동행복권 API 개발기

내가 짜고 내가 쓰는 라이브러리 - dhapi

AUSG 3기 문성혁

dhapi의 시작부터 미래까지

"이거 될 것 같은데" - 계획

"이게 왜 안 되지" - 구현 및 시행착오

"그건 미구현이예요" - 상상

이거 될 것 같은데

기원

기원

아주 먼 옛날, 매주 로그인하던 시절이 있었습니다

동행복권 사이트 접속

'로그인' 클릭

'자동추첨모드' 클릭

'확인' 클릭

로그인 페이지로 이동 (클릭)

'6/45 로또 구매' 팝업 오픈 (클릭)

'1장' ➡️ '5장'으로 변경 (클릭)

'구매하기' 클릭

기원

😇😇😊😊😊😊😊😊😊😊🤨🤨🤨🤨🤨😡

👆

기원

😇😇😊😊😊😊😊😊😊😊🤨🤨🤨🤨🤨😡

기원

Just FYI:

me: alias WS='cd /Users/roeniss/workspace'

me also: gcal, tldr, pet

me also again: glg, grst, dkp

기원

쇼타임

this sequence must be censored before uploaded to Youtube

이거 될 것 같은데

조건

조건

중간에 포기하지 않기 위하여

- 빠른 개발 for 멘탈

- 직관적인 설계 for 나를 포함한 미래의 컨트리뷰터들

- 쉬운 사용법 for 멘탈
  - 쉬운 배포 (릴리즈)
  - 쉬운 설치
  - 쉽고 직관적인 명령어
  - 빠른 실행 속도

이거 될 것 같은데

설계

설계

아키텍쳐를 그려보자

- 플로우 : 인증 ➡️ 요청. 끝.
  - No server : 보안 증가, 운영 리소스 감소

- 언어 : 파이썬
  - 크롤링 많이 해봄
  - requests는 정말 심플한 http library

- 동행복권 사이트 분석
  - 로그인, (매주) 복권 구매, JSessionId, …

설계

기능 분류

전체적인 API 구성을 간략하게 스케치

1) 로또: 9 타입, 구매 & 당첨 확인
2) 예치금 : 잔금 확인, 입금

➡️ 최종적으로는 "6/45로또:구매" 기능만 구현, but 확장 가능하게 설계

이게 왜 안 되지

세팅

세팅

- 가상환경 : venv (python3 -m venv .venv)
- 린팅 : black, pylint

settings.json

```
.vscode > { } settings.json > ...
1  {
2      "python.pythonPath": ".venv/bin/python",
3      "python.linting.pylintPath": ".venv/bin/pylint"
4  }
```

```
34  // Python
35  "python.pythonPath": "/usr/local/bin/python3", // OSX
36  "python.formatting.provider": "black",
37  "python.formatting.blackPath": "/usr/local/bin/black",
38  "python.formatting.blackArgs": ["-t", "py37"],
39  "python.linting.pylintArgs": [
40    "--disable=all",
41    "--enable=F,E,unreachable,duplicate-key,unnecessary-semico
42    "--disable=E0402"
43  ],
```

세팅

- 디렉토리

```
src/
└──── dhapi/
    ├──── lib/
    │     ├──── __init__.py
    │     ├──── auth.py
    │     ├──── controller.py
    │     └──── lotto645.py
    └──── __init__.py
```

세팅

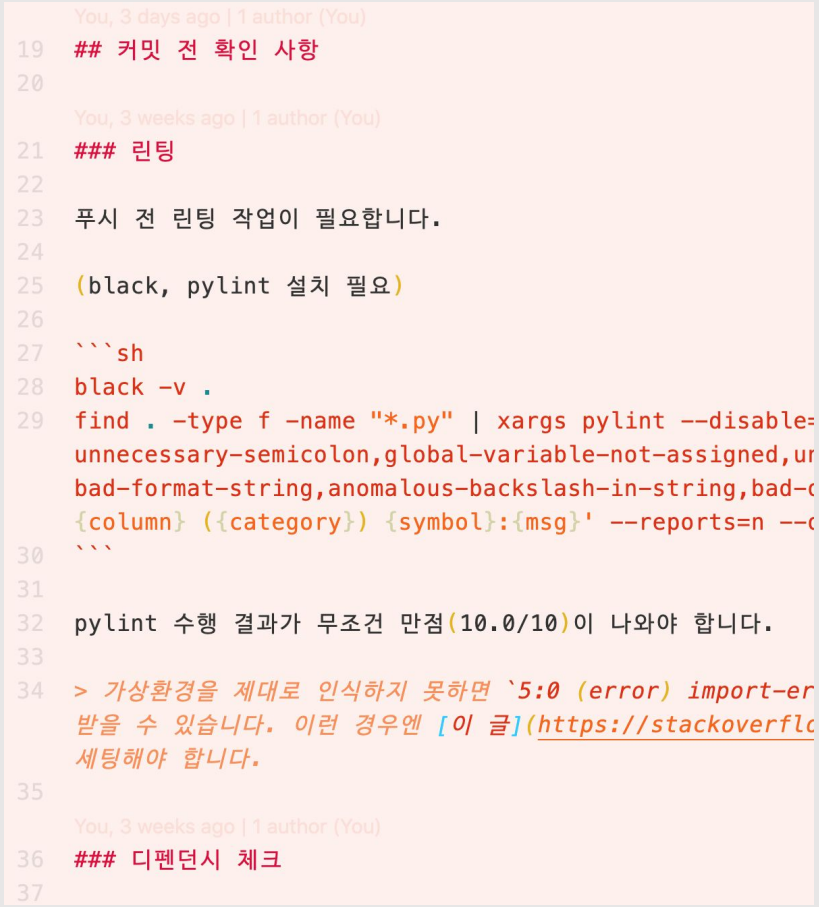
- 테스트

엄,, 내가 안해도 동행복권 서버가 확인해주니까..

세팅

- 배포 전략

별다른 CI/CD는 따로 없지만
컨트리뷰터들 (과 미래의 나)를 위해
CONTRIBUTING.md 작성



이게 왜 안 되지

개발

개발

개발은 재미없는 부분이니까 후루룩

tl;dr: requests로 로그인 ➡️ 세션 (쿠키) 획득 ➡️ 로또 구매 ➡️ 결과 출력

이게 왜 안 되지

배포

배포

홈브류로 만들자

내 패키지가 brew install로 설치가 가능하다고? 😎😎😎

brew install roeniss/dhapi/dhapi 성공! 🥳

그러나 …

배포

홈브류 아웃

- 너무 복잡한 배포 프로세스

소스코드 레포에서 릴리즈 생성 🤢
➡️ tar.gz 파일 만들어서 등록 🤢🤢🤢
➡️ homebrew-dhapi 레포의 내용 갱신 후 푸시 🤮🤮🤮🤮🤮

- Only for macOS

- 너어어어어어어어어무 느려! (caused by pyinstaller)

배포

운이 좋았다고 해야될까? pypi

- 파이썬이기 때문에 가능했음
- 추가 레포 불필요 (로컬에서 바로 배포)
- pip install dhapi (not 'pip install roeniss/dhapi')
- OS independent (음.. 일단 macOS, Windows10은 🤷‍♂️)
- 별로 친절하진 않지만, 어쨌든 공식문서만으로 배포 가능
  (혹시나 해보실 분은 'setup.py'가 필요하다는 것만 기억하세요)

이게 왜~~안~~ 되지

회고

배포

"이 정도면 됐다"

- 빠른 개발 for 멘탈 *by 익숙한 도구, 익숙한 방법*

- 직관적인 설계 for 나를 포함한 미래의 컨트리뷰터들
  by 개발 전 사이트 분석 및 계획 수립, CONTRIBUTING.md 작성

- 쉬운 사용법 for 멘탈
  - 쉬운 배포 (릴리즈) *by pypi*
  - 쉬운 설치 *by pypi*
  - 쉽고 직관적인 명령어 *by CLI style, inspired by `gcal`*
  - 빠른 실행 속도 *by pypi*

그건 미구현이예요

상상

배포

AWS를 끼얹어보자!

1) 매주 자동 구매 후 이메일로 내역 전송

배포

AWS를 끼얹어보자!

2) 마스터에 머지 시 자동으로 테스트 & bump versioning & 릴리즈



동행복권 API 개발기 (완)

pip install dhapi

기여하면 커피 쏨

전체 초안은 아래 발표자 노트에..

(구글 스프레드: https://drive.google.com/file/d/1DjeHYEmp7fByv1cTQB2lz64RJCU7yW2A/view?usp=sharing)