AioPubsub Documentation

Bakhtiyor Ruziev

Feb 23, 2023

Contents

CHAPTER 1

API

class `aio_pubsub.interfaces.`**`PubSub`**

Main interface for implementation

publish(*channel: str*, *message: Any*)

Publish a message to channel

subscribe(*channel: str*)

Subscribe to one or many channels and return a Subscriber object

class `aio_pubsub.interfaces.`**`Subscriber`**

Subscriber interface

A generic interface wrapping multiple backends to provide a consistent pubsub API.

CHAPTER 2

Usage

To use it, you need to implement your pubsub implementation from interfaces or use backends from `aio_pubsub.backends` package:

```
from aio_pubsub.backends.memory import MemoryPubSub
pubsub = MemoryPubSub()
# Create subscriber
subscriber = await pubsub.subscribe("a_chan")

# Push message
await pubsub.publish("a_chan", "hello world!")
await pubsub.publish("a_chan", "hello universe!")

# And listening channel
try:
    async for message in subscriber:
        print(message, flush=True)
except KeyboardInterrupt:
    print("Finish listening")
```

CHAPTER 3

Supported backends

`Disclaimer`: I would not advise you to use this backend, because it is shown only for testing purposes. Better develop your own implementation.

- memory
- redis
- postgresql

Python Module Index

a

Index

A

P

S