

Ring

2017 컨트리뷰톤

2017년 11월 21일 정윤원 유혜수

캐시

캐시

위키백과, 우리 모두의 백과사전.



다른 뜻에 대해서는 **캐시 (동음이의)** 문서를 참조하십시오.

캐시(cache, 문화어: 캐쉬, 고속완충기, 고속완충기억기)는 **컴퓨터 과학**에서 데이터나 값을 미리 복사해 놓는 임시 장소를 가리킨다. 캐시는 캐시의 접근 시간에 비해 원래 데이터를 접근하는 시간이 오래 걸리는 경우나 값을 다시 계산하는 시간을 절약하고 싶은 경우에 사용한다. 캐시에 데이터를 미리 복사해 놓으면 계산이나 접근 시간 없이 더 빠른 속도로 데이터에 접근할 수 있다.

캐시는 시스템의 효율성을 위해 여러 분야에서 두루 쓰인다.

스토리지에서 코드로

- 반복되는 데이터, 저장하려면?
 - `cache.set('something-key', data)`
- 조회해보고 데이터가 없으면 갱신하려면?

```
2 def cached_function():  
3     data = cache.get(key)  
4     if data is None:  
5         data = actual_function()  
6         cache.set(key)  
7     return data
```

스토리지에서 코드로

- 반복되는 데이터, 저장하려면?
- `cache.set('some-key', ...)`
- 조회해보고 데이터가 있는가?

코드에 집중하자



```
2 def cache_decorator(function):
3     data = {}
4     def wrapper(*args, **kwargs):
5         key = function.__name__ + str(args) + str(kwargs)
6         if key not in data:
7             data[key] = function(*args, **kwargs)
```

스토리지에서 코드로

```
@ring.func.memcache(client)
def function():
    ...
```

```
data = function.get()
```

```
data = function.get_or_update()
```

```
data = function()
```

장고 캐시도 해주던데?

```
from django.views.decorators.cache import cache_control
```

```
@cache_control(max_age=3600)
```

```
def my_view(request):
```

```
...
```



장고 캐시도...

```
from django.core.cache import cache
from django.http import HttpRequest
from django.utils.cache import get_cache_key
```

```
def expire_page(path):
    request = HttpRequest()
    request.path = path
    key = get_cache_key(request)
    if cache.has_key(key):
        cache.delete(key)
```



**Smartness is good,
Explicitness is better.**

Ring

Smart

```
2 @ring.func.memcache(client, time=600):  
3 def function():  
4     return ...
```

Ring

Smart

```
2 @ring.func.memcache(client, time=600):  
3 def function():  
4     return ...
```

Explicit

```
7 function.delete() # delete in a line
```

Ring

Smart

```
2 @ring.func.memcache(client, time=600):  
3 def function():  
4     return ...
```

Explicit

```
7 function.delete() # delete in a line
```

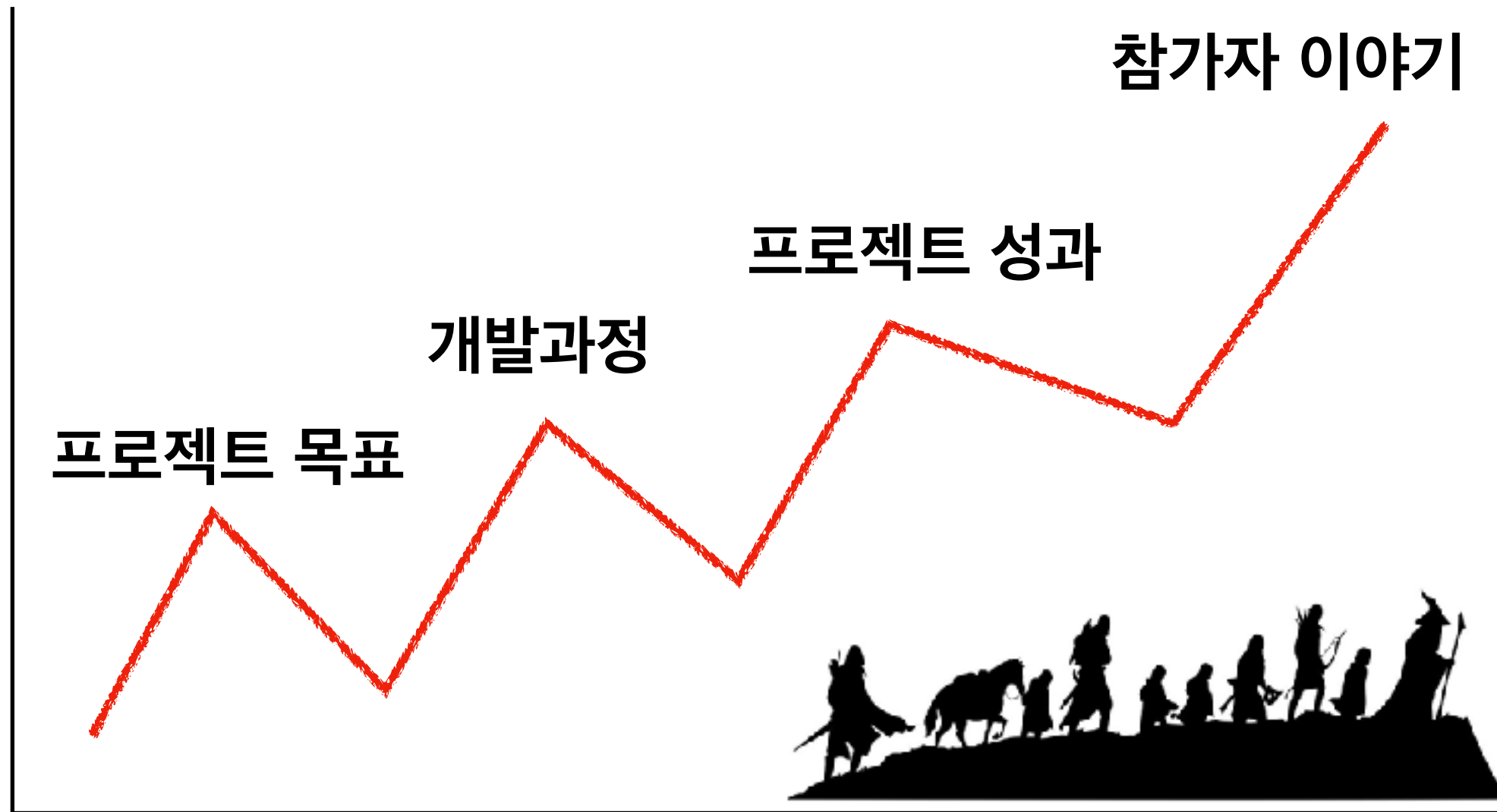
Fully Controllable

```
10 key = function.key()  
11 client.get(key)
```

Ring

- 캐시 프로그래밍의 중심이 스토리지에서 코드로 옮겨옵니다.
- 코드에 집중하고, 스토리지는 필요할때 컨트롤 합시다.
- 그런데 정말로 쉽게

반지 원정대



반지 원정대

개발 시작 전

”

와아 나도 오픈소스라는걸 해보는구나~



반지 원정대

개발 시작 후



반지를 얻는 길은 험난하리니...



프로젝트 목표

오픈소스에 참여하는 경험!

- 코드든 문서든 1건 이상 수정해보자!

- 코딩에 자신이 있거나 파이썬의 속을 파헤치고 싶다면 어울리는 과제도 준비되어 있습니다 😊
- 아무것도 못하고 있지는 말아요! 정 무얼할지 모르겠으면, 저에게 말을 걸어 주세요. 할 수 있는 일을 함께 찾아봅시다.

- 세 가지 참여 방법
 1. 코드 고치기
 2. 문서 고치기 / 추가하기
 3. Ring을 다른 프로젝트에 적용하기

개발 과정

- 온라인 활동

Slack을 활용한
멘토님의 신속 피드백



Soonmok Kwon 9:27 PM

unpack_coder 함수가 이미 json = JsonCoder 를 대체할수 있지 않나요 ?

unpack_coder 의 파라미터를 registry에서 받아오면 되는건가요 ?

1

정운원 9:28 PM

예제로 올려드린 코드에서 사용자가 새 coder를 추가하는 시나리오를 중심으로 봐주시면 좋겠어요
나머진 부수적입니다

```
ring.coder.pickle = PickleCoder # to register a new coder as name of pickle (edited)
```

이게 이상하디는거죠

```
registry.register('pickle', PickleCoder)
```

이렇게 되어야 한다는 것

GitHub 댓글로
의견 교환



yunitto commented 14 days ago

Contributor



python2에서도 돌아가게 하려면 기존의

```
raise TypeError("run() missing 1 required keyword argument: 'action'") from exc
```

를 다음과 같이 바꾸면 될 것 같습니다.

```
raise_from(TypeError("run() missing 1 required keyword argument: 'action'"), exc)
```

[stackoverflow](#) 를 참고했습니다.


개발 과정

- 오프라인 활동: 매 주말 1시~7시 모임

Thursday, October 19th

1 정윤원 6:02 PM
10월 21일-22일 주말은 양일 가운데 하루를 신청받겠습니다. 신청 마감은 20일 밤 24시까지입니다.
참가 희망하시는 분은 이모티콘으로 **1** **2** 중 원하시는 날짜에 표시해주세요

1 4 **2** 4

1  14 replies Last reply 29 days ago

1 정윤원 7:54 PM
치킨도 드시고 코딩도 하러 오세요
봉은사로 303
TGL 경복빌딩 502호, 선정릉역에서 도보 10분 내외)
시간 : 양일 모두 오전 10시~오후 7시까지 대관
eBrain space
@channel 간식 음료 치킨 드시며 코딩하러 오세요

개발 과정

- 즐거운 Tech-Talk

Code editor에 대한 토론을 한 날

Sunday, November 12th

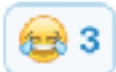
1

정윤원 6:32 PM

@yunitto @Hyunjoon

If you already know how to use Vim or Emacs then feel free to use them. If you have never used Vim or Emacs then avoid them. Programmers may try to convince you to use Vim or Emacs, but this will only derail you. Your focus is learning Python, not learning Vim or Emacs. If you try to use Vim and don't know how to quit, then type :q! or ZZ. If someone told you to use Vim, and they didn't even tell you this, then now you know why you shouldn't listen to them.

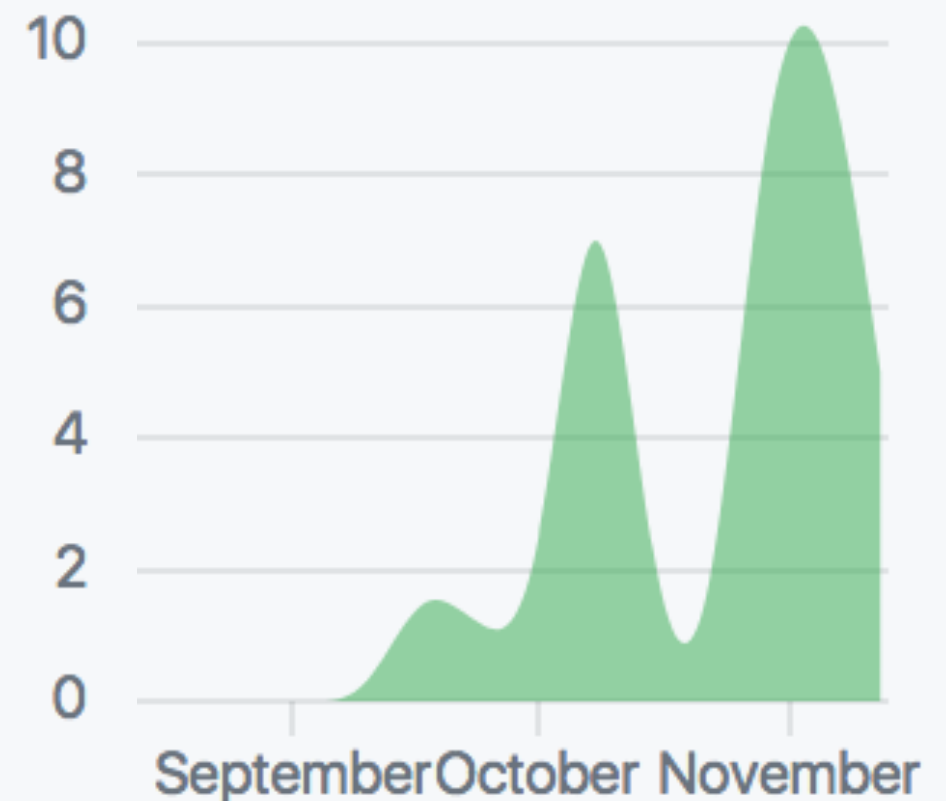
Vim이나 Emacs를 이미 쓸 줄 안다면 마음껏 쓰세요. 하지만 아직 한 번도 써 본 적이 없다면 쓰지 마세요. Vim이나 Emacs를 써야한다고 설득하려 드는 프로그래머들이 있을지도 모르는데, 그저 갈 길에서 멀어질 뿐입니다. 여러분의 목표는 파이썬을 배우는거예요. Vim이나 Emacs를 배우는게 아니고요. Vim을 써보려다가 끄지도 못하게 되었으면 키보드로 :q! 나 ZZ 를 입력하세요. 여러분에게 Vim을 써보라던 사람이 이것조차도 알려주지 않았다면, 왜 그 사람들 말을 들으면 안되는지 알겠죠?



프로젝트 성과

- 기여자 **7명**
- 커밋 **56개**
- 풀 리퀘스트 **31개**
- 새로운 기능 **6개**
- 닫은 이슈 **13개**

참여자 전원
오픈소스 경험 없음



프로젝트 성과

새로운 기능

- Action Trigger like `f.run('get')`
- Encode, Decode exposed to users
- File backend
- Pickle Coder
- Coder Registry

도큐멘테이션 개선

- 예제 코드 추가
- 문서화

버그 수정

- dict storage 동작 일관성

프로젝트 성과

부족했던 점

- 참가자 전부가 기여하지는 못함
 - 9/28 컨트리뷰톤 시작 직후 시작한 추석 연휴 (9/30~10/9)
(길고 긴 연휴는 행복했지만 코딩 실력은 감퇴했습니다)
- 대학생 참가자들의 중간고사
- Ring을 다른 프로젝트에 적용해보진 못함
 - 주로 코드수정, 문서 추가의 방식으로 참여

한 참가자의 이야기



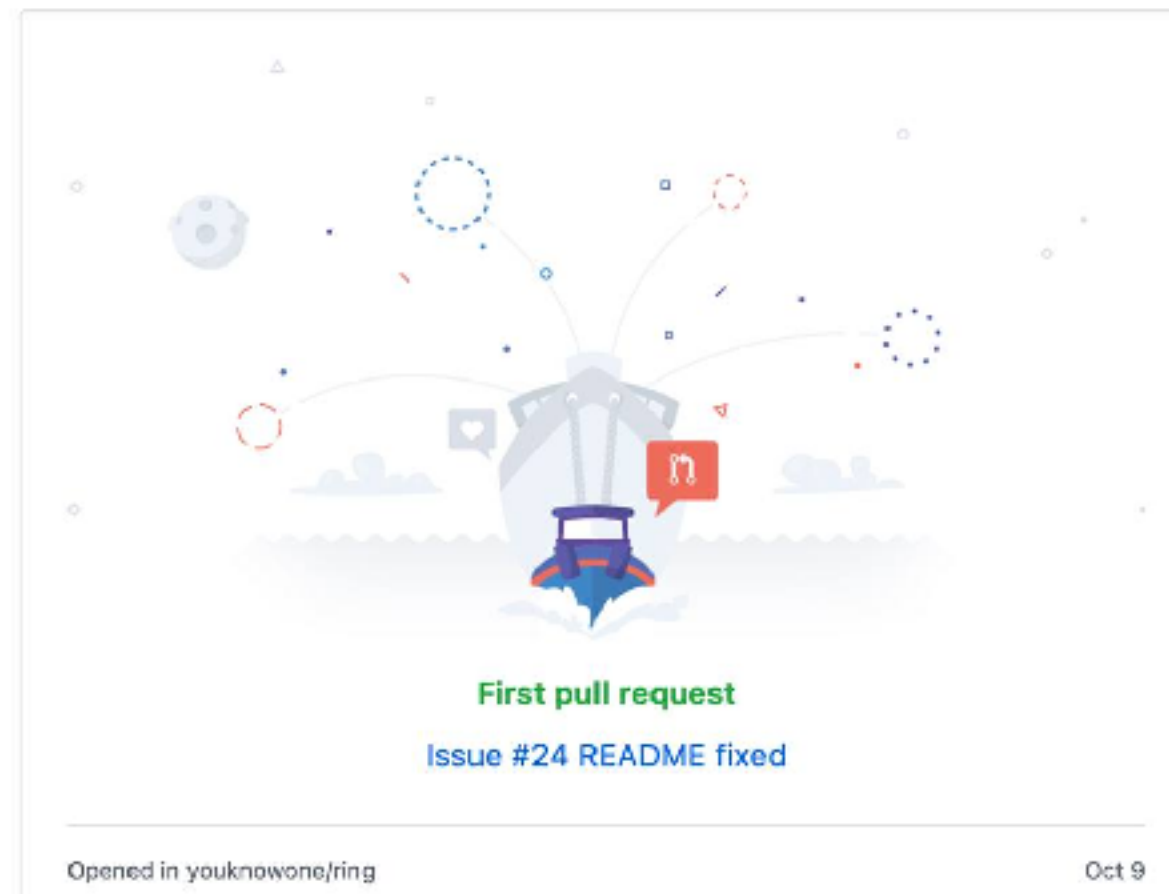
Hyesoo Yoo

yunitto

Computer Science Student

📍 Seoul

✉ hyesoooyoo.kr@gmail.com



한 참가자의 이야기






어려웠던 점

- 정류장에서 Github를 보다가 버스를 놓쳐서 지각
- 파이썬 데코레이터가 뭐지?

```
2 @ring.func.memcache(client, time=600):  
3 def function():  
4     return ...
```

- Python 2.7 / 3.3 / 3.4 / 3.5 / 3.6
 - 폐막식 3일 전에 새 버전이 릴리즈된 라이브러리
- Git fork? fetch? merge? pull? push?

Build Jobs

✓	# 203.1		</> Python: 2.7
✗	# 203.2		</> Python: 3.3
✗	# 203.3		</> Python: 3.4
✗	# 203.4		</> Python: 3.5
✗	# 203.5		</> Python: 3.6

한 참가자의 이야기

가장 크게 느낀 점

- 저 너머에 내가 지금은 보지 못하는 큰 그림이 있구나
- 나도 보고싶다 그거!
- 나도 만들고싶다 그런거!



한 참가자의 이야기

가장 크게 느낀 점

- 저 너머에 내가 지금은 보지 못하는 큰 그림이 있구나
 - 나도 보고싶다 그거!
 - 나도 만들고싶다 그런거!
 - **너가 정말 할 수 있겠어?**

한 참가자의 이야기

가장 크게 느낀 점

- 저 너머에 내가 지금은 보지 못하는 큰 그림이 있구나
- 나도 보고싶다 그거!
- 나도 만들고싶다 그런거!
- ~~너가 정말 할 수 있~~



Ring 엄청 좋아요 많이 사용해주세요

이런 분들께 추천드립니다!

- 캐시 컨트롤하기가 너무 복잡하신 분!
- 함수 코드 위에 @ 한줄 추가로 캐시를 구현하실 분!
- ~~불로불사의 절대반지를 얻고싶으신 분~~

၁ နာမံ ဘုရားရှင်၏ နာမံ ဘုရားရှင်၏
နာမံ ဘုရားရှင်၏ နာမံ ဘုရားရှင်၏

Ash nazg durbatulûk, ash nazg gimbatul,
One Ring to rule them all, One Ring to find them,
모든 **캐시**를 지배하고, 모든 **캐시**를 발견하는 것은 **절대반지**,

ash nazg thrakatulûk, agh burzum-ishi krimpatul.
One Ring to bring them all, And in the Storage bind them.
모든 **캐시**를 불러모아 **스토리지**에 가두는 것은 **절대반지**.

감사합니다!

좋은 행사 주최해 주셔서 감사합니다

정윤원 멘토님 감사합니다 :)

더욱 성장하겠습니다 \(^0^)/