

Report of TP Redis

This TP is for creating a "Publish-Subscribe News System", which permits the server publishes item and the client subscribes item.

My code is composed by 2 files, "pub.py" is to realise the function of the publisher and "sub.py" is to create the Redis client.

For the file "pub.py":

1. The publisher connects the server of Redis, here I use the connection pool which is used to manage all connections to a redis server.

```
pool = redis.ConnectionPool(host='127.0.0.1', port=6379,
decode_responses=True)

r = redis.Redis(connection_pool=pool)
```

2. Then the publisher puts the information of new into the database. For each news, the key is "id" and the value is "text" and "tags". We use Redis Hash to save the data with several values.

```
r.hmset(id1,{"text" : text1, "tags" : tag1}) r.hmset(id2,{"text" : text2,
"tags" : tag2}) r.hmset(id3,{"text" : text3, "tags" : tag3})
```

3. Supposed the news will expire in 20 seconds, I use the function "expire".

```
r.expire(id1,20)
```

4. After that, the publisher publishes these 3 news one by one. For each news, it publishes a channel for each indexed word containing the "id" of the news. For example, the new of "Sport" channel is published by (each one of "tags", "id").

```
for t1 in tag1: r.publish(t1, id1) print("Publish the news of id = " +
id1)
```

The client who has subscribed this channel will get the news "id" and he will receive the full text of the news.

For the file "sub.py":

1. The first part is like that in "pub.py", the client connects the server of Redis.

```
pool = redis.ConnectionPool(host='127.0.0.1', port=6379,
decode_responses=True) r = redis.Redis(connection_pool=pool)
```

2. Then I have created 2 objects of client: "client1" and "client2".

```
client1 = r.psubsub() client2 = r.psubsub()
```

3. The clients begin to subscribe. The parameter is the name of channel.

```
client1.subscribe(tag1)

client1.unsubscribe(tag2)
```

I have also used the function of "unsubscribe" to test if the client can cancel a subscribe.

4. After that there is a while loop which makes the client to get the message of information of news published by the publisher. When the client receives a message, he will get the "id" and will receive the full text of the news searched by its "id".

```
if message1 and message1['type'] == 'message': news1 =  
r.hget(message1['data'], "text") print ("Client1 receives a news about «"  
+message1['channel']+ "»: \n" + news1)
```

Readme

1. Configure the environnement of Redis.
2. Run Redis server.

```
./redis-server
```

3. Run "sub.py".

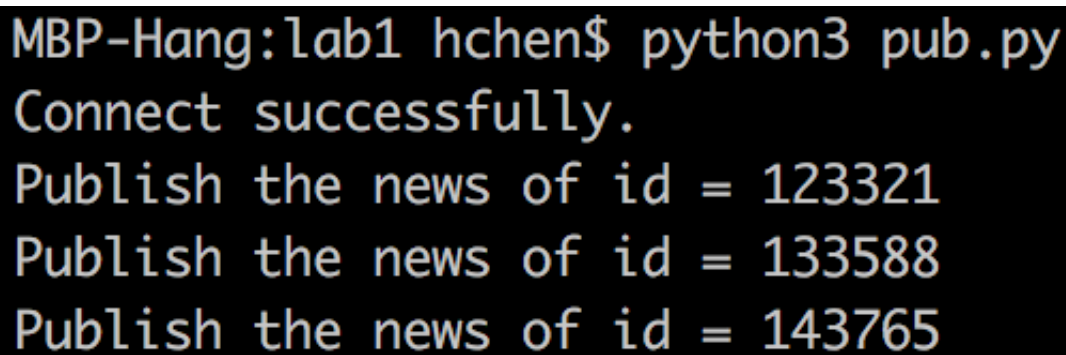
```
python3 sub.py
```

4. Run "pub.py".

```
python3 Pub.py
```

Result

The part of publisher:



```
MBP-Hang:lab1 hchen$ python3 pub.py  
Connect successfully.  
Publish the news of id = 123321  
Publish the news of id = 133588  
Publish the news of id = 143765
```

The part of client:

```
MBP-Hang:lab1 hchen$ python3 sub.py
Client1 connect the channel successfully.
Client2 connect the channel successfully.
Client1 subscribes the news about 《Sport》
Client1 subscribes the news about 《Politic》
Client2 subscribes the news about 《Sport》
Client2 subscribes the news about 《Business》
Client1 unsubscribes the news about 《Politic》
Client2 receives a news about 《Sport》 :
LeBron James Crestfallen After Learning L.A. Doesn't Have Any Rock And Roll Museums.
Client1 receives a news about 《Sport》 :
LeBron James Crestfallen After Learning L.A. Doesn't Have Any Rock And Roll Museums.
Client2 receives a news about 《Business》 :
Ride-hailing firm Uber is paying $148m (£113m) to settle legal action over a cyber-attack that exposed data from 57 million customers and drivers.
```