
tricks

laurens

Sep 24, 2020

CONTENTS:

1	python-for-java-devs	1
1.1	alina_test module	1
1.2	anna package	1
1.3	anna_hello module	1
1.4	documentation package	1
1.5	frederic package	2
1.6	frederic_hello module	3
1.7	joachim package	3
1.8	koen package	3
1.9	koen_hello module	3
1.10	laurens package	3
1.11	laurens_hello module	4
1.12	sara package	4
1.13	tom_hello module	4
2	Indices and tables	5
	Python Module Index	7
	Index	9

PYTHON-FOR-JAVA-DEVS

1.1 alina_test module

1.2 anna package

1.2.1 Submodules

1.2.2 anna.hello module

```
anna.hello.say_hello()
```

1.2.3 anna.hello_wrapper module

1.2.4 Module contents

1.3 anna_hello module

1.4 documentation package

1.4.1 Submodules

1.4.2 documentation.docstring_formats module

Automatically generated docstring, with different docstring formats.

For Pycharm: File -> Settings -> Tools -> Python integrated tools -> Docstrings -> Docstring format

```
documentation.docstring_formats.epytext (word: str, number: int) → list
```

This is for epytext

@param word: a word @param number: a number @return:

```
documentation.docstring_formats.google (word: str, number: int) → list
```

This is for google

Args: word: a word number: a number

Returns:

`documentation.docstring_formats.numpy(word: str, number: int) → list`

This is for numpy

word a word

number a number

`documentation.docstring_formats.plain(word: str, number: int) → list`

This is for plain

`documentation.docstring_formats.restrucleredtext(word: str, number: int) → list`

This is for restrucleredtext

Parameters

- **word** – a word
- **number** – a number

Returns

1.4.3 Module contents

1.5 frederic package

1.5.1 Subpackages

frederic.OOP package

Submodules

frederic.OOP.ngram module

Module contents

1.5.2 Submodules

1.5.3 frederic.hello_class module

class `frederic.hello_class.Hello`

Bases: `object`

static `create_user(name)`

say`()`

static `say_static()`

1.5.4 frederic.hello_world module

```
frederic.hello_world.say(n)
```

1.5.5 frederic.hello_wrapper module

1.5.6 Module contents

1.6 frederic_hello module

1.7 joachim package

1.7.1 Submodules

1.7.2 joachim.hello module

```
joachim.hello.say_hello()
```

1.7.3 joachim.hello_wrapper module

1.7.4 Module contents

1.8 koen package

1.8.1 Submodules

1.8.2 koen.hello module

```
koen.hello.say_hello(name)
```

1.8.3 koen.hello_wrapper module

1.8.4 Module contents

1.9 koen_hello module

1.10 laurens package

1.10.1 Submodules

1.10.2 laurens.hello module

```
class laurens.hello.Hello(creator: str, age: int = 0)  
    Bases: object
```

A simple class without any use

I haven't thought what this can create, but let's give the creator and it's age

Parameters

- **creator** – Please enter your name
- **age** – We advise to use non-negative ages

get_older (*delta_year: int = 1*)

We'll set the age one year older

Parameters **delta_year** – amount of years to increase

Returns the new age counter

`laurens.hello.hello_world(name: str)`

Greets you by your name!

Parameters **name** – Your lovely name

Returns my greetings

1.10.3 Module contents

1.11 laurens_hello module

1.12 sara package

1.12.1 Submodules

1.12.2 sara.hello module

`sara.hello.say_hello()`

1.12.3 sara.hello_wrapper module

1.12.4 Module contents

1.13 tom_hello module

INDICES AND TABLES

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

PYTHON MODULE INDEX

a

anna, 1
anna.hello, 1
anna_hello, 1

d

documentation, 2
documentation.docstring_formats, 1

f

frederic, 3
frederic.hello_class, 2
frederic.hello_world, 3
frederic.OOP, 2
frederic_hello, 3

j

joachim, 3
joachim.hello, 3

k

koen, 3
koen.hello, 3
koen_hello, 3

l

laurens, 4
laurens.hello, 3
laurens_hello, 4

s

sara, 4
sara.hello, 4

t

tom_hello, 4

INDEX

A

anna
 module, 1
anna.hello
 module, 1
anna_hello
 module, 1

C

create_user() (*frederic.hello_class.Hello static method*), 2

D

documentation
 module, 2
documentation.docstring_formats
 module, 1

E

epytext() (*in module documentation.docstring_formats*), 1

F

frederic
 module, 3
frederic.hello_class
 module, 2
frederic.hello_world
 module, 3
frederic.OOP
 module, 2
frederic_hello
 module, 3

G

get_older() (*laurens.hello.Hello method*), 4
google() (*in module documentation.docstring_formats*), 1

H

Hello (*class in frederic.hello_class*), 2

Hello (*class in laurens.hello*), 3
hello_world() (*in module laurens.hello*), 4

J

joachim
 module, 3
joachim.hello
 module, 3

K

koen
 module, 3
koen.hello
 module, 3
koen_hello
 module, 3

L

laurens
 module, 4
laurens.hello
 module, 3
laurens_hello
 module, 4

M

module
 anna, 1
 anna.hello, 1
 anna_hello, 1
 documentation, 2
 documentation.docstring_formats, 1
 frederic, 3
 frederic.hello_class, 2
 frederic.hello_world, 3
 frederic.OOP, 2
 frederic_hello, 3
 joachim, 3
 joachim.hello, 3
 koen, 3
 koen.hello, 3
 koen_hello, 3

```
laurens, 4
laurens.hello, 3
laurens_hello, 4
sara, 4
sara.hello, 4
tom_hello, 4
```

N

```
numpy() (in module documentation.docstring_formats),  
1
```

P

```
plain() (in module documentation.docstring_formats),  
2
```

R

```
restructuredtext() (in module documenta-  
tion.docstring_formats), 2
```

S

```
sara  
    module, 4  
sara.hello  
    module, 4  
say() (frederic.hello_class.Hello method), 2  
say() (in module frederic.hello_world), 3  
say_hello() (in module anna.hello), 1  
say_hello() (in module joachim.hello), 3  
say_hello() (in module koen.hello), 3  
say_hello() (in module sara.hello), 4  
say_static() (frederic.hello_class.Hello static  
method), 2
```

T

```
tom_hello  
    module, 4
```