

---

# **mbtest Documentation**

***Release 2.5.0***

**Simon Brunning**

**May 06, 2020**



CONTENTS:

<b>1</b>	<b>API Reference</b>	<b>3</b>
1.1	The <i>mbtest.server</i> module . . . . .	3
1.2	The <i>mbtest.imposters.imposters</i> module . . . . .	3
1.3	The <i>mbtest.imposters.stubs</i> module . . . . .	4
1.4	The <i>mbtest.imposters.predicates</i> module . . . . .	4
1.5	The <i>mbtest.imposters.responses</i> module . . . . .	7
1.6	The <i>mbtest.imposters.behaviors.copy</i> module . . . . .	9
1.7	The <i>mbtest.imposters.behaviors.lookup</i> module . . . . .	10
1.8	The <i>mbtest.imposters.behaviors.using</i> module . . . . .	11
1.9	The <i>mbtest.matchers</i> module . . . . .	12
1.10	The <i>mbtest.imposters.base</i> module . . . . .	12
<b>2</b>	<b>Indices and tables</b>	<b>13</b>
<b>3</b>	<b>Installation</b>	<b>15</b>
<b>4</b>	<b>Usage</b>	<b>17</b>
<b>5</b>	<b>Indices and tables</b>	<b>19</b>
	<b>Python Module Index</b>	<b>21</b>
	<b>Index</b>	<b>23</b>



Opinionated Python wrapper & utils for the [Mountebank](#) over the wire test double tool.  
Includes [pytest](#) fixture and [PyHamcrest](#) matchers.



## API REFERENCE

1.1 The *mbtest.server* module1.2 The *mbtest.imposters.imposters* module

**class** `mbtest.imposters.imposters.Imposter` (*stubs*, *port=None*, *protocol=<Protocol.HTTP: 'http'>*, *name=None*, *record\_requests=True*)

Represents a [Mountebank imposter](#). Think of an imposter as a mock website, running a protocol, on a specific port. Required behaviors are specified using stubs.

**Parameters**

- **stubs** (`Union[Stub, Iterable[Stub]]`) – One or more Stubs.
- **port** (`Optional[int]`) – Port.
- **protocol** (`Protocol`) – Protocol to run on.
- **name** (`Optional[str]`) – Impostor name - useful for interactive exploration of impostors on <http://localhost:2525/impostors>
- **record\_requests** (`bool`) – Record requests made against this impostor, so they can be asserted against later.

**class** `Protocol`

Imposter `Protocol`.

`HTTP = 'http'`

`HTTPS = 'https'`

`SMTP = 'smtp'`

`TCP = 'tcp'`

**property** `host`

Return type `str`

**property** `url`

Return type `furl`

**as\_structure** ()

Converted to a JSON serializable structure.

Return type `Any`

**Returns** Structure suitable for JSON serialisation.

**static from\_structure** (*structure*)

Converted from a JSON serializable structure.

**Parameters** **structure** (*Any*) – JSON structure to be converted.

**Return type** *Imposter*

**Returns** Converted object.

`mbtest.imposters.imposters.smtp_imposter (name='smtp', record_requests=True)`

Canned SMTP server imposter.

**Return type** *Imposter*

## 1.3 The *mbtest.imposters.stubs* module

**class** `mbtest.imposters.stubs.Stub` (*predicates=None, responses=None*)

Represents a [Mountebank stub](#). Think of a stub as a behavior, triggered by a matching predicate.

**Parameters**

- **predicates** (*Union[BasePredicate, Iterable[BasePredicate], None]*) – Trigger this stub if one of these predicates matches the request
- **responses** (*Union[BaseResponse, Iterable[BaseResponse], None]*) – Use these response behaviors (in order)

**as\_structure** ()

Converted to a JSON serializable structure.

**Return type** *Any*

**Returns** Structure suitable for JSON serialisation.

**static from\_structure** (*structure*)

Converted from a JSON serializable structure.

**Parameters** **structure** (*Any*) – JSON structure to be converted.

**Return type** *Stub*

**Returns** Converted object.

## 1.4 The *mbtest.imposters.predicates* module

**class** `mbtest.imposters.predicates.BasePredicate`

**static from\_structure** (*structure*)

Converted from a JSON serializable structure.

**Parameters** **structure** (*Any*) – JSON structure to be converted.

**Return type** *BasePredicate*

**Returns** Converted object.

**class** `mbtest.imposters.predicates.LogicallyCombinablePredicate`



```
class mbtest.imposters.predicates.Predicate (path=None, method=None, query=None,
                                             body=None, headers=None, xpath=None,
                                             operator=<Operator.EQUALS: 'equals'>,
                                             case_sensitive=True)
```

Represents a [Mountebank predicate](#). A predicate can be thought of as a trigger, which may or may not match a request.

#### Parameters

- **path** ([Union](#)[[str](#), [furl](#), [None](#)]) – URL path.
- **method** ([Optional](#)[[Method](#)]) – HTTP method.
- **query** ([Optional](#)[[Mapping](#)[[str](#), [Union](#)[[str](#), [int](#), [bool](#)]]]) – Query arguments, keys and values.
- **body** ([Optional](#)[[str](#)]) – Body text. Can be a string, or a JSON serialisable data structure.
- **headers** ([Optional](#)[[Mapping](#)[[str](#), [str](#)]]) – Headers, keys and values.
- **xpath** ([Optional](#)[[str](#)]) – xpath query
- **operator** ([Operator](#)) –
- **case\_sensitive** ([bool](#)) –

```
exception InvalidPredicateOperator
```

```
class Method
```

Predicate HTTP method.

```
DELETE = 'DELETE'
```

```
GET = 'GET'
```

```
HEAD = 'HEAD'
```

```
POST = 'POST'
```

```
PUT = 'PUT'
```

```
class Operator
```

Predicate operator.

```
EQUALS = 'equals'
```

```
DEEP_EQUALS = 'deepEquals'
```

```
CONTAINS = 'contains'
```

```
STARTS_WITH = 'startsWith'
```

```
ENDS_WITH = 'endsWith'
```

```
MATCHES = 'matches'
```

```
EXISTS = 'exists'
```

```
classmethod has_value (name)
```

Return type [bool](#)

```
as_structure ()
```

Converted to a JSON serializable structure.

Return type [Any](#)

Returns Structure suitable for JSON serialisation.

**static from\_structure** (*structure*)

Converted from a JSON serializable structure.

**Parameters** **structure** (*Any*) – JSON structure to be converted.

**Return type** *Predicate*

**Returns** Converted object.

**class** mbtest.imposters.predicates.**AndPredicate** (*left, right*)

**as\_structure** ()

Converted to a JSON serializable structure.

**Return type** *Any*

**Returns** Structure suitable for JSON serialisation.

**static from\_structure** (*structure*)

Converted from a JSON serializable structure.

**Parameters** **structure** (*Any*) – JSON structure to be converted.

**Return type** *AndPredicate*

**Returns** Converted object.

**class** mbtest.imposters.predicates.**OrPredicate** (*left, right*)

**as\_structure** ()

Converted to a JSON serializable structure.

**Return type** *Any*

**Returns** Structure suitable for JSON serialisation.

**static from\_structure** (*structure*)

Converted from a JSON serializable structure.

**Parameters** **structure** (*Any*) – JSON structure to be converted.

**Return type** *OrPredicate*

**Returns** Converted object.

**class** mbtest.imposters.predicates.**TcpPredicate** (*data*)

TODO

**as\_structure** ()

Converted to a JSON serializable structure.

**Return type** *Any*

**Returns** Structure suitable for JSON serialisation.

**static from\_structure** (*structure*)

Converted from a JSON serializable structure.

**Parameters** **structure** (*Any*) – JSON structure to be converted.

**Return type** *TcpPredicate*

**Returns** Converted object.

```
class mbtest.imposters.predicates.InjectionPredicate (inject)
    TODO

    Injection requires Mountebank version 2.0 or higher. http://www.mbtest.org/docs/api/injection

    as_structure ()
        Converted to a JSON serializable structure.

        Return type Any

        Returns Structure suitable for JSON serialisation.

    static from_structure (structure)
        Converted from a JSON serializable structure.

        Parameters structure (Any) – JSON structure to be converted.

        Return type InjectionPredicate

        Returns Converted object.
```

## 1.5 The *mbtest.imposters.responses* module

```
class mbtest.imposters.responses.BaseResponse

    static from_structure (structure)
        Converted from a JSON serializable structure.

        Parameters structure (Any) – JSON structure to be converted.

        Return type BaseResponse

        Returns Converted object.

class mbtest.imposters.responses.Response (body="", status_code=200, wait=None, re-
                                           peat=None, headers=None, mode=None,
                                           copy=None, decorate=None, lookup=None,
                                           shell_transform=None)

    Represents a Mountebank 'is' response behavior.

    Parameters

    • body (str) – Body text for response. Can be a string, or a JSON serialisable data structure.
    • status_code (Union[int, str]) – HTTP status code
    • wait (Union[int, str, None]) – Add latency, in ms.
    • repeat (Optional[int]) – Repeat this many times before moving on to next response.
    • headers (Optional[Mapping[str, str]]) – Response HTTP headers
    • mode (Optional[Mode]) – Mode - text or binary
    • copy (Optional[Copy]) – Copy behavior
    • decorate (Optional[str]) – Decorate behavior.
    • lookup (Optional[Lookup]) – Lookup behavior
    • shell_transform (Union[str, Iterable[str], None]) – shellTransform behavior
```

```
class Mode
    An enumeration.

    TEXT = 'text'

    BINARY = 'binary'

property body
    Return type str

as_structure()
    Converted to a JSON serializable structure.

    Return type Any

    Returns Structure suitable for JSON serialisation.

static from_structure(structure)
    Converted from a JSON serializable structure.

    Parameters structure (Any) – JSON structure to be converted.

    Return type Response

    Returns Converted object.

class mbtest.imposters.responses.TcpResponse(data)

as_structure()
    Converted to a JSON serializable structure.

    Return type Any

    Returns Structure suitable for JSON serialisation.

static from_structure(structure)
    Converted from a JSON serializable structure.

    Parameters structure (Any) – JSON structure to be converted.

    Return type TcpResponse

    Returns Converted object.

class mbtest.imposters.responses.Proxy(to, wait=None, inject_headers=None,
                                       mode=<Mode.ALWAYS: 'proxyAlways'>)
    Represents a Mountebank proxy.

    Parameters to (Union[furl, str]) – The origin server, to which the request should proxy.

class Mode
    Defines the replay behavior of the proxy.

    ONCE = 'proxyOnce'

    ALWAYS = 'proxyAlways'

    TRANSPARENT = 'proxyTransparent'

as_structure()
    Converted to a JSON serializable structure.

    Return type Any

    Returns Structure suitable for JSON serialisation.
```

**static from\_structure** (*structure*)

Converted from a JSON serializable structure.

**Parameters** **structure** (*Any*) – JSON structure to be converted.

**Return type** *Proxy*

**Returns** Converted object.

**class** `mbtest.imposters.responses.InjectionResponse` (*inject*)

TODO

Injection requires Mountebank version 2.0 or higher. <http://www.mbtest.org/docs/api/injection>

**as\_structure** ()

Converted to a JSON serializable structure.

**Return type** *Any*

**Returns** Structure suitable for JSON serialisation.

**static from\_structure** (*structure*)

Converted from a JSON serializable structure.

**Parameters** **structure** (*Any*) – JSON structure to be converted.

**Return type** *InjectionResponse*

**Returns** Converted object.

## 1.6 The *mbtest.imposters.behaviors.copy* module

**class** `mbtest.imposters.behaviors.copy.Copy` (*from\_, into, using*)

Represents a *copy behavior*.

**Parameters**

- **from** – The name of the request field to copy from, or, if the request field is an object, then an object specifying the path to the request field.
- **into** (*str*) – The token to replace in the response with the selected request value.
- **using** (*Using*) – The configuration needed to select values from the response.

**as\_structure** ()

Converted to a JSON serializable structure.

**Return type** *Any*

**Returns** Structure suitable for JSON serialisation.

**static from\_structure** (*structure*)

Converted from a JSON serializable structure.

**Parameters** **structure** (*Any*) – JSON structure to be converted.

**Return type** *Copy*

**Returns** Converted object.

## 1.7 The *mbtest.imposters.behaviors.lookup* module

**class** `mbtest.imposters.behaviors.lookup.Lookup` (*key*, *datasource\_path*, *datasource\_key\_column*, *into*)

Represents a `lookup` behavior.

**Parameters**

- **key** (*Key*) – How to select the key from the request.
- **datasource\_path** (*Union[str, Path]*) – The path to the data source.
- **datasource\_key\_column** (*str*) – The header of the column to match against the key.
- **into** (*str*) – The token to replace in the response with the selected request value.

**as\_structure** ()

Converted to a JSON serializable structure.

**Return type** *Any*

**Returns** Structure suitable for JSON serialisation.

**static from\_structure** (*structure*)

Converted from a JSON serializable structure.

**Parameters** **structure** (*Any*) – JSON structure to be converted.

**Return type** *Lookup*

**Returns** Converted object.

**class** `mbtest.imposters.behaviors.lookup.Key` (*from\_*, *using*, *index=0*)

The information on how to select the key from the request.

**Parameters**

- **from** – The name of the request field to copy from, or, if the request field is an object, then an object specifying the path to the request field.
- **using** (*Using*) – The configuration needed to select values from the response
- **index** (*int*) – Index of the item from the result array to be selected.

**as\_structure** ()

Converted to a JSON serializable structure.

**Return type** *Any*

**Returns** Structure suitable for JSON serialisation.

**static from\_structure** (*structure*)

Converted from a JSON serializable structure.

**Parameters** **structure** (*Any*) – JSON structure to be converted.

**Return type** *Key*

**Returns** Converted object.

## 1.8 The *mbtest.imposters.behaviors.using* module

**class** `mbtest.imposters.behaviors.using.Using` (*method*, *selector*)

How to select values from the response.

**Parameters**

- **method** (*Method*) – The method used to select the value(s) from the request.
- **selector** (*str*) – The selector used to select the value(s) from the request.

**class** `Method`

An enumeration.

**REGEX** = 'regex'

**XPATH** = 'xpath'

**JSONPATH** = 'jsonpath'

**as\_structure** ()

Converted to a JSON serializable structure.

**Return type** *Any*

**Returns** Structure suitable for JSON serialisation.

**static from\_structure** (*structure*)

Converted from a JSON serializable structure.

**Parameters** **structure** (*Any*) – JSON structure to be converted.

**Return type** *Using*

**Returns** Converted object.

**class** `mbtest.imposters.behaviors.using.UsingRegex` (*selector*, *ignore\_case=False*, *multiline=False*)

Select values from the response using a regular expression.

**Parameters**

- **selector** (*str*) – The selector used to select the value(s) from the request.
- **ignore\_case** (*bool*) – Uses a case-insensitive regular expression
- **multiline** (*bool*) – Uses a multiline regular expression

**as\_structure** ()

Converted to a JSON serializable structure.

**Return type** *Any*

**Returns** Structure suitable for JSON serialisation.

**static from\_structure** (*structure*)

Converted from a JSON serializable structure.

**Parameters** **structure** (*Any*) – JSON structure to be converted.

**Return type** *UsingRegex*

**Returns** Converted object.

**class** `mbtest.imposters.behaviors.using.UsingXpath` (*selector*, *ns=None*)

Select values from the response using an xpath expression.

**Parameters**

- **selector** (*str*) – The selector used to select the value(s) from the request.
- **ns** (*Optional[Mapping[str, str]]*) – The ns object maps namespace aliases to URLs

**as\_structure** ()

Converted to a JSON serializable structure.

**Return type** *Any***Returns** Structure suitable for JSON serialisation.**static from\_structure** (*structure*)

Converted from a JSON serializable structure.

**Parameters** **structure** (*Any*) – JSON structure to be converted.**Return type** *UsingXpath***Returns** Converted object.**class** `mbtest.imposters.behaviors.using.UsingJsonpath` (*selector*)

Select values from the response using a jsonpath expression.

**Parameters** **selector** (*str*) – The selector used to select the value(s) from the request.**static from\_structure** (*structure*)

Converted from a JSON serializable structure.

**Parameters** **structure** – JSON structure to be converted.**Return type** *UsingJsonpath***Returns** Converted object.

## 1.9 The *mbtest.matchers* module

### 1.10 The *mbtest.imposters.base* module

**class** `mbtest.imposters.base.JsonSerializable`Object capable of being converted to a JSON serializable structure (using `as_structure()`) or from such a structure ((using `from_structure()`).**abstract as\_structure** ()

Converted to a JSON serializable structure.

**Return type** *Any***Returns** Structure suitable for JSON serialisation.**abstract static from\_structure** (*structure*)

Converted from a JSON serializable structure.

**Parameters** **structure** (*Any*) – JSON structure to be converted.**Return type** *JsonSerializable***Returns** Converted object.



## INDICES AND TABLES

- `genindex`
- `modindex`
- `search`



## INSTALLATION

Install from [Pypi](#) as usual, using `pip`, `tox`, or `setup.py`.

Also requires [Mountebank](#) to have been installed:

```
$ npm install mountebank@2.2 --production
```



**USAGE**

A basic example:

```
import requests
from hamcrest import assert_that
from brunns.matchers.response import is_response
from mbtest.matchers import had_request
from mbtest.imposters import Imposter, Predicate, Response, Stub

def test_request_to_mock_server(mock_server):
    # Set up mock server with required behavior
    imposter = Imposter(Stub(Predicate(path="/test"),
                             Response(body="sausages")))

    with mock_server(imposter) as server:
        # Make request to mock server - exercise code under test here
        response = requests.get("{} /test".format(imposter.url))

        assert_that("We got the expected response",
                     response, is_response().with_status_code(200).and_body("sausages
↪"))
        assert_that("The mock server recorded the request",
                     server, had_request().with_path("/test").and_method("GET"))
```

Needs a pytest fixture, most easily defined in `conftest.py`:

```
import pytest
from mbtest import server

@pytest.fixture(scope="session")
def mock_server(request):
    return server.mock_server(request)
```



## INDICES AND TABLES

- `genindex`
- `modindex`
- `search`





## PYTHON MODULE INDEX

### m

- `mbtest.imposters.base`, [12](#)
- `mbtest.imposters.behaviors.copy`, [9](#)
- `mbtest.imposters.behaviors.lookup`, [10](#)
- `mbtest.imposters.behaviors.using`, [11](#)
- `mbtest.imposters.imposters`, [3](#)
- `mbtest.imposters.predicates`, [4](#)
- `mbtest.imposters.responses`, [7](#)
- `mbtest.imposters.stubs`, [4](#)



## INDEX

### A

ALWAYS (*mbtest.imposters.responses.Proxy.Mode* attribute), 8

AndPredicate (*class in mbtest.imposters.predicates*), 6

as\_structure() (*mbtest.imposters.base.JsonSerializable* method), 12

as\_structure() (*mbtest.imposters.behaviors.copy.Copy* method), 9

as\_structure() (*mbtest.imposters.behaviors.lookup.Key* method), 10

as\_structure() (*mbtest.imposters.behaviors.lookup.Lookup* method), 10

as\_structure() (*mbtest.imposters.behaviors.using.Using* method), 11

as\_structure() (*mbtest.imposters.behaviors.using.Using* method), 11

as\_structure() (*mbtest.imposters.behaviors.using.UsingXpath* method), 12

as\_structure() (*mbtest.imposters.imposters.Imposter* method), 3

as\_structure() (*mbtest.imposters.predicates.AndPredicate* method), 6

as\_structure() (*mbtest.imposters.predicates.InjectionPredicate* method), 7

as\_structure() (*mbtest.imposters.predicates.OrPredicate* method), 6

as\_structure() (*mbtest.imposters.predicates.Predicate* method), 5

as\_structure() (*mbtest.imposters.predicates.TcpPredicate* method), 6

as\_structure() (*mbtest.imposters.responses.InjectionResponse* method), 9

as\_structure() (*mbtest.imposters.responses.Proxy* method), 8

as\_structure() (*mbtest.imposters.responses.Response* method), 8

as\_structure() (*mbtest.imposters.responses.TcpResponse* method), 8

as\_structure() (*mbtest.imposters.stubs.Stub* method), 4

### B

BasePredicate (*class in mbtest.imposters.predicates*), 4

BaseResponse (*class in mbtest.imposters.responses*), 7

BINARY (*mbtest.imposters.responses.Response.Mode* attribute), 8

body() (*mbtest.imposters.responses.Response* property), 8

CONTAINS (*mbtest.imposters.predicates.Predicate.Operator* attribute), 5

copy (*class in mbtest.imposters.behaviors.copy*), 9

DEEP\_EQUALS (*mbtest.imposters.predicates.Predicate.Operator* attribute), 5

DELETE (*mbtest.imposters.predicates.Predicate.Method* attribute), 5

ENDS\_WITH (*mbtest.imposters.predicates.Predicate.Operator* attribute), 5

EQUALS (*mbtest.imposters.predicates.Predicate.Operator* attribute), 5

EXISTS (*mbtest.imposters.predicates.Predicate.Operator* attribute), 5

from\_structure() (*mbtest.imposters.base.JsonSerializable* static method), 12

from\_structure() (*mbtest.imposters.behaviors.copy.Copy* static method), 9

from\_structure() (*mbtest.imposters.behaviors.lookup.Key* static method), 10

from\_structure() (*mbtest.imposters.behaviors.lookup.Lookup* static method), 10

from\_structure() (*mbtest.imposters.behaviors.using.Using* static method), 11

from\_structure() (*mbtest.imposters.behaviors.using.UsingJsonpath* static method), 12

`from_structure()` (*mbtest.imposters.behaviors.using.UsingRegex* (class in *mbtest.imposters.responses*), 9  
*static method*), 11  
`from_structure()` (*mbtest.imposters.behaviors.using.UsingXpath* (class in *mbtest.imposters.responses*), 9  
*static method*), 12  
`from_structure()` (*mbtest.imposters.imposters.Imposter* (class in *mbtest.imposters.imposters*), 3  
*static method*), 3  
`from_structure()` (*mbtest.imposters.predicates.AndPredicate* (class in *mbtest.imposters.predicates*), 6  
*static method*), 6  
`from_structure()` (*mbtest.imposters.predicates.BasePredicate* (class in *mbtest.imposters.predicates*), 4  
*static method*), 4  
`from_structure()` (*mbtest.imposters.predicates.InjectionPredicate* (class in *mbtest.imposters.predicates*), 7  
*static method*), 7  
`from_structure()` (*mbtest.imposters.predicates.OrPredicate* (class in *mbtest.imposters.predicates*), 6  
*static method*), 6  
`from_structure()` (*mbtest.imposters.predicates.Predicate* (class in *mbtest.imposters.predicates*), 4  
*static method*), 5  
`from_structure()` (*mbtest.imposters.predicates.TcpPredicate* (class in *mbtest.imposters.predicates*), 6  
*static method*), 6  
`from_structure()` (*mbtest.imposters.responses.BaseResponse* (class in *mbtest.imposters.responses*), 7  
*static method*), 7  
`from_structure()` (*mbtest.imposters.responses.InjectionResponse* (class in *mbtest.imposters.responses*), 9  
*static method*), 9  
`from_structure()` (*mbtest.imposters.responses.Proxy* (class in *mbtest.imposters.responses*), 8  
*static method*), 8  
`from_structure()` (*mbtest.imposters.responses.Response* (class in *mbtest.imposters.responses*), 8  
*static method*), 8  
`from_structure()` (*mbtest.imposters.responses.TcpResponse* (class in *mbtest.imposters.responses*), 8  
*static method*), 8  
`from_structure()` (*mbtest.imposters.stubs.Stub* (class in *mbtest.imposters.stubs*), 4  
*static method*), 4

## G

`GET` (*mbtest.imposters.predicates.Predicate.Method* attribute), 5

## H

`has_value()` (*mbtest.imposters.predicates.Predicate.Operator* (class in *mbtest.imposters.predicates*), 5  
*class method*), 5  
`HEAD` (*mbtest.imposters.predicates.Predicate.Method* attribute), 5  
`host()` (*mbtest.imposters.imposters.Imposter* property), 3  
`HTTP` (*mbtest.imposters.imposters.Imposter.Protocol* attribute), 3  
`HTTPS` (*mbtest.imposters.imposters.Imposter.Protocol* attribute), 3

## I

`Imposter` (class in *mbtest.imposters.imposters*), 3  
`Imposter.Protocol` (class in *mbtest.imposters.imposters*), 3  
`InjectionPredicate` (class in *mbtest.imposters.predicates*), 6

## O

`ONCE` (*mbtest.imposters.responses.Proxy.Mode* attribute), 8  
`OrPredicate` (class in *mbtest.imposters.predicates*), 6

## P

POST (*mbtest.imposters.predicates.Predicate.Method* attribute), 5

Predicate (class in *mbtest.imposters.predicates*), 4

Predicate.InvalidPredicateOperator, 5

Predicate.Method (class in *mbtest.imposters.predicates*), 5

Predicate.Operator (class in *mbtest.imposters.predicates*), 5

Proxy (class in *mbtest.imposters.responses*), 8

Proxy.Mode (class in *mbtest.imposters.responses*), 8

PUT (*mbtest.imposters.predicates.Predicate.Method* attribute), 5

## R

REGEX (*mbtest.imposters.behaviors.using.Using.Method* attribute), 11

Response (class in *mbtest.imposters.responses*), 7

Response.Mode (class in *mbtest.imposters.responses*), 7

## S

SMTP (*mbtest.imposters.imposters.Imposter.Protocol* attribute), 3

smtp\_imposter() (in module *mbtest.imposters.imposters*), 4

STARTS\_WITH (*mbtest.imposters.predicates.Predicate.Operator* attribute), 5

Stub (class in *mbtest.imposters.stubs*), 4

## T

TCP (*mbtest.imposters.imposters.Imposter.Protocol* attribute), 3

TcpPredicate (class in *mbtest.imposters.predicates*), 6

TcpResponse (class in *mbtest.imposters.responses*), 8

TEXT (*mbtest.imposters.responses.Response.Mode* attribute), 8

TRANSPARENT (*mbtest.imposters.responses.Proxy.Mode* attribute), 8

## U

url() (*mbtest.imposters.imposters.Imposter* property), 3

Using (class in *mbtest.imposters.behaviors.using*), 11

Using.Method (class in *mbtest.imposters.behaviors.using*), 11

UsingJsonpath (class in *mbtest.imposters.behaviors.using*), 12

UsingRegex (class in *mbtest.imposters.behaviors.using*), 11

UsingXPath (class in *mbtest.imposters.behaviors.using*), 11

## X

XPATH (*mbtest.imposters.behaviors.using.Using.Method* attribute), 11