sphinxcontrib.datatemplates Documentation

Release 0.0.0

Doug Hellmann

Sep 19, 2020

Contents

`sphinxcontrib.datatemplates` helps you use static data in machine readable format in your documentation by letting you define Jinja2 templates to turn JSON, YAML, XML, or CSV data into reStructuredText for Sphinx to render as part of its output.

- Repo: https://github.com/sphinxcontrib/sphinxcontrib.datatemplates
- Docs: http://sphinxcontribdatatemplates.readthedocs.io/

CHAPTER 1

Using datatemplate

The `datatemplate` directive is the interface between the data source and the rendering template.

1.1 Directives

.. datatemplate:csv:: `source-path`
Load file at `source-path` (relative to the documentation build directory) via `csv.reader()` or `csv.DictReader` depending on `header` and render using `template` given in directive body.

.. datatemplate:dbm:: source-path::
Load DB at `source-path` (relative to the documentation build directory) via `dbm.open()` and render using `template` given in directive body.

.. datatemplate:import-module:: `module-name`
Load module `module-name` (must be importable in `conf.py`) via `importlib.import_module()` and render using `template` given in directive body.

.. datatemplate:json:: `source-path`
Load file at `source-path` (relative to the documentation build directory) via `json.load()` and render using `template` given in directive body.

.. datatemplate:nodata::
Load `None` as `data` and render using `template` given in directive body.

.. datatemplate:xml:: `source-path`
Load file at `source-path` (relative to the documentation build directory) via `xml.etree.ElementTree.parse()` (actually using `defusedxml`) and render using `template` given in directive body.

.. datatemplate:yaml:: `source-path`
Load file at `source-path` (relative to the documentation build directory) via PyYAML (`yaml.safe_load()`) and render using `template` given in directive body.

1.2 Template Context

When a `datatemplate` directive is processed, the data from the `source` is passed to the template through its context so that the symbol `data` is available as a global variable.

Important: The data is loaded from the source and passed directly to the template. No pre-processing is done on the data, so the template needs to handle aspects like `None` values and fields that have values that may interfere with parsing reStructuredText.

The application configuration for a project will be passed to the template as the symbol `config`. This can be used, for example, to access HTML context via `config.html_context`. Refer to the *Inline Sample (JSON)* for an example.

The Sphinx build environment for a project will be passed to the template as the symbol `env`. This can be used to access all of the information that Sphinx has about the current build, including settings, and document names. Refer to the *No Data Sample* for an example.

1.3 Template Helpers

These helper functions are exposed using their short name (without the module prefix) in the template context.

`sphinxcontrib.datatemplates.helpers.`**`escape_rst`**(*s*)

Escape string for inclusion in RST documents.

See https://docutils.sourceforge.io/docs/ref/rst/restructuredtext.html#escaping-mechanism

Parameters **s** – String for escaping

`sphinxcontrib.datatemplates.helpers.`**`escape_rst_url`**(*s*)

Escape string for inclusion in URLs in RST documents.

See https://docutils.sourceforge.io/docs/ref/rst/restructuredtext.html#escaping-mechanism

Parameters **s** – String for escaping

`sphinxcontrib.datatemplates.helpers.`**`make_list_table`**(*headers*, *data*, *title=''*, *columns=None*)

Build a list-table directive.

Parameters

- **headers** – List of header values.
- **data** – Iterable of row data, yielding lists or tuples with rows.
- **title** – Optional text to show as the table title.
- **columns** – Optional widths for the columns.

`sphinxcontrib.datatemplates.helpers.`**`make_list_table_from_mappings`**(*headers*, *data*, *title*, *columns=None*)

Build a list-table directive.

Parameters

- **headers** – List of tuples containing header title and key value.
- **data** – Iterable of row data, yielding mappings with rows.

- **`title`** – Optional text to show as the table title.
- **`columns`** – Optional widths for the columns.

CHAPTER 2

No Data Sample

2.1 Loading the Template

```
.. datatemplate:nodata::

   Inline data:

   - {{ data }}

   Document titles from the Sphinx environment:

   {% for doc, title in env.titles.items() %}
   - ``{{ title }} ({{ doc }})``
   {% endfor %}
```

2.2 Rendered Output

Inline data:

- None

Document titles from the Sphinx environment:

- `<title>CLI Samples</title> (cli)`
- `<title>CSV Samples</title> (csv)`
- `<title>Release History</title> (history)`
- `<title>sphinxcontrib.datatemplates -- Render Your Data Readable</title> (index)`
- `<title>Inline Sample (JSON)</title> (inline)`
- `<title>JSON Samples</title> (json)`

- `<title>Legacy Samples</title>` (legacy)
- `<title>XML Samples</title>` (xml)
- `<title>YAML Samples</title>` (yaml)
- `<title>Using datatemplate</title>` (using)
- `<title>DBM Samples</title>` (dbm)
- `<title>Import-Module Samples</title>` (import-module)

CHAPTER 3

JSON Samples

3.1 Data File

```
{
    "key1": "value1",
    "key2": [
        "list item 1",
        "list item 2",
        "list item 3"
    ],
    "nested-list": [
        ["a", "b", "c"],
        ["A", "B", "C"]
    ],
    "mapping-series": [
        {"cola": "a", "colb": "b", "colc": "c"},
        {"cola": "A", "colb": "B", "colc": "C"}
    ]
}
```

3.2 Template File

```
.. -*- mode: rst -*-

Static Heading
--------------

Individual Item
~~~~~~~~~~~~~~~

{{ data['key1'] }}
```

(continues on next page)

(continued from previous page)

```
List of Items
~~~~~~~~~~~~~

{% for item in data['key2'] %}
- {{item}}
{% endfor %}

Nested List Table
~~~~~~~~~~~~~~~~~

Rendering a table from a list of nested sequences using hard-coded
headers.

{{ make_list_table(
    ['One', 'Two', 'Three'],
    data['nested-list'],
    title='Table from nested lists',
    ) }}

Mapping Series Table
~~~~~~~~~~~~~~~~~~~~

Rendering a table from a list of nested dictionaries using dynamic
headers.

{{ make_list_table_from_mappings(
    [('One', 'cola'), ('Two', 'colb'), ('Three', 'colc')],
    data['mapping-series'],
    title='Table from series of mappings',
    ) }}
```

3.3 Loading the Template

```
.. datatemplate:json:: sample.json
   :template: sample.tmpl
```

3.4 Rendered Output

3.4.1 Static Heading

Individual Item

value1

List of Items

- list item 1
- list item 2

- list item 3

Nested List Table

Rendering a table from a list of nested sequences using hard-coded headers.

Table 1: Table from nested lists

One	Two	Three
a	b	c
A	B	C

Mapping Series Table

Rendering a table from a list of nested dictionaries using dynamic headers.

Table 2: Table from series of mappings

One	Two	Three
a	b	c
A	B	C

CHAPTER 4

YAML Samples

4.1 Single Document

4.1.1 Data File

```
---
key1: value1
key2:
  - list item 1
  - list item 2
  - list item 3
nested-list:
  - ['a', 'b', 'c']
  - ['A', 'B', 'C']
mapping-series:
  - cola: a
    colb: b
    colc: c
  - cola: A
    colb: B
    colc: C
```

4.1.2 Template File

```
.. -*- mode: rst -*-

Static Heading
--------------

Individual Item
~~~~~~~~~~~~~~~
```

(continues on next page)

(continued from previous page)

```
{{ data['key1'] }}

List of Items
~~~~~~~~~~~~~

{% for item in data['key2'] %}
- {{item}}
{% endfor %}

Nested List Table
~~~~~~~~~~~~~~~~~

Rendering a table from a list of nested sequences using hard-coded
headers.

{{ make_list_table(
    ['One', 'Two', 'Three'],
    data['nested-list'],
    title='Table from nested lists',
    ) }}

Mapping Series Table
~~~~~~~~~~~~~~~~~~~~

Rendering a table from a list of nested dictionaries using dynamic
headers.

{{ make_list_table_from_mappings(
    [('One', 'cola'), ('Two', 'colb'), ('Three', 'colc')],
    data['mapping-series'],
    title='Table from series of mappings',
    ) }}
```

4.1.3 Loading the Template

```
.. datatemplate:yaml:: sample.yaml
   :template: sample.tmpl
```

4.1.4 Rendered Output

Static Heading

Individual Item

value1

List of Items

- list item 1
- list item 2

- list item 3

Nested List Table

Rendering a table from a list of nested sequences using hard-coded headers.

Table 1: Table from nested lists

One	Two	Three
a	b	c
A	B	C

Mapping Series Table

Rendering a table from a list of nested dictionaries using dynamic headers.

Table 2: Table from series of mappings

One	Two	Three
a	b	c
A	B	C

4.2 Multiple Documents

4.2.1 Data File

```
---
key1: value1

---
key: value
key1: different value
```

4.2.2 Template File

```
.. -*- mode: rst -*-

Static Heading
--------------

Individual Item
~~~~~~~~~~~~~~~

{{ data[0]|tojson }}

List of Items
~~~~~~~~~~~~~

{% for item in data %}
```

(continues on next page)

(continued from previous page)

```
- {{item|tojson}}

  - {{item.key}}
  - {{item.key1}}
{% endfor %}


Mapping Series Table
~~~~~~~~~~~~~~~~~~~~

Rendering a table from a list of nested dictionaries using dynamic
headers.

{{ make_list_table_from_mappings(
    [('Key', 'key'), ('Key One', 'key1')],
    data,
    title='Table from series of mappings',
    ) }}
```

4.2.3 Loading the Template

```
.. datatemplate:yaml:: sample-multiple.yaml
   :template: sample-multiple.tmpl
   :multiple-documents:
```

4.2.4 Rendered Output

Static Heading

Individual Item

{"key1": "value1"}

List of Items

- {"key1": "value1"}
  -
  - value1
- {"key": "value", "key1": "different value"}
  - value
  - different value

Mapping Series Table

Rendering a table from a list of nested dictionaries using dynamic headers.

Table 3: Table from series of mappings

Key	Key One
None	value1
value	different value

CHAPTER 5

XML Samples

5.1 Data File

```
<sample>
    <key1>value1</key1>
    <key2>
        <item>list item 1</item>
        <item>list item 2</item>
        <item special='yes'>list item 3</item>
    </key2>
    <mappingseries>
        <mapping>
            <cola special='yes'>a</cola>
            <colb>b</colb>
            <colc>c</colc>
        </mapping>
        <mapping>
            <cola>A</cola>
            <colb special='yes'>B</colb>
            <colc>C</colc>
        </mapping>
    </mappingseries>
</sample>
```

5.2 Template File

```
.. -*- mode: rst -*-

Static Heading
--------------
```

(continues on next page)

(continued from previous page)

```
Individual Item
~~~~~~~~~~~~~~~

{{ data.find('key1').text }}

List of Items
~~~~~~~~~~~~~

{% for item in data.find('key2') %}
- {{item.text}}
{% endfor %}

XPath for Items
~~~~~~~~~~~~~~~

See `XPath support <https://docs.python.org/3/library/xml.etree.elementtree.html
↪#xpath-support>`_

{% for item in data.findall(".//*[@special='yes']") %}
- {{item.text}}
{% endfor %}
```

5.3 Loading the Template

```
.. datatemplate:xml:: sample.xml
   :template: xml-sample.tmpl
```

5.4 Rendered Output

5.4.1 Static Heading

Individual Item

value1

List of Items

- list item 1
- list item 2
- list item 3

XPath for Items

See XPath support

- list item 3
- a

- B

CHAPTER 6

Import-Module Samples

6.1 Template File

```
.. -*- mode: rst -*-

Static Heading
--------------

List of Directory Entries
~~~~~~~~~~~~~~~~~~~~~~~~~

{% for item in data.scandir() %}
- {{item.name}}'s size is {{item.stat().st_size}} Bytes
{% endfor %}

File Path of the Null Device
~~~~~~~~~~~~~~~~~~~~~~~~~~~

``{{data.devnull}}``
```

6.2 Loading the Template

```
.. datatemplate:import-module:: os
   :template: import-module-sample.tmpl
```

6.3 Rendered Output

6.3.1 Static Heading

List of Directory Entries

- inline.rst's size is 1676 Bytes
- sampledbm.dat's size is 519 Bytes
- legacy.rst's size is 411 Bytes
- conf.py's size is 15710 Bytes
- _static's size is 4096 Bytes
- history.rst's size is 3198 Bytes
- sample.yaml's size is 212 Bytes
- yaml.rst's size is 954 Bytes
- _templates's size is 4096 Bytes
- index.rst's size is 791 Bytes
- sample.xml's size is 483 Bytes
- xml.rst's size is 416 Bytes
- import-module.rst's size is 419 Bytes
- make_dbm.py's size is 114 Bytes
- sample-multiple.yaml's size is 55 Bytes
- using.rst's size is 1968 Bytes
- sampledbm.dir's size is 29 Bytes
- _build's size is 4096 Bytes
- sample.csv's size is 36 Bytes
- json.rst's size is 410 Bytes
- nodata.rst's size is 581 Bytes
- sample.json's size is 319 Bytes
- csv.rst's size is 532 Bytes
- cli.rst's size is 1112 Bytes
- dbm.rst's size is 433 Bytes

File Path of the Null Device

```
/dev/null
```

CHAPTER 7

CSV Samples

7.1 Data File

```
a        b        c
Eins        Zwei        Drei
1        2        3
I        II        III
```

7.2 Template File

```
.. -*- mode: rst -*-

Static Heading
--------------

Individual Cell in Row
~~~~~~~~~~~~~~~~~~~~~~

{{ data[0].a }}

List of Cells in Row
~~~~~~~~~~~~~~~~~~~~

{% for item in data[0].items() %}
- {{item[0]}}: {{item[1]}}
{% endfor %}

Mapping Series Table
~~~~~~~~~~~~~~~~~~~~

Rendering a table from a list of nested dictionaries using dynamic
```

(continues on next page)

(continued from previous page)

```
headers.

{{ make_list_table_from_mappings(
    [('One', 'a'), ('Two', 'b'), ('Three', 'c')],
    data,
    title='Table from series of mappings',
    ) }}
```

7.3 Loading the Template

```
.. datatemplate:csv:: sample.csv
    :template: csv-sample.tmpl
    :headers:
    :dialect: excel-tab
```

7.4 Rendered Output

7.4.1 Static Heading

Individual Cell in Row

Eins

List of Cells in Row

- a: Eins
- b: Zwei
- c: Drei

Mapping Series Table

Rendering a table from a list of nested dictionaries using dynamic headers.

Table 1: Table from series of mappings

One	Two	Three
Eins	Zwei	Drei
1	2	3
I	II	III

CHAPTER 8

DBM Samples

8.1 Creating Data File

```
import dbm.dumb

with dbm.dumb.open("sampledbm", "c") as db:
    db[b"Hi"] = b"Hello"
    db[b"Bye"] = b"Goodbye"
```

8.2 Template File

```
.. -*- mode: rst -*-

Static Heading
--------------

Individual Item
~~~~~~~~~~~~~~~

- With decoding {{ data['Hi'].decode('ascii') }}
- Without decoding {{ data['Hi'] }}

List of Items
~~~~~~~~~~~~~

{% for item in data.items() %}
- {{item[0]}} -> {{item[1]}}
{% endfor %}
```

8.3 Loading the Template

```
.. datatemplate:dbm:: sampledbm
   :template: dbm-sample.tmpl
```

8.4 Rendered Output

8.4.1 Static Heading

Individual Item

- With decoding Hello
- Without decoding b'Hello'

List of Items

- b'Hi' -> b'Hello'
- b'Bye' -> b'Goodbye'

CHAPTER 9

Inline Sample (JSON)

This example demonstrates how to use an inline template, as well as accessing the *HTML context* available to all `datatemplate` directives.

9.1 Data File

```
{
    "key1": "value1",
    "key2": [
        "list item 1",
        "list item 2",
        "list item 3"
    ],
    "nested-list": [
        ["a", "b", "c"],
        ["A", "B", "C"]
    ],
    "mapping-series": [
        {"cola": "a", "colb": "b", "colc": "c"},
        {"cola": "A", "colb": "B", "colc": "C"}
    ]
}
```

9.2 HTML Context

```
# from conf.py
html_context = {
    'sample': 'Sample context value set in conf.py',
}
```

9.3 Template File

```
Individual Item
~~~~~~~~~~~~~~~

{{ data['key1'] }}

List of Items
~~~~~~~~~~~~~

{% for item in data['key2'] %}
- {{item}}
{% endfor %}

HTML Context
~~~~~~~~~~~~

{% for key, value in config.html_context.items() %}
- ``{{key}}`` = ``{{value}}``
{% endfor %}
```

9.4 Loading the Template

```
.. datatemplate:json::
   :source: sample.json

   Individual Item
   ~~~~~~~~~~~~~~~

   {{ data['key1'] }}

   List of Items
   ~~~~~~~~~~~~~

   {% for item in data['key2'] %}
   - {{item}}
   {% endfor %}

   HTML Context
   ~~~~~~~~~~~~

   {% for key, value in config.html_context.items() %}
   - ``{{key}}`` = ``{{value}}``
   {% endfor %}
```

9.5 Rendered Output

9.5.1 Individual Item

value1

9.5.2 List of Items

- list item 1
- list item 2
- list item 3

9.5.3 HTML Context

- `sample` = `Sample context value set in conf.py`
- `using_theme` = `False`
- `html_theme` = `sphinx_rtd_theme`
- `current_version` = `0.7.0`
- `version_slug` = `0.7.0`
- `MEDIA_URL` = `https://media.readthedocs.org/`
- `STATIC_URL` = `https://assets.readthedocs.org/static/`
- `PRODUCTION_DOMAIN` = `readthedocs.org`
- `versions` = `[('latest', '/en/latest/'), ('stable', '/en/stable/'), ('0.7.0', '/en/0.7.0/'), ('0.6.1', '/en/0.6.1/'), ('0.6.0', '/en/0.6.0/'), ('0.5.0', '/en/0.5.0/'), ('0.4.0', '/en/0.4.0/'), ('0.3.0', '/en/0.3.0/'), ('0.2.0', '/en/0.2.0/'), ('0.1.0', '/en/0.1.0/')]`
- `downloads` = `[]`
- `subprojects` = `[]`
- `slug` = `sphinxcontribdatatemplates`
- `name` = `sphinxcontrib.datatemplates`
- `rtd_language` = `en`
- `programming_language` = `py`
- `canonical_url` = `https://sphinxcontribdatatemplates.readthedocs.io/en/latest/`
- `analytics_code` = `None`
- `single_version` = `False`
- `conf_py_path` = `/doc/source/`
- `api_host` = `https://readthedocs.org`
- `github_user` = `sphinx-contrib`
- `proxied_api_host` = `/_`
- `github_repo` = `datatemplates`
- `github_version` = `0.7.0`
- `display_github` = `True`
- `bitbucket_user` = `None`
- `bitbucket_repo` = `None`

- `bitbucket_version` = `0.7.0`
- `display_bitbucket` = `False`
- `gitlab_user` = `None`
- `gitlab_repo` = `None`
- `gitlab_version` = `0.7.0`
- `display_gitlab` = `False`
- `READTHEDOCS` = `True`
- `new_theme` = `True`
- `source_suffix` = `.rst`
- `ad_free` = `False`
- `docsearch_disabled` = `False`
- `user_analytics_code` = ""
- `global_analytics_code` = `UA-17997319-1`
- `commit` = `13ae1c28`

CHAPTER 10

Legacy Samples

The `datatemplate` directive is should no longer be used. It is deprecated, and will be removed in the next release.

10.1 Data File

```
---
key1: value1
key2:
  - list item 1
  - list item 2
  - list item 3
nested-list:
  - ['a', 'b', 'c']
  - ['A', 'B', 'C']
mapping-series:
  - cola: a
    colb: b
    colc: c
  - cola: A
    colb: B
    colc: C
```

10.2 Template File

```
.. -*- mode: rst -*-

Static Heading
--------------

Individual Item
```

(continues on next page)

(continued from previous page)

```
~~~~~~~~~~~~~~~

{{ data['key1'] }}

List of Items
~~~~~~~~~~~~~

{% for item in data['key2'] %}
- {{item}}
{% endfor %}

Nested List Table
~~~~~~~~~~~~~~~~~

Rendering a table from a list of nested sequences using hard-coded
headers.

{{ make_list_table(
    ['One', 'Two', 'Three'],
    data['nested-list'],
    title='Table from nested lists',
    ) }}

Mapping Series Table
~~~~~~~~~~~~~~~~~~~~

Rendering a table from a list of nested dictionaries using dynamic
headers.

{{ make_list_table_from_mappings(
    [('One', 'cola'), ('Two', 'colb'), ('Three', 'colc')],
    data['mapping-series'],
    title='Table from series of mappings',
    ) }}
```

10.3 Rendered Output

10.3.1 Static Heading

Individual Item

value1

List of Items

- list item 1
- list item 2
- list item 3

Nested List Table

Rendering a table from a list of nested sequences using hard-coded headers.

Table 1: Table from nested lists

One	Two	Three
a	b	c
A	B	C

Mapping Series Table

Rendering a table from a list of nested dictionaries using dynamic headers.

Table 2: Table from series of mappings

One	Two	Three
a	b	c
A	B	C

CHAPTER 11

CLI Samples

11.1 Help

```
usage: datatemplate [-h] [--config-file CONFIG_FILE] {render,dump} ...

optional arguments:
  -h, --help            show this help message and exit
  --config-file CONFIG_FILE
                        the path to conf.py

commands:
  valid commands

  {render,dump}
    render              render a template to stdout
    dump                dump the data to stdout without a template
```

11.2 Data File

```
---
key1: value1

---
key: value
key1: different value
```

11.3 Template File

```
.. -*- mode: rst -*-

Static Heading
--------------

Individual Item
~~~~~~~~~~~~~~~

{{ data[0]|tojson }}

List of Items
~~~~~~~~~~~~~

{% for item in data %}
- {{item|tojson}}

  - {{item.key}}
  - {{item.key1}}
{% endfor %}


Mapping Series Table
~~~~~~~~~~~~~~~~~~~~

Rendering a table from a list of nested dictionaries using dynamic
headers.

{{ make_list_table_from_mappings(
    [('Key', 'key'), ('Key One', 'key1')],
    data,
    title='Table from series of mappings',
    ) }}
```

11.4 Rendering a Template

```
$ datatemplate render -o multiple-documents \
  doc/source/_templates/sample-multiple.tmpl \
  doc/source/sample-multiple.yaml
```

```
Traceback (most recent call last):
  File "/home/docs/checkouts/readthedocs.org/user_builds/sphinxcontribdatatemplates/
↪envs/0.7.0/bin/datatemplate", line 10, in <module>
    sys.exit(main())
  File "/home/docs/checkouts/readthedocs.org/user_builds/sphinxcontribdatatemplates/
↪envs/0.7.0/lib/python3.7/site-packages/sphinxcontrib/datatemplates/cli.py", line 76,
↪ in main
    return args.func(args, conf)
  File "/home/docs/checkouts/readthedocs.org/user_builds/sphinxcontribdatatemplates/
↪envs/0.7.0/lib/python3.7/site-packages/sphinxcontrib/datatemplates/cli.py", line␣
↪107, in render
    with io.open(args.template, 'r', encoding='utf-8-sig') as f:
FileNotFoundError: [Errno 2] No such file or directory: 'doc/source/_templates/sample-
↪multiple.tmpl'
```

(continues on next page)

(continued from previous page)

11.5 Experimenting by Dumping Data

11.5.1 CSV Data With Headers

```
$ datatemplate dump -o dialect:excel-tab \
  -o headers \
  doc/source/sample.csv
```

```
Traceback (most recent call last):
  File "/home/docs/checkouts/readthedocs.org/user_builds/sphinxcontribdatatemplates/
↪envs/0.7.0/bin/datatemplate", line 10, in <module>
    sys.exit(main())
  File "/home/docs/checkouts/readthedocs.org/user_builds/sphinxcontribdatatemplates/
↪envs/0.7.0/lib/python3.7/site-packages/sphinxcontrib/datatemplates/cli.py", line 76,
↪ in main
    return args.func(args, conf)
  File "/home/docs/checkouts/readthedocs.org/user_builds/sphinxcontribdatatemplates/
↪envs/0.7.0/lib/python3.7/site-packages/sphinxcontrib/datatemplates/cli.py", line␣
↪134, in dump
    with load(**conf) as data:
  File "/home/docs/.pyenv/versions/3.7.3/lib/python3.7/contextlib.py", line 112, in __
↪enter__
    return next(self.gen)
  File "/home/docs/checkouts/readthedocs.org/user_builds/sphinxcontribdatatemplates/
↪envs/0.7.0/lib/python3.7/site-packages/sphinxcontrib/datatemplates/loaders.py",␣
↪line 93, in load_csv
    with open(absolute_resolved_path, 'r', newline='', encoding=encoding) as f:
FileNotFoundError: [Errno 2] No such file or directory: '/home/docs/checkouts/
↪readthedocs.org/user_builds/sphinxcontribdatatemplates/checkouts/0.7.0/doc/source/
↪doc/source/sample.csv'
```

11.5.2 CSV Data Without Headers

```
$ datatemplate dump -o dialect:excel-tab \
  doc/source/sample.csv
```

```
Traceback (most recent call last):
  File "/home/docs/checkouts/readthedocs.org/user_builds/sphinxcontribdatatemplates/
↪envs/0.7.0/bin/datatemplate", line 10, in <module>
    sys.exit(main())
  File "/home/docs/checkouts/readthedocs.org/user_builds/sphinxcontribdatatemplates/
↪envs/0.7.0/lib/python3.7/site-packages/sphinxcontrib/datatemplates/cli.py", line 76,
↪ in main
    return args.func(args, conf)
  File "/home/docs/checkouts/readthedocs.org/user_builds/sphinxcontribdatatemplates/
↪envs/0.7.0/lib/python3.7/site-packages/sphinxcontrib/datatemplates/cli.py", line␣
↪134, in dump
    with load(**conf) as data:
  File "/home/docs/.pyenv/versions/3.7.3/lib/python3.7/contextlib.py", line 112, in __
↪enter__
```

(continues on next page)

(continued from previous page)

```
    return next(self.gen)
  File "/home/docs/checkouts/readthedocs.org/user_builds/sphinxcontribdatatemplates/
↪envs/0.7.0/lib/python3.7/site-packages/sphinxcontrib/datatemplates/loaders.py",␣
↪line 93, in load_csv
    with open(absolute_resolved_path, 'r', newline='', encoding=encoding) as f:
FileNotFoundError: [Errno 2] No such file or directory: '/home/docs/checkouts/
↪readthedocs.org/user_builds/sphinxcontribdatatemplates/checkouts/0.7.0/doc/source/
↪doc/source/sample.csv'
```

CHAPTER 12

Release History

12.1 0.7.0

- add sphinx builder environment to template context
- cli: add a 'dump' subcommand
- cli: create 'render' subcommand
- treat flag options as special
- add "datatemplate" console script
- Update domain.py
- pass the entire application config to the template
- add sample for nothing
- add 3.7 and 3.8 to list of supported python versions
- add html_context to template context
- add data loaders registry
- make directive option conversion results more intuitive
- note source-file as dependency
- pass all options to template, allow unknown options

12.2 0.6.1

- pbr versioning

12.3 0.6.0

- pbr is required not just for setup. See #43
- better option validators
- Use directive head (arguments) for source path
- Allow specifying template in directive body

12.4 0.5.0

- Fix linting errors
- Add domain for Python Modules
- Move import to the top of the module
- Use default template manager when the builder does not have one
- Necessary method for parallel builds
- list instead of tuple
- Add option to load multiple documents from yaml
- Restore Python3.6 compat
- Add support for DBM formats
- Set __version__
- ensure each directive page shows how to use the template

12.5 0.4.0

- clarify/expand warning about legacy directive
- add a doc page to show that the legacy form of the directive still works
- turn off -W option in sphinx build
- Wrap directives in minimal domain
- stupid copy-paste merging
- linting error
- DataTemplate from 0.3.0 as DataTemplateLegacy for compat
- method for path resolution
- Add directive "datatemplate" for backwards compat
- Update yaml.rst
- Split datatemplate directive by file type
- Ignore venv, vscode settings
- add option for encoding

12.6 0.3.0

- add examples to readme
- add twine check to linter
- fix packaging metadata
- add a table to show the template input types
- clean up bad comment in travis config
- tell travis to use py3.6 and not ignore failures
- remove extra doc format builds
- remove superfluous travis command for go tools
- tell git to ignore build artifacts
- set up travis configuration
- address flake8 errors
- move dependency settings from tox to setup.cfg
- Add dialect support, better dotumentation
- Use yaml.safe_load
- Add a little bit of documentation for XML
- Use defusedxml
- Add XML support
- Add CSV support

12.7 0.2.0

- Use sphinx.util.logging for logging calls
- Fix noqa flagging of import exception
- optionally exec the conf.py file and pass settings to the template
- make test-template support python 2 and 3
- update github URL in documentation
- update the source repo URL in readme
- update to python 3.5
- add license file
- Add links to repo and docs from README and docs frontpage
- add a command line tool to make testing templates easier

12.8 0.1.0

- more protection against differences in builders
- avoid errors for builders without template lookup
- add usage instructions
- add table helpers and samples
- don't force a theme setting
- remove debug print
- add JSON support
- add YAML support
- fix flake8 warnings for sphinx conf.py
- add ourself to the doc extensions we use
- basic project setup

CHAPTER 13

Indices and tables

- genindex
- modindex
- search

Python Module Index

S

Index

D

E

M

S