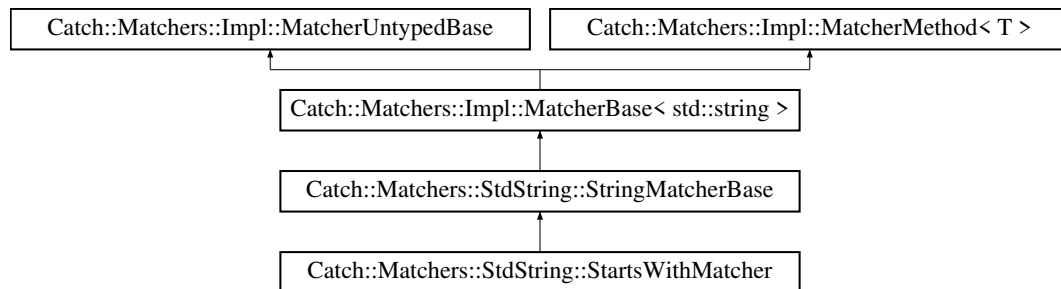
- char const \* **file**
- std::size\_t **line**

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.64 Catch::Matchers::StdString::StartsWithMatcher Struct Reference

Inheritance diagram for Catch::Matchers::StdString::StartsWithMatcher:



### 5.64.1 \*

Public Member Functions

- **StartsWithMatcher** ([CasedString](#) const &comparator)
- bool **match** (std::string const &source) const override

### 5.64.2 \*

Additional Inherited Members

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.65 Catch::StreamEndStop Struct Reference

### 5.65.1 \*

Public Member Functions

- std::string **operator+** () const

The documentation for this struct was generated from the following file:

- dep/catch.hpp



## 5.66 `Catch::StringMaker< T, typename >` Struct Template Reference

### 5.66.1 \*

Static Public Member Functions

- `template<typename Fake = T>`  
`static std::enable_if<::Catch::Detail::IsStreamInsertable< Fake >::value, std::string >::type` **convert** (const Fake &value)
- `template<typename Fake = T>`  
`static std::enable_if<!::Catch::Detail::IsStreamInsertable< Fake >::value, std::string >::type` **convert** (const Fake &value)

The documentation for this struct was generated from the following file:

- `dep/catch.hpp`

## 5.67 `Catch::StringMaker< bool >` Struct Template Reference

### 5.67.1 \*

Static Public Member Functions

- `static std::string` **convert** (bool b)

The documentation for this struct was generated from the following file:

- `dep/catch.hpp`

## 5.68 `Catch::StringMaker< Catch::Detail::Approx >` Struct Template Reference

### 5.68.1 \*

Static Public Member Functions

- `static std::string` **convert** (`Catch::Detail::Approx` const &value)

The documentation for this struct was generated from the following file:

- `dep/catch.hpp`

## 5.69 **Catch::StringMaker**< char \* > Struct Template Reference

### 5.69.1 \*

Static Public Member Functions

- static std::string **convert** (char \*str)

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.70 **Catch::StringMaker**< char > Struct Template Reference

### 5.70.1 \*

Static Public Member Functions

- static std::string **convert** (char c)

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.71 **Catch::StringMaker**< char const \* > Struct Template Reference

### 5.71.1 \*

Static Public Member Functions

- static std::string **convert** (char const \*str)

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.72 **Catch::StringMaker**< char[SZ]> Struct Template Reference

### 5.72.1 \*

Static Public Member Functions

- static std::string **convert** (const char \*str)

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.73 Catch::StringMaker< double > Struct Template Reference

### 5.73.1 \*

Static Public Member Functions

- static std::string **convert** (double value)

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.74 Catch::StringMaker< float > Struct Template Reference

### 5.74.1 \*

Static Public Member Functions

- static std::string **convert** (float value)

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.75 Catch::StringMaker< int > Struct Template Reference

### 5.75.1 \*

Static Public Member Functions

- static std::string **convert** (int value)

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.76 Catch::StringMaker< long > Struct Template Reference

### 5.76.1 \*

Static Public Member Functions

- static std::string **convert** (long value)

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.77 `Catch::StringMaker< long long >` Struct Template Reference

### 5.77.1 \*

Static Public Member Functions

- static `std::string convert` (long long value)

The documentation for this struct was generated from the following file:

- `dep/catch.hpp`

## 5.78 `Catch::StringMaker< R C::* >` Struct Template Reference

### 5.78.1 \*

Static Public Member Functions

- static `std::string convert` (R C::\*p)

The documentation for this struct was generated from the following file:

- `dep/catch.hpp`

## 5.79 `Catch::StringMaker< R, typename std::enable_if< is_range< R >::value &&!::Catch::Detail::IsStreamInsertable< R >::value >::type >` Struct Template Reference

### 5.79.1 \*

Static Public Member Functions

- static `std::string convert` (R const &range)

The documentation for this struct was generated from the following file:

- `dep/catch.hpp`

## 5.80 Catch::StringMaker< signed char > Struct Template Reference

### 5.80.1 \*

Static Public Member Functions

- static std::string **convert** (signed char c)

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.81 Catch::StringMaker< signed char[SZ]> Struct Template Reference

### 5.81.1 \*

Static Public Member Functions

- static std::string **convert** (const char \*str)

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.82 Catch::StringMaker< std::nullptr\_t > Struct Template Reference

### 5.82.1 \*

Static Public Member Functions

- static std::string **convert** (std::nullptr\_t)

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.83 Catch::StringMaker< std::string > Struct Template Reference

### 5.83.1 \*

Static Public Member Functions

- static std::string **convert** (const std::string &str)

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.84 Catch::StringMaker< std::wstring > Struct Template Reference

### 5.84.1 \*

Static Public Member Functions

- static std::string **convert** (const std::wstring &wstr)

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.85 Catch::StringMaker< T \* > Struct Template Reference

### 5.85.1 \*

Static Public Member Functions

- template<typename U >  
static std::string **convert** (U \*p)

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.86 Catch::StringMaker< T[SZ]> Struct Template Reference

### 5.86.1 \*

Static Public Member Functions

- static std::string **convert** (T const(&arr)[SZ])

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.87 Catch::StringMaker< unsigned char > Struct Template Reference

### 5.87.1 \*

Static Public Member Functions

- static std::string **convert** (unsigned char c)

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.88 Catch::StringMaker< unsigned char[SZ]> Struct Template Reference

### 5.88.1 \*

Static Public Member Functions

- static std::string **convert** (const char \*str)

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.89 Catch::StringMaker< unsigned int > Struct Template Reference

### 5.89.1 \*

Static Public Member Functions

- static std::string **convert** (unsigned int value)

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.90 Catch::StringMaker< unsigned long > Struct Template Reference

### 5.90.1 \*

Static Public Member Functions

- static std::string **convert** (unsigned long value)

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.91 Catch::StringMaker< unsigned long long > Struct Template Reference

### 5.91.1 \*

Static Public Member Functions

- static std::string **convert** (unsigned long long value)

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.92 Catch::StringMaker< wchar\_t \* > Struct Template Reference

### 5.92.1 \*

Static Public Member Functions

- static std::string **convert** (wchar\_t \*str)

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.93 Catch::StringMaker< wchar\_t const \* > Struct Template Reference

### 5.93.1 \*

Static Public Member Functions

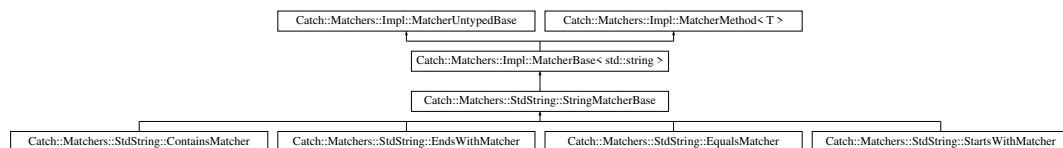
- static std::string **convert** (wchar\_t const \*str)

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.94 Catch::Matchers::StdString::StringMatcherBase Struct Reference

Inheritance diagram for Catch::Matchers::StdString::StringMatcherBase:



### 5.94.1 \*

Public Member Functions

- **StringMatcherBase** (std::string const &operation, [CasedString](#) const &comparator)
- std::string **describe** () const override

### 5.94.2 \*

Public Attributes

- [CasedString](#) **m\_comparator**
- std::string **m\_operation**



## 5.94.3 \*

Additional Inherited Members

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.95 Catch::StringRef Class Reference

```
#include <catch.hpp>
```

## 5.95.1 \*

Public Types

- using **size\_type** = std::size\_t

## 5.95.2 \*

Public Member Functions

- **StringRef** ([StringRef](#) const &other) noexcept
- **StringRef** ([StringRef](#) &&other) noexcept
- **StringRef** (char const \*rawChars) noexcept
- **StringRef** (char const \*rawChars, size\_type size) noexcept
- **StringRef** (std::string const &stdString) noexcept
- auto **operator=** ([StringRef](#) const &other) noexcept -> [StringRef](#) &
- **operator std::string** () const
- void **swap** ([StringRef](#) &other) noexcept
- auto **operator==** ([StringRef](#) const &other) const noexcept -> bool
- auto **operator!=** ([StringRef](#) const &other) const noexcept -> bool
- auto **operator[]** (size\_type index) const noexcept -> char
- auto **empty** () const noexcept -> bool
- auto **size** () const noexcept -> size\_type
- auto **numberOfCharacters** () const noexcept -> size\_type
- auto **c\_str** () const -> char const \*
- auto **substr** (size\_type start, size\_type size) const noexcept -> [StringRef](#)

## 5.95.3 \*

Friends

- struct **StringRefTestAccess**

### 5.95.4 Detailed Description

A non-owning string class (similar to the forthcoming `std::string_view`) Note that, because a [StringRef](#) may be a substring of another string, it may not be null terminated. `c_str()` must return a null terminated string, however, and so the [StringRef](#) will internally take ownership (taking a copy), if necessary. In theory this ownership is not externally visible - but it does mean (substring) `StringRef`s should not be shared between threads.

The documentation for this class was generated from the following file:

- `dep/catch.hpp`

## 5.96 SupplyRunner Class Reference

Gets Ingredients from the Storeroom.

```
#include <SupplyRunner.hpp>
```

### 5.96.1 \*

Public Member Functions

- [SupplyRunner](#) (Storeroom &storeroom\_)  
*Constructs a [SupplyRunner](#) with the Storeroom it gets its Ingredients from.*
- `std::vector< Ingredient >` [get\\_ingredients](#) (IngredientMap ingredients)  
*Gets ingredients from the Storeroom, if they exist, as a vector.*

### 5.96.2 Detailed Description

Gets Ingredients from the Storeroom.

### 5.96.3 Member Function Documentation

#### 5.96.3.1 `get_ingredients()`

```
std::vector<Ingredient> SupplyRunner::get_ingredients (
    IngredientMap ingredients )
```

Gets ingredients from the Storeroom, if they exist, as a vector.

Note

Will remove ingredients from the storeroom if all the ingredeints asked for are all found in sufficient numbers.

Exceptions

<i>const</i>	<code>char*</code> if there are not enough ingredients in the Storeroom.
--------------	--------------------------------------------------------------------------

## Note

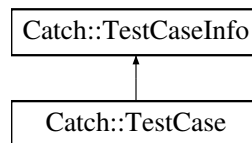
Upon not finding enough ingredients to return, the [SupplyRunner](#) will not modify the Storeroom in any way, including removing Ingredients from the Storeroom (decreasing the Ingredient count on the map).

The documentation for this class was generated from the following file:

- include/SupplyRunner.hpp

## 5.97 Catch::TestCase Class Reference

Inheritance diagram for Catch::TestCase:



### 5.97.1 \*

#### Public Member Functions

- **TestCase** ([ITestInvoker](#) \*testCase, [TestCaseInfo](#) const &info)
- **TestCase** **withName** (std::string const &\_newName) const
- void **invoke** () const
- [TestCaseInfo](#) const & **getTestCaseInfo** () const
- bool **operator==** ([TestCase](#) const &other) const
- bool **operator<** ([TestCase](#) const &other) const

### 5.97.2 \*

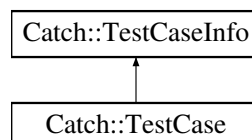
#### Additional Inherited Members

The documentation for this class was generated from the following file:

- dep/catch.hpp

## 5.98 Catch::TestCaseInfo Struct Reference

Inheritance diagram for Catch::TestCaseInfo:



### 5.98.1 \*

#### Public Types

- enum **SpecialProperties** {  
**None** = 0, **IsHidden** = 1 << 1, **ShouldFail** = 1 << 2, **MayFail** = 1 << 3,  
**Throws** = 1 << 4, **NonPortable** = 1 << 5, **Benchmark** = 1 << 6 }

### 5.98.2 \*

#### Public Member Functions

- **TestCaseInfo** (std::string const &\_name, std::string const &\_className, std::string const &\_description, std::vector< std::string > const &\_tags, [SourceLineInfo](#) const &\_lineInfo)
- bool **isHidden** () const
- bool **throws** () const
- bool **okToFail** () const
- bool **expectedToFail** () const
- std::string **tagsAsString** () const

### 5.98.3 \*

#### Public Attributes

- std::string **name**
- std::string **className**
- std::string **description**
- std::vector< std::string > **tags**
- std::vector< std::string > **lcaseTags**
- [SourceLineInfo](#) **lineInfo**
- SpecialProperties **properties**

### 5.98.4 \*

#### Friends

- void **setTags** ([TestCaseInfo](#) &testCaseInfo, std::vector< std::string > tags)

The documentation for this struct was generated from the following file:

- dep/catch.hpp

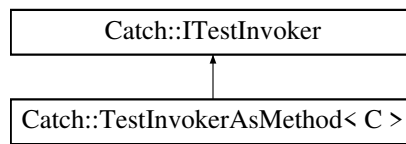
## 5.99 Catch::TestFailureException Struct Reference

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.100 Catch::TestInvokerAsMethod< C > Class Template Reference

Inheritance diagram for Catch::TestInvokerAsMethod< C >:



### 5.100.1 \*

Public Member Functions

- **TestInvokerAsMethod** (void(C::\*testAsMethod)()) noexcept
- void **invoke** () const override

The documentation for this class was generated from the following file:

- dep/catch.hpp

## 5.101 Catch::Timer Class Reference

### 5.101.1 \*

Public Member Functions

- void **start** ()
- auto **getElapsedNanoseconds** () const -> uint64\_t
- auto **getElapsedMicroseconds** () const -> uint64\_t
- auto **getElapsedMilliseconds** () const -> unsigned int
- auto **getElapsedSeconds** () const -> double

The documentation for this class was generated from the following file:

- dep/catch.hpp

## 5.102 Catch::Totals Struct Reference

### 5.102.1 \*

Public Member Functions

- **Totals operator-** (Totals const &other) const
- **Totals & operator+=** (Totals const &other)
- **Totals delta** (Totals const &prevTotals) const

### 5.102.2 \*

Public Attributes

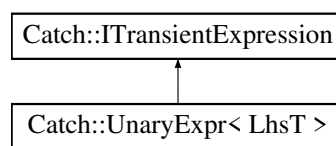
- [Counts](#) **assertions**
- [Counts](#) **testCases**

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.103 **Catch::UnaryExpr< LhsT >** Class Template Reference

Inheritance diagram for **Catch::UnaryExpr< LhsT >**:



### 5.103.1 \*

Public Member Functions

- **UnaryExpr** (LhsT lhs)

### 5.103.2 \*

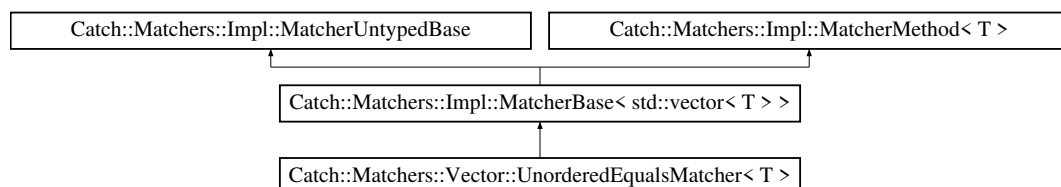
Additional Inherited Members

The documentation for this class was generated from the following file:

- dep/catch.hpp

## 5.104 **Catch::Matchers::Vector::UnorderedEqualsMatcher< T >** Struct Template Reference

Inheritance diagram for **Catch::Matchers::Vector::UnorderedEqualsMatcher< T >**:



### 5.104.1 \*

Public Member Functions

- **UnorderedEqualsMatcher** (std::vector< T > const &target)
- bool **match** (std::vector< T > const &vec) const override
- std::string **describe** () const override

## 5.104.2 \*

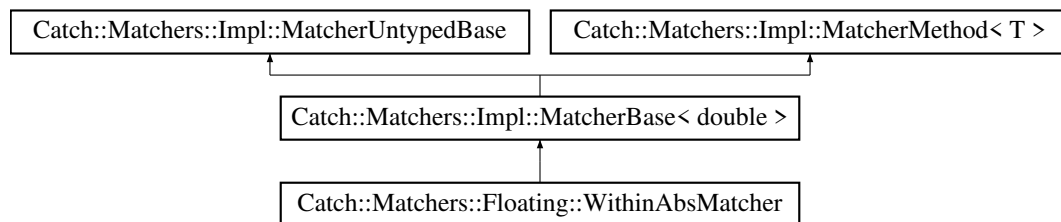
## Additional Inherited Members

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.105 Catch::Matchers::Floating::WithinAbsMatcher Struct Reference

Inheritance diagram for Catch::Matchers::Floating::WithinAbsMatcher:



## 5.105.1 \*

## Public Member Functions

- **WithinAbsMatcher** (double target, double margin)
- bool **match** (double const &matchee) const override
- std::string **describe** () const override

## 5.105.2 \*

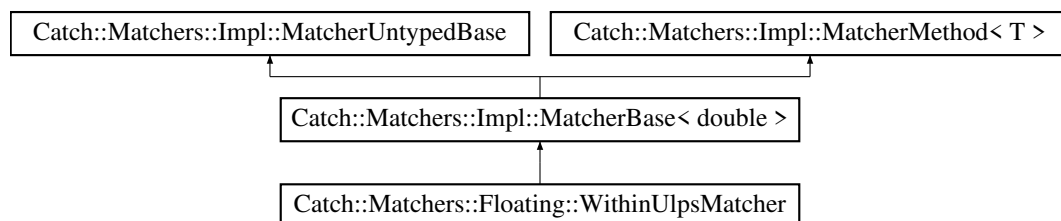
## Additional Inherited Members

The documentation for this struct was generated from the following file:

- dep/catch.hpp

## 5.106 Catch::Matchers::Floating::WithinUlpsMatcher Struct Reference

Inheritance diagram for Catch::Matchers::Floating::WithinUlpsMatcher:



## 5.106.1 \*

## Public Member Functions

- **WithinUlpsMatcher** (double target, int ulps, FloatingPointKind baseType)
- bool **match** (double const &matchee) const override
- std::string **describe** () const override

### 5.106.2 \*

#### Additional Inherited Members

The documentation for this struct was generated from the following file:

- `dep/catch.hpp`



# Index

Cashier, 17  
    serve\_customer, 18  
Catch::AssertionHandler, 14  
Catch::AssertionInfo, 15  
Catch::AssertionReaction, 15  
Catch::AutoReg, 15  
Catch::BenchmarkLooper, 16  
Catch::BinaryExpr< LhsT, RhsT >, 16  
Catch::CaseSensitive, 17  
Catch::Counts, 22  
Catch::Decomposer, 24  
Catch::Detail::Approx, 13  
Catch::Detail::IsStreamInsertable< T >, 29  
Catch::ExceptionTranslatorRegistrar, 26  
Catch::ExprLhs< LhsT >, 26  
Catch::IExceptionTranslator, 27  
Catch::IExceptionTranslatorRegistry, 27  
Catch::IMutableRegistryHub, 27  
Catch::IRegistryHub, 28  
Catch::IResultCapture, 28  
Catch::IRunner, 29  
Catch::is\_range< T >, 29  
    value, 29  
Catch::IStream, 30  
Catch::ITestCaseRegistry, 30  
Catch::ITestInvoker, 30  
Catch::ITransientExpression, 31  
Catch::LazyExpression, 32  
Catch::Matchers::Floating::WithinAbsMatcher, 61  
Catch::Matchers::Floating::WithinUlpMatcher, 61  
Catch::Matchers::Impl::MatchAllOf< ArgT >, 33  
Catch::Matchers::Impl::MatchAnyOf< ArgT >, 34  
Catch::Matchers::Impl::MatcherBase< T >, 34  
Catch::Matchers::Impl::MatcherMethod< ObjectT >, 35  
Catch::Matchers::Impl::MatcherMethod< PtrT \* >, 35  
Catch::Matchers::Impl::MatcherUntypedBase, 36  
Catch::Matchers::Impl::MatchNotOf< ArgT >, 37  
Catch::Matchers::StdString::CasedString, 16  
Catch::Matchers::StdString::ContainsMatcher, 19  
Catch::Matchers::StdString::EndsWithMatcher, 24  
Catch::Matchers::StdString::EqualsMatcher, 25  
Catch::Matchers::StdString::RegexMatcher, 41  
Catch::Matchers::StdString::StartsWithMatcher, 46  
Catch::Matchers::StdString::StringMatcherBase, 54  
Catch::Matchers::Vector::ContainsElementMatcher< T  
    >, 19  
Catch::Matchers::Vector::ContainsMatcher< T >, 20  
Catch::Matchers::Vector::EqualsMatcher< T >, 25  
Catch::Matchers::Vector::UnorderedEqualsMatcher< T  
    >, 60  
Catch::MatchExpr< ArgT, MatcherT >, 36  
Catch::MessageBuilder, 38  
Catch::MessageInfo, 38  
Catch::MessageStream, 39  
Catch::NameAndTags, 39  
Catch::NonCopyable, 40  
Catch::not\_this\_one, 40  
Catch::pluralise, 41  
Catch::RegistrarForTagAliases, 42  
Catch::ResultDisposition, 42  
Catch::ResultWas, 42  
Catch::ReusableStringStream, 43  
Catch::ScopedMessage, 43  
Catch::Section, 44  
Catch::SectionEndInfo, 44  
Catch::SectionInfo, 45  
Catch::SourceLineInfo, 45  
Catch::StreamEndStop, 46  
Catch::StringMaker< bool >, 47  
Catch::StringMaker< Catch::Detail::Approx >, 47  
Catch::StringMaker< char >, 48  
Catch::StringMaker< char \* >, 48  
Catch::StringMaker< char const \* >, 48  
Catch::StringMaker< char[SZ]>, 48  
Catch::StringMaker< double >, 49  
Catch::StringMaker< float >, 49  
Catch::StringMaker< int >, 49  
Catch::StringMaker< long >, 49  
Catch::StringMaker< long long >, 50  
Catch::StringMaker< R C::\* >, 50  
Catch::StringMaker< R, typename std::enable\_if<  
    is\_range< R >::value &&!Catch::Detail::IsStreamInsertable<  
    R >::value >::type >, 50  
Catch::StringMaker< signed char >, 51  
Catch::StringMaker< signed char[SZ]>, 51  
Catch::StringMaker< std::nullptr\_t >, 51  
Catch::StringMaker< std::string >, 51  
Catch::StringMaker< std::wstring >, 52  
Catch::StringMaker< T \* >, 52  
Catch::StringMaker< T, typename >, 47  
Catch::StringMaker< T[SZ]>, 52  
Catch::StringMaker< unsigned char >, 52  
Catch::StringMaker< unsigned char[SZ]>, 53  
Catch::StringMaker< unsigned int >, 53  
Catch::StringMaker< unsigned long >, 53  
Catch::StringMaker< unsigned long long >, 53  
Catch::StringMaker< wchar\_t \* >, 54  
Catch::StringMaker< wchar\_t const \* >, 54

- Catch::[StringRef](#), [55](#)
- Catch::[TestCase](#), [57](#)
- Catch::[TestCaseInfo](#), [57](#)
- Catch::[TestFailureException](#), [58](#)
- Catch::[TestInvokerAsMethod< C >](#), [59](#)
- Catch::[Timer](#), [59](#)
- Catch::[Totals](#), [59](#)
- Catch::[UnaryExpr< LhsT >](#), [60](#)
- Catch\_global\_namespace\_dummy, [18](#)
- charge\_money
  - Customer, [23](#)
- Cook, [21](#)
  - prepare\_dish, [21](#)
- Customer, [22](#)
  - charge\_money, [23](#)
  - expel, [23](#)
  - get\_order, [23](#)
  - refund\_money, [23](#)
- expel
  - Customer, [23](#)
- get\_ingredients
  - SupplyRunner, [56](#)
- get\_order
  - Customer, [23](#)
- Kitchen, [31](#)
  - prepare\_dish, [32](#)
- Order, [40](#)
- prepare\_dish
  - Cook, [21](#)
  - Kitchen, [32](#)
- refund\_money
  - Customer, [23](#)
- serve\_customer
  - Cashier, [18](#)
- SupplyRunner, [56](#)
  - get\_ingredients, [56](#)
- value
  - Catch::[is\\_range< T >](#), [29](#)