
aiolambda

Release 0.0.3

Nov 05, 2019

Contents:

1	Bind	1
1.1	Description	1
1.2	Example	1
1.3	Curry	1
1.4	Decorator	2
2	Compose	3
2.1	Description	3
2.2	Example	3
3	aiofunctools	5
4	Indices and tables	7

1.1 Description

Function used to pass-through execution if input is an Exception.

In addition, it currrify function.

1.2 Example

```
def x5(x: int) -> int:
    return x * 5

maybe_x5 = bind(x5)
assert maybe_x5(1) == 5

error = maybe_x5(Exception())
```

1.3 Curry

Curried functions admit less inputs than mandatory ones and return a function with that inputs fixed.

In [this great book](#) is explained better.

```
def _sum(x: int, y: int) -> int:
    return x + y

maybe_sum = bind(_sum)
maybe_sum1 = maybe_sum(1)
assert maybe_sum1(4) == 5
```

(continues on next page)

(continued from previous page)

```
error = maybe_sum1(Exception())
```

1.4 Decorator

```
@bind
def x5(x: int) -> int:
    return x * 5
```

2.1 Description

Function used to compose other functions.

Allow async functions.

First input function is first executed.

2.2 Example

Compose:

```
from aiofunctools import compose

async def test(_input):

    return await compose(
        first,
        awaitable_second,
        third)(_input)
```

It is same code than:

```
from aiofunctools import compose

async def test(_input):
    return third(
        await awaitable_second(
            first(_input)))
```


CHAPTER 3

aiofunctools

Library to help in Python functional programming. It's asyncio compatible.

Basic idea behind it is [Railway Oriented Programing](#).

This allows us to:

- simplify our code.
- improve error management.
- be cool! be functional!

Examples:

Old code example:

```
async def create_user_handler(request) -> Response:
    try:
        user = check_valid_user(request)
        create_user(user)
    except InvalidBody:
        return_422('Invalid body')
    except UserAlreadyExists:
        return_409('User already exists')
    return_201(user)
```

ROP example:

```
async def create_user_handler(request) -> Response:
    return await compose(
        check_valid_user,
        create_user,
        return_201
    )(request)
```


CHAPTER 4

Indices and tables

- `genindex`
- `search`