

Python Docstrings

Instructor

Battista Biggio

M.Sc. in Computer Engineering, Cybersecurity and Artificial Intelligence

University of Cagliari, Italy

Docstrings

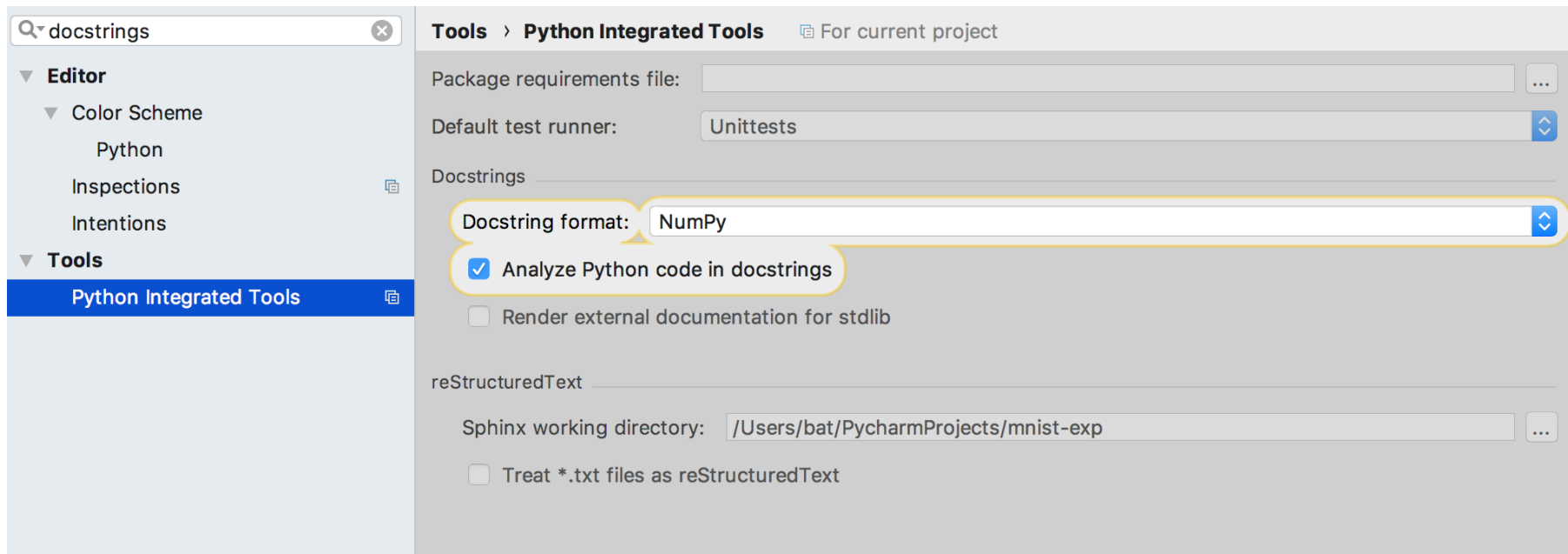
- Strings used for code documentation
- Enable automatic generation of docs in different formats (HTML, etc.)
- Different formats for Python
 - numpydoc
 - ...

Required Packages

- In our practical sessions, we will use
 - **sphinx**: to automatically generate the documentation from code docstrings
 - **numpydoc**: as our standard to format our docstrings
 - <https://numpydoc.readthedocs.io/en/latest/>
 - **sphinx_rtd_theme**: HTML popular theme for docs

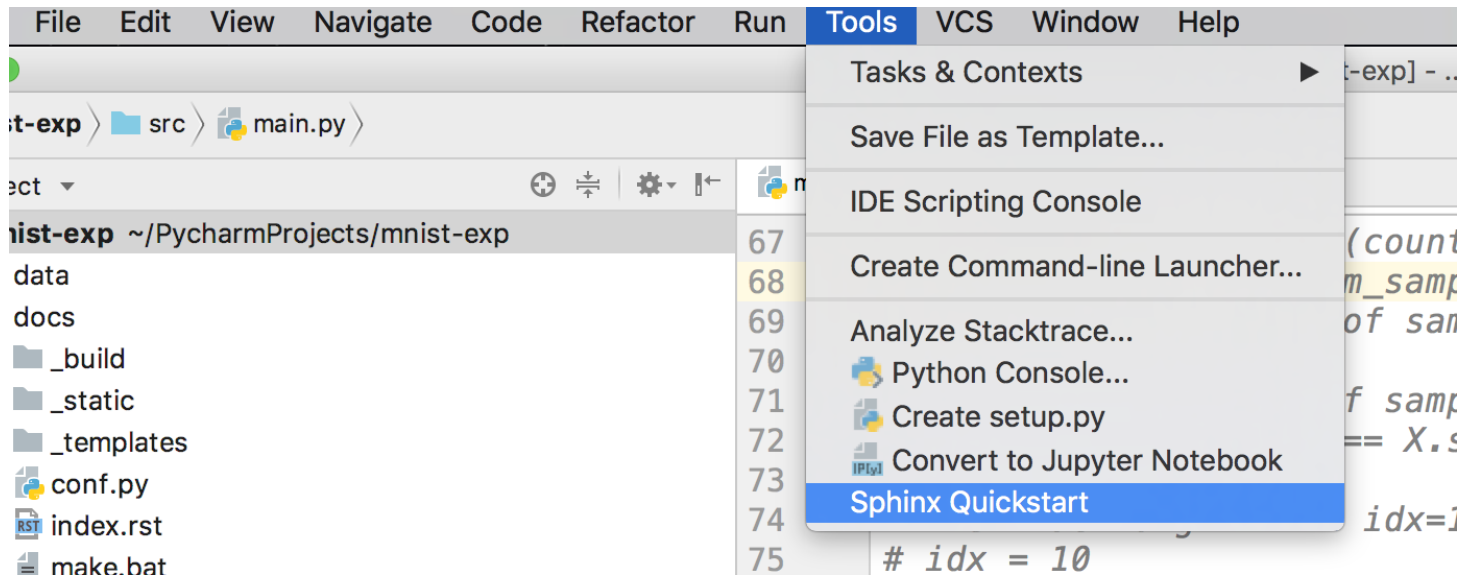
Configuring PyCharm

- Step 1: Set numpydoc as the default docstring format



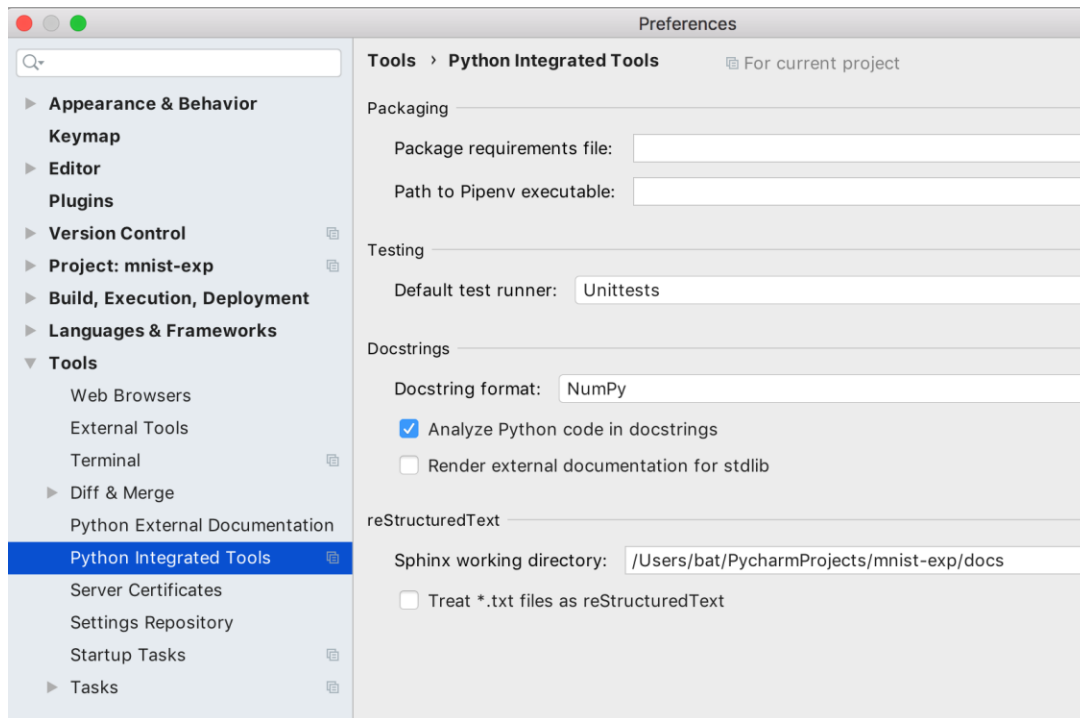
Configuring PyCharm

- Step 2: Set-up Sphinx configuration



Sphinx Quickstart

- Automatically configures sphinx environment
 - Set sphinx working directory «docs»
- Suggested preferences:
 - Do not separate source and build



Sphinx Quickstart

- The summary of your configuration is stored in **conf.py**

- Set path to your src folder

```
# -- Path setup -----  
import os  
import sys  
sys.path.insert(0, os.path.abspath('../src/'))
```

- To use the sphinx rtd theme, set:

```
html_theme = 'sphinx_rtd_theme'
```

- Add autodoc extension:

```
extensions = ['sphinx.ext.autodoc']
```

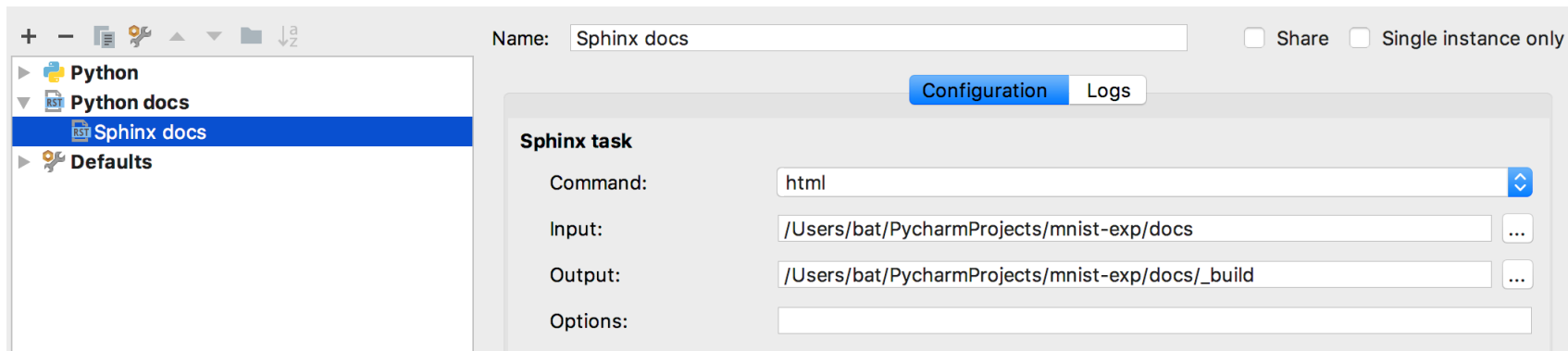
- **Index.rst** contains the name of the modules to be included

- To add ours, let's just add the following commands into that file

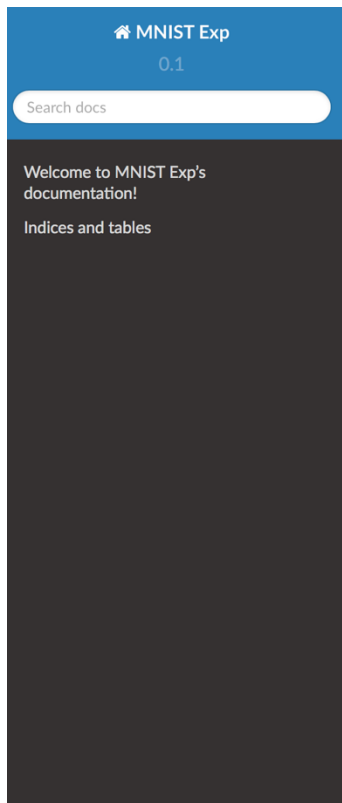
```
.. automodule:: your_file_or_package_name  
   :members:  
   :undoc-members:
```

Run Configuration for Sphinx docs in PyCharm

- Run → Edit Configurations
- Create a configuration 'Sphinx docs' from the defaults
- Set Input and Output folder
 - Output is just input_folder/_build



Finally...



[Docs](#) » Welcome to MNIST Exp's documentation!

[View page source](#)

Welcome to MNIST Exp's documentation!

`main.count_digits(y)` [\[source\]](#)

Count the number of elements in each class

Parameters: `y: ndarray`
the labels of each sample.

Returns: `y: ndarray`
the number of elements in each class.

`main.load_data(filename)` [\[source\]](#)

Load data from a csv file

Parameters: `filename: string`
Filename to be loaded.

Returns: `X: ndarray`
the data matrix.
`y: ndarray`
the labels of each sample.

See also...

- Blog post by Brendan Hasz
 - <https://brendanhasz.github.io/2019/01/05/sphinx.html>