

Concurrent

В отличие от `asynсrg`, в `motor` объектами библиотеки можно пользоваться из разных потоков.

```
import asyncio
import os

from bson import ObjectId
from motor.motor_asyncio import AsyncIOMotorClient

def get_client():
    return AsyncIOMotorClient(os.getenv("MONGO_URL"))

def get_db():
    return get_client().metaclass

def get_user_collection():
    return get_db().user

async def insert_user(coll) -> ObjectId:
    document = {'first_name': 'Alexander', 'last_name': 'Opryshko'}
    result = await coll.insert_one(document)
    return result.inserted_id

async def run():
    coll = get_user_collection()
    await asyncio.gather(insert_user(coll), insert_user(coll))

if __name__ == "__main__":
    asyncio.run(run())
```

Запустить пример можно в [mercury](#):

```
python3 connectors/mongo/parallel.py
```