

Python

базовый тренинг

Ружин Алексей
ruzin@me.com

Документирование

- `pip install Sphinx`
- `sphinx-quickstart` (из консоли)
- `docs`
- `conf.py`
- `index.rst`

Иерархия

```
.. toctree::  
    :maxdepth: 2  
    :caption: Contents:
```

intro

api

Вставка docstrings

- `.. automodule:: name`
- `.. autoclass:: Name`
 - `:members:`
 - `:special-members:`
 - `:undoc-members:`

docstrings

////

Gets something good.

Args:

data_view (str): URL pattern name

Raises:

Http404: If the widget can not be found.

Returns:

ModelSelect2Mixin: Widget from cache.

////

docstrings: 'code snippet'

////

Example usage::

```
class MyModelForm(forms.ModelForm):  
    class Meta:  
        model = MyModel  
        fields = ('my_field', )  
        widgets = {  
            'my_field': Select2Widget  
        }
```

////

Перекрестные ссылки

- :class:`A`
- :mod:`lesson_10`
- :meth:`lesson_10.A.method_name`

Логирование

- `import logging`
- `logging.basicConfig(
 level=logging.DEBUG,
 format=« »
 filename=« »
)`

Логирование

- `logging.debug('msg')`
- `.info()`, `.warning()`, `.error()`, `.critical()`

Логирование

- `logger = logging.getLogger(__name__)`
- `handler`
 - `logging.FileHandler(),`
 - `logging.StreamHandler(),`
 - `...`
- `formatter = logging.Formatter(« »)`
- `logger.addHandler(...)`
- `filters`

Отладка

- `import pdb`
- `pdb.set_trace()`

Профилирование

- `python -m cProfile [-o output_file] klass.py`
- `python -m pstats output_file`
`sort cumulative`
`stats 10`

Профилирование

- ```
import cProfile, pstats, StringIO
pr = cProfile.Profile()
pr.enable()
... do something ...
pr.disable()
s = StringIO.StringIO()
sortby = 'cumulative'
ps = pstats.Stats(pr, stream=s).sort_stats(sortby)
ps.print_stats()
print s.getvalue()
```