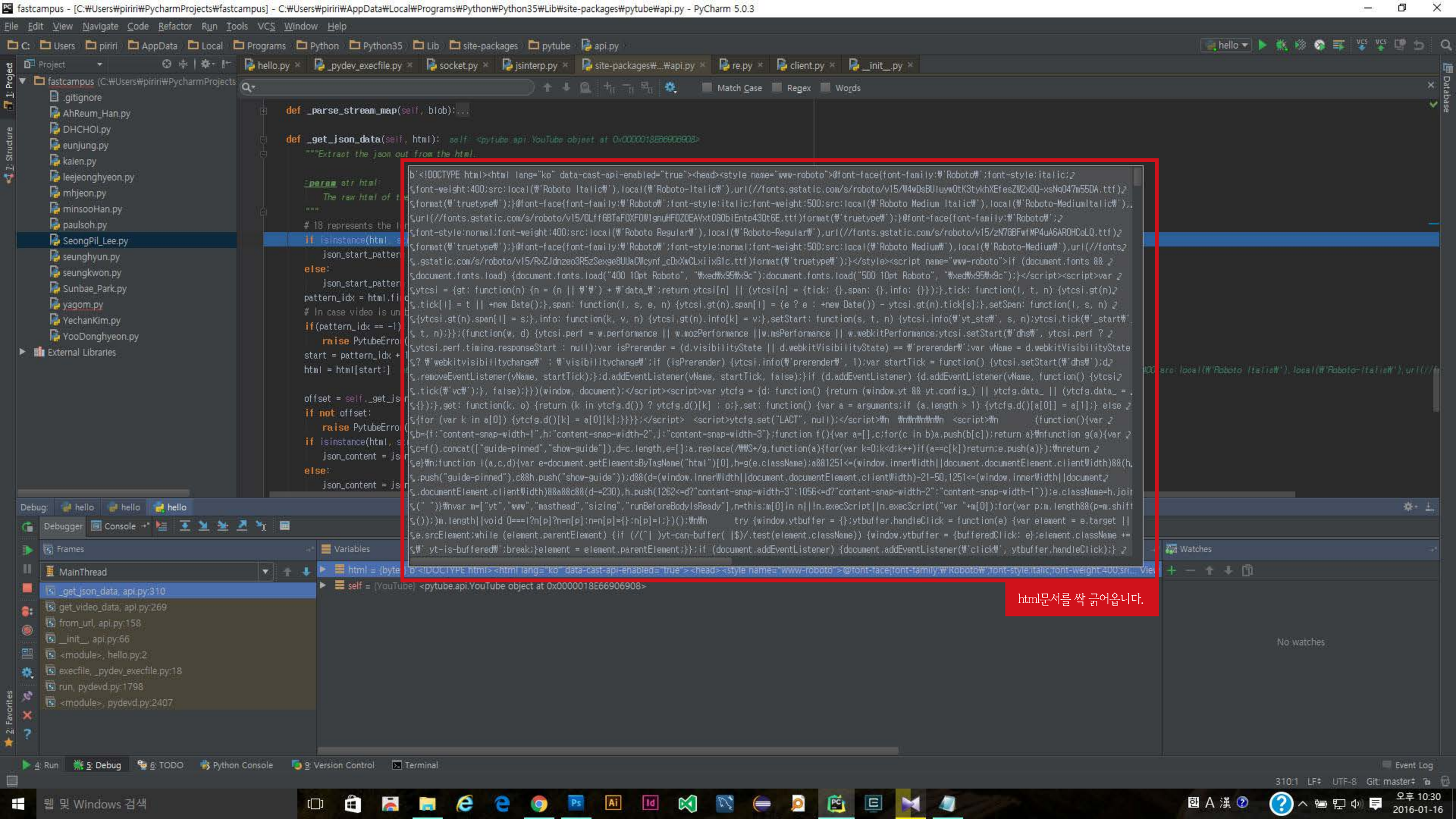
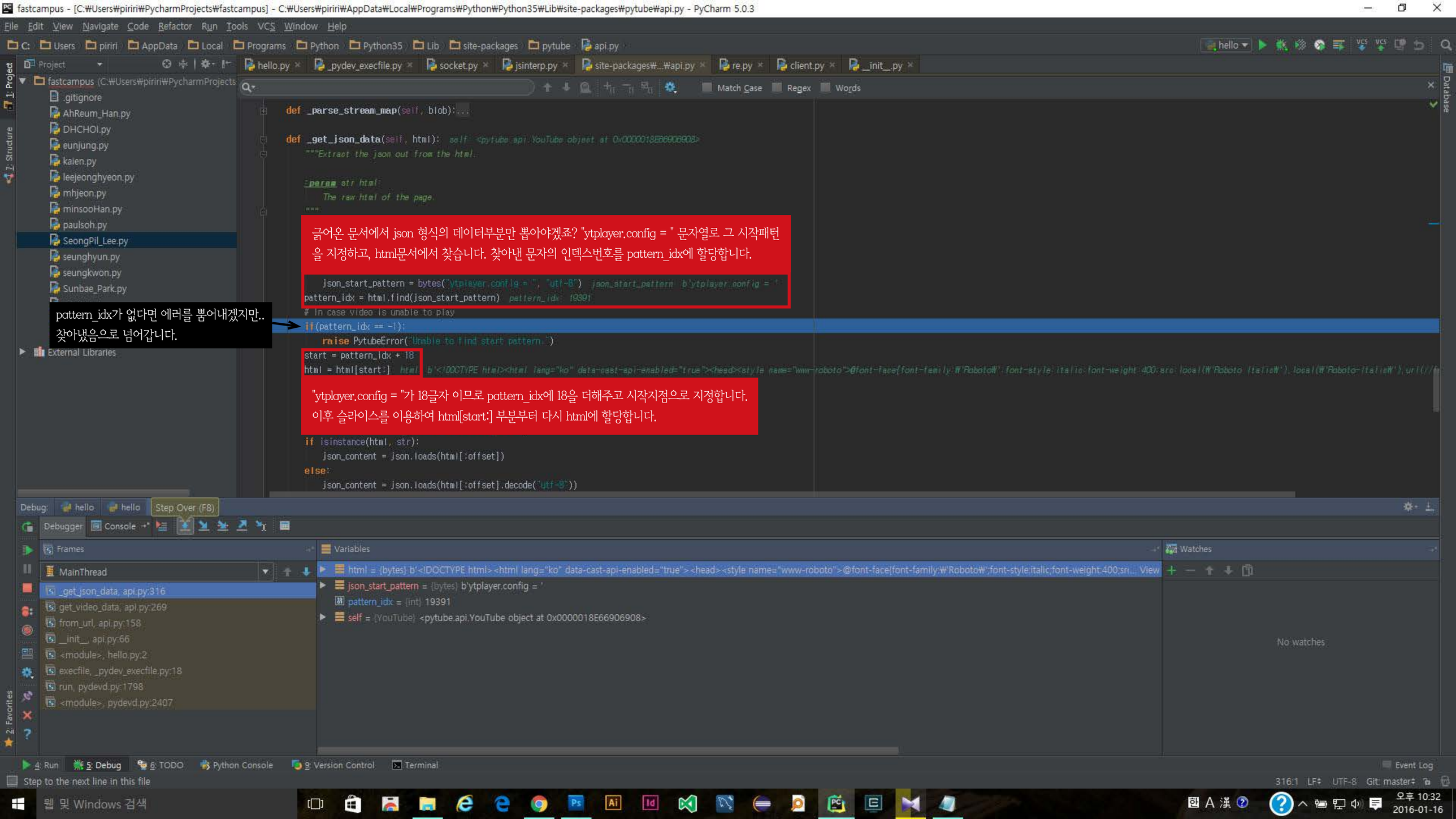
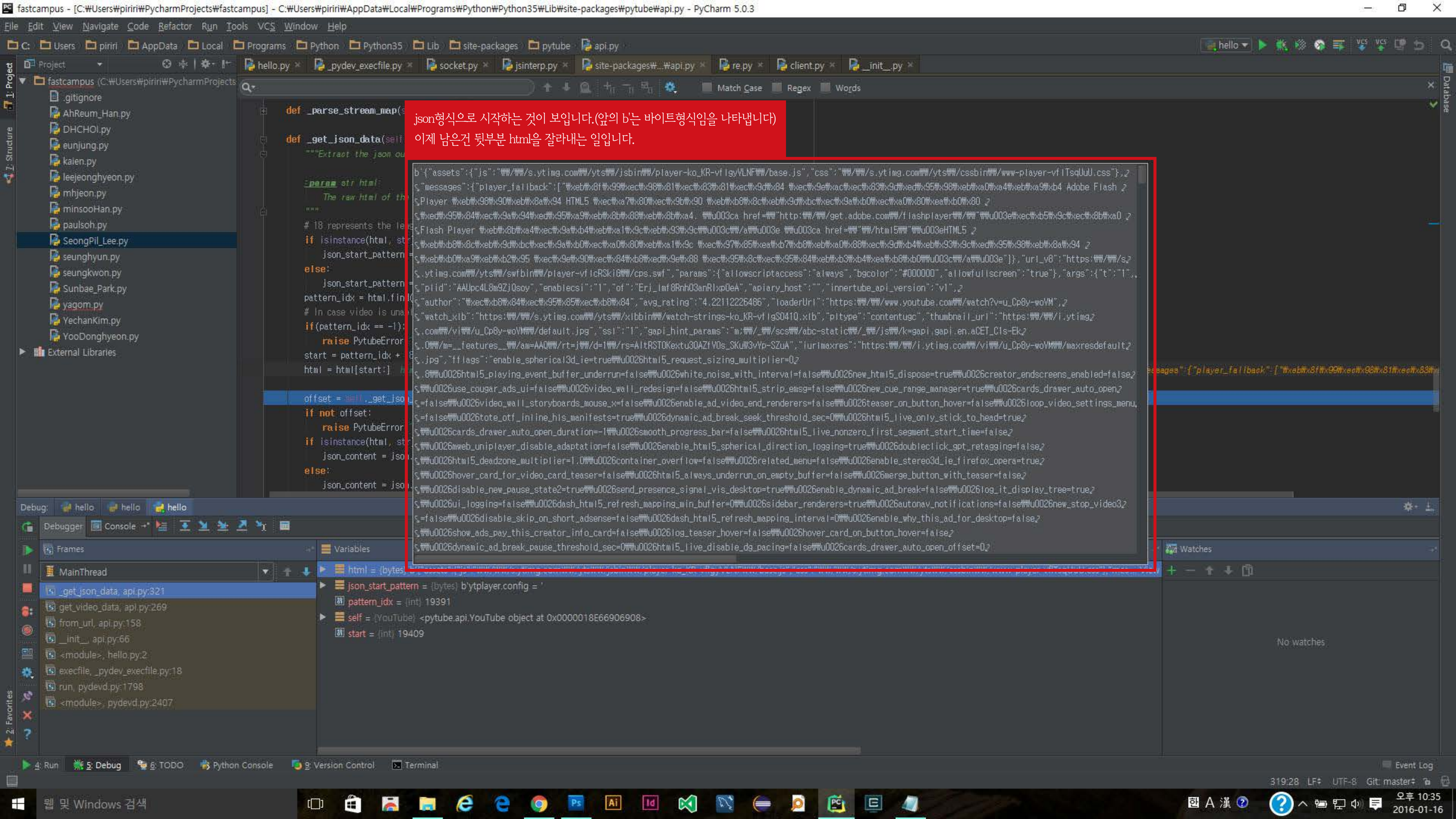

fastcampus - [C:₩Users₩piriri₩PycharmProjects₩fastcampus] - C:₩Users₩piriri₩AppData₩Local₩Programs₩Python₩Python35₩Lib₩site-packages₩pytube₩api.py - PyCharm 5.0.3
def _parse_stream_map(self, blob):...
def _get_json_data(self, html):
"""Extract the json out from the html.
b'<!DOCTYPE html><html lang="ko" data-cast-api-enabled="true"><head><style name="www-roboto">@font-face{font-family:'Roboto';font-style:italic;
html = {byte
self = {YouTube} <pytube.api.YouTube object at 0x0000018E66906908>
html문서를 싹 긁어옵니다.


fastcampus - [C:₩Users₩piriri₩PycharmProjects₩fastcampus] - C:₩Users₩piriri₩AppData₩Local₩Programs₩Python₩Python35₩Lib₩site-packages₩pytube₩api.py - PyCharm 5.0.3
File Edit View Navigate Code Refactor Run Tools VCS Window Help
hello.py _pydev_execfile.py socket.py jsinterp.py site-packages₩...₩api.py re.py client.py __init__.py
Match Case Regex Words
def _parse_stream_map(self, blob):...
def _get_json_data(self, html):
"""Extract the json out from the html.
:param str html:
The raw html of the page.
"""
json_start_pattern = bytes("ytplayer.config = ", "utf-8")
pattern_idx = html.find(json_start_pattern)
# In case video is unable to play
if(pattern_idx == -1):
raise PytubeError("Unable to find start pattern.")
start = pattern_idx + 18
html = html[start:]
if isinstance(html, str):
json_content = json.loads(html[:offset])
else:
json_content = json.loads(html[:offset].decode("utf-8"))
긁어온 문서에서 json 형식의 데이터부분만 뽑아야겠죠? "ytplayer.config = " 문자열로 그 시작패턴을 지정하고, html문서에서 찾습니다. 찾아낸 문자의 인덱스번호를 pattern_idx에 할당합니다.
pattern_idx가 없다면 에러를 뿜어내겠지만..
찾아냈음으로 넘어갑니다.
"ytplayer.config = "가 18글자 이므로 pattern_idx에 18을 더해주고 시작지점으로 지정합니다.
이후 슬라이스를 이용하여 html[start:] 부분부터 다시 html에 할당합니다.
Debug: hello hello Step Over (F8)
Debugger Console
Frames
MainThread
_get_json_data, api.py:316
get_video_data, api.py:269
from_url, api.py:158
__init__, api.py:66
<module>, hello.py:2
execfile, _pydev_execfile.py:18
run, pydevd.py:1798
<module>, pydevd.py:2407
Variables
html = {bytes} b'<!DOCTYPE html><html lang="ko" data-cast-api-enabled="true"><head><style name="www-roboto">@font-face{font-family:'Roboto';font-style:italic;font-weight:400;sr...
json_start_pattern = {bytes} b'ytplayer.config = '
pattern_idx = {int} 19391
self = {YouTube} <pytube.api.YouTube object at 0x0000018E66906908>
Watches
No watches
4: Run 5: Debug 6: TODO Python Console 9: Version Control Terminal
Event Log
Step to the next line in this file
316:1 LF UTF-8 Git: master
웹 및 Windows 검색
오후 10:32
2016-01-16


json형식으로 시작하는 것이 보입니다.(앞의 b'는 바이트형식임을 나타냅니다)
이제 남은건 뒷부분 html을 잘라내는 일입니다.

fastcampus - [C:₩Users₩piriri₩PycharmProjects₩fastcampus] - C:₩Users₩piriri₩AppData₩Local₩Programs₩Python₩Python35₩Lib₩site-packages₩pytube₩api.py - PyCharm 5.0.3

File Edit View Navigate Code Refactor Run Tools VCS Window Help

C: Users piriri AppData Local Programs Python Python35 Lib site-packages pytube api.py

hello

Project

fastcampus (C:₩Users₩piriri₩PycharmProjects

- .gitignore
- AhReum_Han.py
- DHCHOI.py
- eunjung.py
- kaien.py
- leejeonghyeon.py
- mhjeon.py
- minsooHan.py
- paulsoh.py
- SeongPil_Lee.py
- seunghyun.py
- seungkwon.py
- Sunbae_Park.py
- yagom.py
- YechanKim.py
- YooDonghyeon.py

External Libraries

hello.py _pydev_execfile.py socket.py jsinterp.py site-packages₩...₩api.py re.py client.py __init__.py

Match Case Regex Words

```
            json_content = json.loads(html[:offset])
        else:
            json_content = json.loads(html[:offset].decode("utf-8"))

        return json_content

    def _get_json_offset(self, html):  self: <pytube.api.YouTube object at 0x0000018E66906908>  html: b'{"assets":{"js":"##/##/s.ytimg.com##/yts##/jsbin##/player-ko_KR-vflgyVLNF##/base.js","css":"##/##/s.ytimg.com##/yts##/cssbin##/www-player-vflTsqUuU.cs
        """Find where the json object starts

        :param str html:
            The raw html of the YouTube page.
        """
        unmatched_brackets_num = 0
        index = 1
        for idx, ch in enumerate(html):
            if isinstance(ch, int):
                ch = chr(ch)
            if ch == "{":
                unmatched_brackets_num += 1
            elif ch == "}":
                unmatched_brackets_num -= 1
                if unmatched_brackets_num == 0:
                    break
        else:
            raise PytubeError("Unable to determine json offset.")
        return index + idx
```

여기부턴 노가다를 시작합니다. json 형식이 끝나는 부분을 찾을 때까지 탐색을 합니다.(자세한 로직은 잘 모르겠습니다...)

Debug: hello hello hello

Debugger Console

Frames

MainThread

- _get_json_offset, api.py:337
- _get_json_data, api.py:321
- get_video_data, api.py:269
- from_url, api.py:158
- __init__, api.py:66
- <module>, hello.py:2
- execfile, _pydev_execfile.py:18
- run, pydevd.py:1798
- <module>, pydevd.py:2407

Variables

- html = {bytes} b'{"assets":{"js":"##/##/s.ytimg.com##/yts##/jsbin##/player-ko_KR-vflgyVLNF##/base.js","css":"##/##/s.ytimg.com##/yts##/cssbin##/www-player-vflTsqUuU.css"},"mes... View
- self = {YouTube} <pytube.api.YouTube object at 0x0000018E66906908>

Watches

No watches

4: Run 5: Debug 6: TODO Python Console 9: Version Control Terminal Event Log

337:1 LF: UTF-8 Git: master:

웹 및 Windows 검색

오후 10:36 2016-01-16

fastcampus - [C:₩Users₩piriri₩PycharmProjects₩fastcampus] - C:₩Users₩piriri₩AppData₩Local₩Programs₩Python₩Python35₩Lib₩site-packages₩pytube₩api.py - PyCharm 5.0.3

File Edit View Navigate Code Refactor Run Tools VCS Window Help

C: Users piriri AppData Local Programs Python Python35 Lib site-packages pytube api.py

hello

Project

fastcampus (C:₩Users₩piriri₩PycharmProjects

- .gitignore
- AhReum_Han.py
- DHCHOI.py
- eunjung.py
- kaien.py
- leejeonghyeon.py
- mhjeon.py
- minsooHan.py
- paulsoh.py
- SeongPil_Lee.py
- seunghyun.py
- seungkwon.py
- Sunbae_Park.py
- yagom.py
- YechanKim.py
- YooDonghyeon.py

External Libraries

hello.py _pydev_execfile.py socket.py jsinterp.py site-packages₩...₩api.py re.py client.py __init__.py

Match Case Regex Words

```
        else:
            json_start_pattern = bytes("ytplayer.config = ", "utf-8")  json_start_pattern: b'ytplayer.config = '
        pattern_idx = html.find(json_start_pattern)  pattern_idx: 19391
        # In case video is unable to play
        if(pattern_idx == -1):
                                                                                    )
                                                                                    ytimg.com₩₩/yts₩₩/jsbin₩₩/player-ko_KR-vflgyVLNF₩₩/base.js","css":"₩₩/₩₩/s.ytimg.com₩₩/yts₩₩/cssbin₩₩/www-player-vflTsqUuU.css"},"messages":{"player_fallback":["₩xeb₩x81₩x90₩xec₩x95₩x81₩xec₩x83₩x
        offset = self._get_json_offset(html)  offset: 30290
        if not offset:
            raise PytubeError("Unable to extract json.")
        if isinstance(html, str):
            json_content = json.loads(html[:offset])
        else:
            json_content = json.loads(html[:offset].decode(      ))
    def
        :param str html:
            The raw html of the YouTube page.
        """
        unmatched_brackets_num = 0
        index = 1
```

json형식이 끝나는 부분의 인덱스를 찾아냈습니다! 이를 offset에 할당합니다.

이제 html을 검사합니다. str 객체가 아니므로(*바이트코드 상태입니다.) else로 넘어가갑니다. html[:offset]이 됐습니다. 이제 완전한 json형식의 문장입니다. 이를 utf - 8 형식으로 디코드 한 후, 로드해서 json_content에 할당합니다.

Debug: hello hello hello

Debugger Console

Frames

MainThread

- _get_json_data, api.py:327
- get_video_data, api.py:269
- from_url, api.py:158
- __init__, api.py:66
- <module>, hello.py:2
- execfile, _pydev_execfile.py:18
- run, pydevd.py:1798
- <module>, pydevd.py:2407

Variables

- html = {bytes} b'{"assets":{"js":"₩₩/₩₩/s.ytimg.com₩₩/yts₩₩/jsbin₩₩/player-ko_KR-vflgyVLNF₩₩/base.js","css":"₩₩/₩₩/s.ytimg.com₩₩/yts₩₩/cssbin₩₩/www-player-vflTsqUuU.css"},"mes... View
- json_start_pattern = {bytes} b'ytplayer.config = '
- offset = {int} 30290
- pattern_idx = {int} 19391
- self = {YouTube} <pytube.api.YouTube object at 0x0000018E66906908>
- start = {int} 19409

Watches

No watches

4: Run 5: Debug 6: TODO Python Console 9: Version Control Terminal Event Log

327:1 LF UTF-8 Git: master

웹 및 Windows 검색

오후 10:48 2016-01-16

fastcampus - [C:₩Users₩piriri₩PycharmProjects₩fastcampus] - C:₩Users₩piriri₩AppData₩Local₩Programs₩Python₩Python35₩Lib₩site-packages₩pytube₩api.py - PyCharm 5.0.3

File Edit View Navigate Code Refactor Run Tools VCS Window Help

C: Users piriri AppData Local Programs Python Python35 Lib site-packages pytube api.py

hello

Project

fastcampus (C:₩Users₩piriri₩PycharmProjects

- .gitignore
- AhReum_Han.py
- DHCHOI.py
- eunjung.py
- kaien.py
- leejeonghyeon.py
- mhjeon.py
- minsooHan.py
- paulsoh.py
- SeongPil_Lee.py
- seunghyun.py
- seungkwon.py
- Sunbae_Park.py
- yagom.py
- YechanKim.py
- YooDonghyeon.py

External Libraries

hello.py _pydev_execfile.py socket.py jsinterp.py site-packages₩...₩api.py __init__.py re.py client.py __init__.py

Match Case Regex Words

```
        else:
            json_start_pattern = bytes("ytplayer.config = ", "utf-8")  json_start_pattern: b'ytplayer.config = '
        pattern_idx = html.find(json_start_pattern)  pattern_idx: 19391
        # In case video is unable to play
        if(pattern_idx == -1):
            raise PytubeError("Unable to find start pattern.")
        start = pattern_idx + 18  start: 19409
        html = html[start:]  html: b'{"assets":{"js":"₩₩/₩₩/s.ytimg.com₩₩/yts₩₩/jsbin₩₩/player-ko_KR-vflgyVLNF₩₩/base.js","css":"₩₩/₩₩/s.ytimg.com₩₩/yts₩₩/cssbin₩₩/www-player-vflTsqUuU.css"},"messages":{"player_fallback":["₩xeb₩x8f₩x99₩xec₩x95₩x8f₩xec₩x83₩x

        offset = self._get_json_offset(html)  offset: 30290
        if not offset:
            raise PytubeError("Unable to extract json.")
        if isinstance(html, str):
            json_content = json.loads(html[:offset])
        else:
            json_content = json.loads(html[:offset].decode("utf-8"))  json_content: {'html5': True, 'url': 'https://s.ytimg.com/yts/swfbin/player-vflcRSkiG/watch_as3.swf', 'params': {'allowfullscreen': 'true', 'bgcolor': '#000000', 'allowscriptaccess': 'alw

        return json_content

    de
        :param str html:
            The raw html of the YouTube page.
        """
        unmatched_brackets_num = 0
        index = 1
```

이로써 _get_json_data 메서드는 html 데이터를 인자로 받아 json_content를 반환한다는 사실을 알게됩니다.

Debug: hello hello hello

Debugger Console

Frames

MainThread

- _get_json_data, api.py:329
- get_video_data, api.py:269
- from_url, api.py:158
- __init__, api.py:66
- <module>, hello.py:2
- execfile, _pydev_execfile.py:18
- run, pydevd.py:1798
- <module>, pydevd.py:2407

Variables

- html = {bytes} b'{"assets":{"js":"₩₩/₩₩/s.ytimg.com₩₩/yts₩₩/jsbin₩₩/player-ko_KR-vflgyVLNF₩₩/base.js","css":"₩₩/₩₩/s.ytimg.com₩₩/yts₩₩/cssbin₩₩/www-player-vflTsqUuU.css"},"mes... View
- json_content = {dict} {'html5': True, 'url': 'https://s.ytimg.com/yts/swfbin/player-vflcRSkiG/watch_as3.swf', 'params': {'allowfullscreen': 'true', 'bgcolor': '#000000', 'allowscriptaccess': 'always'}, '... View
- json_start_pattern = {bytes} b'ytplayer.config = '
- offset = {int} 30290
- pattern_idx = {int} 19391
- self = {YouTube} <pytube.api.YouTube object at 0x0000018E66906908>
- start = {int} 19409

Watches

No watches

4: Run 5: Debug 6: TODO Python Console 9: Version Control Terminal Event Log

329:1 LF: UTF-8 Git: master:

웹 및 Windows 검색

오후 10:51 2016-01-16


fastcampus - [C:₩Users₩piriri₩PycharmProjects₩fastcampus] - C:₩Users₩piriri₩AppData₩Local₩Programs₩Python₩Python35₩Lib₩site-packages₩pytube₩api.py - PyCharm 5.0.3
restriction_pattern = "og:restrictions:age"
else:
restriction_pattern = bytes("og:restrictions:age", "utf-8")
if restriction_pattern in html:
raise AgeRestricted("Age restricted video. Unable to download "
"without being signed in.")
# Extract out the json data from the html response body.
json_object = self._get_json_data(html)
반환된 json_content를 json_object로 할당
into the json object. We
# do this just so we just can return one object for the video data.
encoded_stream_map = json_object.get("args", {}).get(
"url_encoded_fmt_stream_map")
json_object["args"]["stream_map"] = self._parse_stream_map(
encoded_stream_map)
return json_object
def _parse_stream_map(self, blob):...
def _get_json_data(self, html):
"""Extract the json out from the html.
:param str html:
The raw html of the page.
"""
Frames
MainThread
get_video_data, api.py:269
from_url, api.py:158
__init__, api.py:66
<module>, hello.py:2
execfile, _pydev_execfile.py:18
run, pydevd.py:1798
<module>, pydevd.py:2407
Variables
html = {bytes} b'<!DOCTYPE html> <html lang="ko" data-cast-api-enabled="true"> <head> <style name="www-roboto">@font-face{font-family:₩'Roboto₩';font-style:italic;font-weight:400;src...
response = {HTTPResponse} <http.client.HTTPResponse object at 0x0000018E6698DE10>
restriction_pattern = {bytes} b'og:restrictions:age'
self = {YouTube} <pytube.api.YouTube object at 0x0000018E66906908>
Watches
No watches

fastcampus - [C:₩Users₩piriri₩PycharmProjects₩fastcampus] - C:₩Users₩piriri₩AppData₩Local₩Programs₩Python₩Python35₩Lib₩site-packages₩pytube₩api.py - PyCharm 5.0.3

File Edit View Navigate Code Refactor Run Tools VCS Window Help

C: Users piriri AppData Local Programs Python Python35 Lib site-packages pytube api.py

hello

Project

fastcampus (C:₩Users₩piriri₩PycharmProjects

.gitignore
AhReum_Han.py
DHCHOI.py
eunjung.py
kaien.py
leejeonghyeon.py
mhjeon.py
minsooHan.py
paulsoh.py
SeongPil_Lee.py
seunghyun.py
seungkwon.py
Sunbae_Park.py
yagom.py
YechanKim.py
YooDonghyeon.py
External Libraries

hello.py _pydev_execfile.py socket.py jsinterp.py site-packages₩...₩api.py __init__.py re.py client.py __init__.py

Match Case Regex Words

```
        restriction_pattern = "og:restrictions:age"
    else:
        restriction_pattern = bytes("og:restrictions:age", "utf-8")  restriction_pattern: b'og:restrictions:age'

    if restriction_pattern in html:
        raise AgeRestricted('Age restricted video. Unable to download '
                            'without being signed in.')

    # Extract out the json data from the html response body.
    json_object = self._get_json_data(html)  json_object: {'html5': True, 'url': 'https://s.ytimg.com/yts/swfbin/player-vflcRSkiG/watch_as3.swf', 'params': {'allowfullscreen': 'true', 'bgcolor': '#000000', 'allowscriptaccess': 'always'}, 'url_v9as2': 'h

    # Here we decode the stream map and bundle it into the json object. We
    # do this just so we just can return one object for the video data.
    encoded_stream_map = json_object.get("args", {}).get(  encoded_stream_map: 'type=video%2Fmp4%3B+codecs%3D%22avc1.64001F%2C+mp4a.40.2%22&itag=22&fallback_host=tc.v17.cache3.googlevideo.com&url=https%3A%2F%2Fr7---an-ab02a0nfpgxapox-jwwk.googlevideo.co
        "url_encoded_fmt_stream_map")
    json_object[    ][                ] = self._parse_stream_map(
        encoded_stream_map)
```

json_object에서 "args"키로 찾은 밸류를 {} 빈 딕셔너리에 넣고 또 그 밸류에서
url_encoded_fmt_stream_map키?를 찾아내고?? 이를 encoded_stream_map에
할당합니다.(이해가 잘 안가는 부분..)
_parse_stream_map메서드 인자로 encoded_stream_map을 넣습니다.
이후 반환한 값을 json_object의 ["args"]값의 [stream_map]의 값으로 설정합니다.

```
        The raw html of the page.
    """
```

Debug: hello hello hello

Debugger Console

Frames

MainThread

get_video_data, api.py:275
from_url, api.py:158
__init__, api.py:66
<module>, hello.py:2
execfile, _pydev_execfile.py:18
run, pydevd.py:1798
<module>, pydevd.py:2407

Variables

encoded_stream_map = {str} 'type=video%2Fmp4%3B+codecs%3D%22avc1.64001F%2C+mp4a.40.2%22&itag=22&fallback_host=tc.v17.cache3.googlevideo.com&url=https%3A%2F%2Fr7--... View
html = {bytes} b'<!DOCTYPE html><html lang="ko" data-cast-api-enabled="true"><head><style name="www-roboto">@font-face{font-family:₩'Roboto₩';font-style:italic;font-weight:400;src... View
json_object = {dict} {'html5': True, 'url': 'https://s.ytimg.com/yts/swfbin/player-vflcRSkiG/watch_as3.swf', 'params': {'allowfullscreen': 'true', 'bgcolor': '#000000', 'allowscriptaccess': 'always'}, 'u... View
response = {HTTPResponse} <http.client.HTTPResponse object at 0x0000018E6698DE10>
restriction_pattern = {bytes} b'og:restrictions:age'
self = {YouTube} <pytube.api.YouTube object at 0x0000018E66906908>

Watches

No watches

4: Run 5: Debug 6: TODO Python Console 9: Version Control Terminal Event Log

275:1 LF: UTF-8 Git: master:

웹 및 Windows 검색

오후 10:57
2016-01-16

fastcampus - [C:₩Users₩piriri₩PycharmProjects₩fastcampus] - C:₩Users₩piriri₩AppData₩Local₩Programs₩Python₩Python35₩Lib₩site-packages₩pytube₩api.py - PyCharm 5.0.3

File Edit View Navigate Code Refactor Run Tools VCS Window Help

C: Users piriri AppData Local Programs Python Python35 Lib site-packages pytube api.py

hello

Project

fastcampus (C:₩Users₩piriri₩PycharmProjects

- .gitignore
- AhReum_Han.py
- DHCHOI.py
- eunjung.py
- kaien.py
- leejeonghyeon.py
- mhjeon.py
- minsooHan.py
- paulsoh.py
- SeongPil_Lee.py
- seunghyun.py
- seungkwon.py
- Sunbae_Park.py
- yagom.py
- YechanKim.py
- YooDonghyeon.py

External Libraries

hello.py _pydev_execfile.py socket.py jsinterp.py site-packages₩...₩ __init__.py

잠시 _parse_stream_map메서드를 알아보기로 합니다.
아까 할당한 encoded_stream_map을 blob으로 받습니다.

```
        # Extract out the json data from the html response body.
        json_object = self._get_json_data(html)

        # Here we decode the stream map and bundle it into the json object. We
        # do this just so we just can return one object for the video data,
        encoded_stream_map = json_object.get("args", {}).get(
            "url_encoded_fmt_stream_map")
        json_object["args"]["stream_map"] = self._parse_stream_map(
            encoded_stream_map)
        return json_object

    def _parse_stream_map(self, blob):  self: <pytube.api.YouTube object at 0x0000018E66906908>  blob: 'type=video%2Fmp4%3B+codecs%3D%22avc1.64001F%2C+mp4a.40.2%22&itag=22&fallback_host=tc.v17.cache3.googlevideo.com&url=https%3A%2F%2Fr7---sn-ab02a0nfpgxa
        """A modified version of `urlparse.parse_qs` that's able to decode
        YouTube's stream map.

        :param str blob:
            An encoded blob of text containing the stream map data.
        """
        dct = defaultdict(list)
        videos = [video.split("&") for video in videos]

        # Split at the equals sign so we can break this key value pairs and
```

유사 딕셔너리 객체를 반환하는 defaultdict 메서드 인자로 list를 넣습니다.
이를 dct에 할당합니다. 대충 '딕셔너리 안에 리스트가 있나?'로
이해하고 넘어갔습니다...ㅠㅠ

Debug: hello hello hello

Debugger Console

Frames

MainThread

- _parse_stream_map, api.py:286
- get_video_data, api.py:276
- from_url, api.py:158
- __init__, api.py:66
- <module>, hello.py:2
- execfile, _pydev_execfile.py:18
- run, pydevd.py:1798
- <module>, pydevd.py:2407

Variables

blob = {str} 'type=video%2Fmp4%3B+codecs%3D%22avc1.64001F%2C+mp4a.40.2%22&itag=22&fallback_host=tc.v17.cache3.googlevideo.com&url=https%3A%2F%2Fr7---sn-ab02a0nfpg View

self = {YouTube} <pytube.api.YouTube object at 0x0000018E66906908>

Watches

No watches

4: Run 5: Debug 6: TODO Python Console 9: Version Control Terminal Event Log

286:1 LF UTF-8 Git: master

웹 및 Windows 검색

오후 11:00 2016-01-16

fastcampus - [C:₩Users₩piriri₩PycharmProjects₩fastcampus] - C:₩Users₩piriri₩AppData₩Local₩Programs₩Python₩Python35₩Lib₩site-packages₩pytube₩api.py - PyCharm 5.0.3

File Edit View Navigate Code Refactor Run Tools VCS Window Help

C: Users piriri AppData Local Programs Python Python35 Lib site-packages pytube api.py

hello

Project

fastcampus (C:₩Users₩piriri₩PycharmProjects

- .gitignore
- AhReum_Han.py
- DHCHOI.py
- eunjung.py
- kaien.py
- leejeonghyeon.py
- mhjeon.py
- minsooHan.py
- paulsoh.py
- SeongPil_Lee.py
- seunghyun.py
- seungkwon.py
- Sunbae_Park.py
- yagom.py
- YechanKim.py
- YooDonghyeon.py

External Libraries

hello.py | _pydev_execfile.py | socket.py | jsinterp.py | site-packages₩...₩api.py | __init__.py | re.py | client.py | __init__.py

Match Case Regex Words

```
        return json_object

    def _parse_stream_map(self, blob):  self = <pytube.api.YouTube object at 0x0000018E66906908>  blob = 'type=video%2Fmp4%3B+codecs%3D%22avc1.64001F%2C+mp4a.40.2%22&itag=22&fallback_host=tc.v17.cache3.googlevideo.com&url=https%3A%2F%2Fr7---sn-ab02a0nfpg%2
        """A modified version of `urlparse.parse_qs` that's able to decode
        YouTube's stream map.

        :param str blob:
            An encoded blob of text containing the stream map data.
        """
        dct = defaultdict(list)  dct = defaultdict(<class 'list'>, {})

        # Split the comma separated videos.
        videos = blob.split(",")

        # Unquote the characters and split to parameters.
        videos = [video.split(   ) for video in videos]  video=22&fallback_host=tc.v17.cache3.googlevideo.com&url=https%3A%2F%2Fr7---sn-ab02a0nfpgxapox-jwwk.googlevideo.com%2Fvideoplayback%

        # Split at the equals sign so we can break this key value pairs and
        # toss it into a dictionary.
        for video in videos:
            for kv in video:
                key, value = kv.split("=")
                dct[key].append(unquote(value))
        log.debug("decoded stream map: %s", dct)
        return dct

    def _get_json_data(self, html):
```

인자로 받은 blobd을 쉼표로 split 하고 videos에 할당합니다.
(앗! 컴프리헨션이다!)
videos에서 하나씩 꺼낸 배열을 video에 할당하고 그 video도
&으로 split하여 다시 videos에 할당합니다.

Debug: hello hello hello

Debugger Console

Frames

MainThread

- _parse_stream_map, api.py:292
- get_video_data, api.py:276
- from_url, api.py:158
- __init__, api.py:66
- <module>, hello.py:2
- execfile, _pydev_execfile.py:18
- run, pydevd.py:1798
- <module>, pydevd.py:2407

Variables

- blob = {str} 'type=video%2Fmp4%3B+codecs%3D%22avc1.64001F%2C+mp4a.40.2%22&itag=22&fallback_host=tc.v17.cache3.googlevideo.com&url=https%3A%2F%2Fr7---sn-ab02a0nfpg... View
- dct = {defaultdict} defaultdict(<class 'list'>, {})
- self = {YouTube} <pytube.api.YouTube object at 0x0000018E66906908>
- videos = {list} ['type=video%2Fmp4%3B+codecs%3D%22avc1.64001F%2C+mp4a.40.2%22&itag=22&fallback_host=tc.v17.cache3.googlevideo.com&url=https%3A%2F%2Fr7---sn-ab02a0nf... View

Watches

No watches

4: Run 5: Debug 6: TODO Python Console 9: Version Control Terminal Event Log

296:29 LF: UTF-8 Git: master:

웹 및 Windows 검색

오후 11:03 2016-01-16

fastcampus - [C:₩Users₩piriri₩PycharmProjects₩fastcampus] - C:₩Users₩piriri₩AppData₩Local₩Programs₩Python₩Python35₩Lib₩site-packages₩pytube₩api.py - PyCharm 5.0.3

File Edit View Navigate Code Refactor Run Tools VCS Window Help

C: Users piriri AppData Local Programs Python Python35 Lib site-packages pytube api.py

hello

Project

fastcampus (C:₩Users₩piriri₩PycharmProjects

- .gitignore
- AhReum_Han.py
- DHCHOI.py
- eunjung.py
- kaien.py
- leejeonghyeon.py
- mhjeon.py
- minsooHan.py
- paulsoh.py
- SeongPil_Lee.py
- seunghyun.py
- seungkwon.py
- Sunbae_Park.py
- yagom.py
- YechanKim.py
- YooDonghyeon.py

External Libraries

hello.py | _pydev_execfile.py | socket.py | jsinterp.py | site-packages₩...₩api.py | __init__.py | re.py | client.py | __init__.py

Match Case Regex Words

```
        videos = blob.split('&')

        # Unquote the characters and split to parameters.
        videos = [video.split('=') for video in videos]  videos: ['type=video%2Fmp4%3B+codecs%3D%22avc1.64001F%2C+mp4a.40.2%22&itag=22&fallback_host=tc.v17.cache3.googlevideo.com&url=https%3A%2F%2Fr7---sn-ab02a0nfpgxapox-jwwk.googlevideo.com%2Fvideoplaybac

        # Split at the equals sign so we can break this key value pairs and
        # toss it into a dictionary.
        for video in videos:
            for kv in video:
                key, value = kv.split('=')
                dct[key].append(unquote(value))
        log.debug("decoded stream map: %s", dct)
        return dct

    def _get_json_data(self, html):
        """Extract the json out from the html.

        :param str html:
            The raw html of the page.
        """
        # 18 represents the length of "ytplayer.config = ",
        if isinstance(html, str):
            json_start_pattern = "ytplayer.config = "
        else:
            json_start_pattern = bytes("ytplayer.config = ", "utf-8")
        pattern_idx = html.find(json_start_pattern)
        # In case video is unable to play
```

Debug: hello hello hello

Debugger Console

Frames

MainThread

- _parse_stream_map, api.py:292
- get_video_data, api.py:276
- from_url, api.py:158
- __init__, api.py:66
- <module>, hello.py:2
- execfile, _pydev_execfile.py:18
- run, pydevd.py:1798
- <module>, pydevd.py:2407

Variables

blob = {str} 'type=video%2Fmp4%3B+codecs%3D%22avc1.64001F%2C+mp4a.40.2%22&itag=22&fallback_host=tc.v17.cache3.googlevideo.com&url=https%3A%2F%2Fr7---sn-ab02a0nfpg' View

현재 videos는 리스트로 되어있습니다.

{}

ect at 0x0000018E66906908>

videos = {list} ['type=video%2Fmp4%3B+codecs%3D%22avc1.64001F%2C+mp4a.40.2%22&itag=22&fallback_host=tc.v17.cache3.googlevideo.com&url=https%3A%2F%2Fr7---sn-ab02a0nf View

Watches

No watches

4: Run 5: Debug 6: TODO Python Console 9: Version Control Terminal Event Log

292:1 LF UTF-8 Git: master

웹 및 Windows 검색

오후 11:06 2016-01-16

fastcampus - [C:₩Users₩piriri₩PycharmProjects₩fastcampus] - C:₩Users₩piriri₩AppData₩Local₩Programs₩Python₩Python35₩Lib₩site-packages₩pytube₩api.py - PyCharm 5.0.3

File Edit View Navigate Code Refactor Run Tools VCS Window Help

C: Users piriri AppData Local Programs Python Python35 Lib site-packages pytube api.py

hello

Project

fastcampus (C:₩Users₩piriri₩PycharmProjects

- .gitignore
- AhReum_Han.py
- DHCHOI.py
- eunjung.py
- kaien.py
- leejeonghyeon.py
- mhjeon.py
- minsooHan.py
- paulsoh.py
- SeongPil_Lee.py
- seunghyun.py
- seungkwon.py
- Sunbae_Park.py
- yagom.py
- YechanKim.py
- YooDonghyeon.py

External Libraries

hello.py _pydev_execfile.py socket.py jsinterp.py site-packages₩...₩api.py parse.py __init__.py re.py client.py __init__.py

Match Case Regex Words

```
    videos = blob.split(',')

    # Unquote the characters and split to parameters.
    videos = [video.split('&') for video in videos]  videos: [['type=video%2Fmp4%3B+codecs%3D%22avc1.64001F%2C+mp4a.40.2%22', 'itag=22', 'fallback_host=tc.v17.cache3.googlevideo.com', 'url=https%3A%2F%2Fr7---sn-ab02a0nfpg<apos>-jwvk.googlevideo.com%2Fvi

    # Split at the equals sign so we can break this key value pairs and
    # toss it into a dictionary
    for video in videos:  video: ['type=video%2Fmp4%3B+codecs%3D%22avc1.64001F%2C+mp4a.40.2%22', 'itag=22', 'fallback_host=tc.v17.cache3.googlevideo.com', 'url=https%3A%2F%2Fr7---sn-ab02a0nfpg<apos>-jwvk.googlevideo.com%2Fvideoplayback%3Fsource%3Dyoutube
        for kv in video:  kv: 'type=video%2Fmp4%3B+codecs%3D%22avc1.64001F%2C+mp4a.40.2%22'
            key, value = kv.split('=')  key: 'type'  value: 'video%2Fmp4%3B+codecs%3D%22avc1.64001F%2C+mp4a.40.2%22'
            dct[key].append(unquote(value))
    log.debug('decoded stream map: %s', dct)
    return dct
```

for문을 2번 돌려 순회합니다. videos 속에 video들을 다시 kv로 순회하고
kv를 =로 split하여 key와 value로 나눕니다.
하까 만든 dct에서 dct[key]로 접근하여 value값을 append합니다.
unquote 메서드는 각 하나의 문자를 %xx escapes로 대체하는 메서드라고
doc에 써있는데 더 찾기엔 시간이 없어서 넘어갔습니다.....

```
    if isinstance(html, str):
        json_start_pattern = "ytplayer.config = "
    else:
        json_start_pattern = bytes("ytplayer.config = ", "utf-8")
    pattern_idx = html.find(json_start_pattern)
    # In case video is unable to play
```

Debug: hello hello hello

Debugger Console

Frames

MainThread

- _parse_stream_map, api.py:299
- get_video_data, api.py:276
- from_url, api.py:158
- __init__, api.py:66
- <module>, hello.py:2
- execfile, _pydev_execfile.py:18
- run, pydevd.py:1798
- <module>, pydevd.py:2407

Variables

- blob = {str} 'type=video%2Fmp4%3B+codecs%3D%22avc1.64001F%2C+mp4a.40.2%22&itag=22&fallback_host=tc.v17.cache3.googlevideo.com&url=https%3A%2F%2Fr7---sn-ab02a0nfpg> View
- dct = {defaultdict} defaultdict(<class 'list'>, {})
- key = {str} 'type'
- kv = {str} 'type=video%2Fmp4%3B+codecs%3D%22avc1.64001F%2C+mp4a.40.2%22'
- self = {YouTube} <pytube.api.YouTube object at 0x0000018E66906908>
- value = {str} 'video%2Fmp4%3B+codecs%3D%22avc1.64001F%2C+mp4a.40.2%22'
- video = {list} ['type=video%2Fmp4%3B+codecs%3D%22avc1.64001F%2C+mp4a.40.2%22', 'itag=22', 'fallback_host=tc.v17.cache3.googlevideo.com', 'url=https%3A%2F%2Fr7---sn-ab02a0n View
- videos = {list} [['type=video%2Fmp4%3B+codecs%3D%22avc1.64001F%2C+mp4a.40.2%22', 'itag=22', 'fallback_host=tc.v17.cache3.googlevideo.com', 'url=https%3A%2F%2Fr7---sn-ab02a0 View

Watches

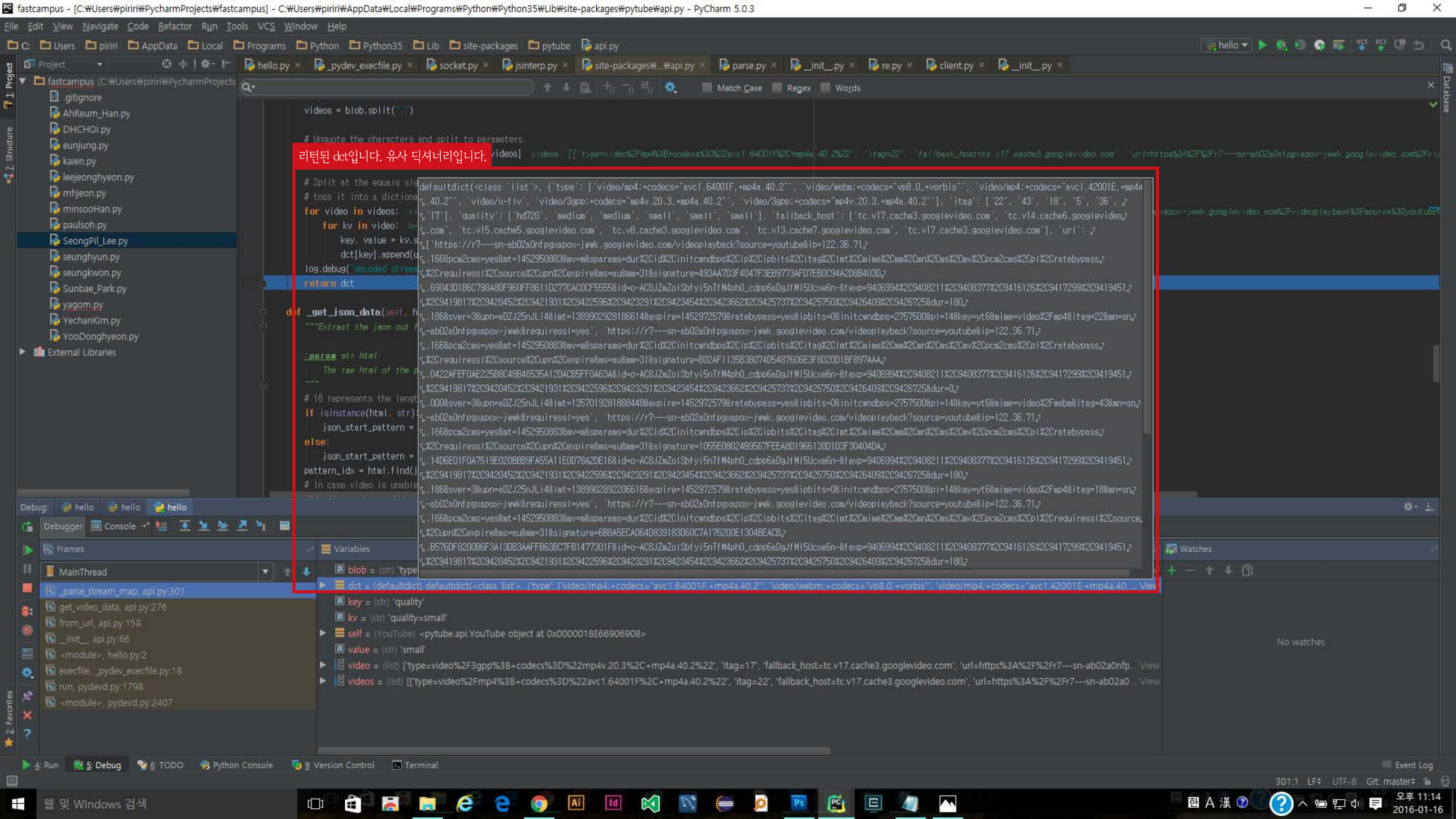
No watches

4: Run 5: Debug 6: TODO Python Console 9: Version Control Terminal Event Log

299:37 LF: UTF-8 Git: master:

웹 및 Windows 검색

오후 11:10 2016-01-16


fastcampus - [C:₩Users₩piriri₩PycharmProjects₩fastcampus] - C:₩Users₩piriri₩AppData₩Local₩Programs₩Python₩Python35₩Lib₩site-packages₩pytube₩api.py - PyCharm 5.0.3
File Edit View Navigate Code Refactor Run Tools VCS Window Help
hello.py _pydev_execfile.py socket.py jsinterp.py site-packages₩...₩api.py parse.py __init__.py re.py client.py __init__.py
Match Case Regex Words
videos = blob.split(",")
# Unquote the characters and split to parameters.
videos = [parse_qs(video) for video in videos]
리턴된 dct입니다. 유사 딕셔너리입니다.
defaultdict(<class 'list'>, {'type': ['video/mp4;+codecs="avc1.64001F,+mp4a.40.2"', 'video/webm;+codecs="vp8.0,+vorbis"', 'video/mp4;+codecs="avc1.42001E,+mp4a.40.2"', 'video/x-flv', 'video/3gpp;+codecs="mp4v.20.3,+mp4a.40.2"', 'video/3gpp;+codecs="mp4v.20.3,+mp4a.40.2"'], 'itag': ['22', '43', '18', '5', '36', '17'], 'quality': ['hd720', 'medium', 'medium', 'small', 'small', 'small'], 'fallback_host': ['tc.v17.cache3.googlevideo.com', 'tc.v14.cache6.googlevideo.com', 'tc.v15.cache5.googlevideo.com', 'tc.v8.cache3.googlevideo.com', 'tc.v13.cache7.googlevideo.com', 'tc.v17.cache3.googlevideo.com'], 'url': ['https://r7---sn-ab02a0nfpgxapox-jwwk.googlevideo.com/videoplayback?source=youtube&ip=122.36.71...
Debug: hello hello hello
Debugger Console
Frames
MainThread
_parse_stream_map, api.py:301
get_video_data, api.py:276
from_url, api.py:158
__init__, api.py:66
<module>, hello.py:2
execfile, _pydev_execfile.py:18
run, pydevd.py:1798
<module>, pydevd.py:2407
Variables
blob = {str} 'type
dct = {defaultdict} defaultdict(<class 'list'>, {'type': ['video/mp4;+codecs="avc1.64001F,+mp4a.40.2"', 'video/webm;+codecs="vp8.0,+vorbis"', 'video/mp4;+codecs="avc1.42001E,+mp4a.40...
key = {str} 'quality'
kv = {str} 'quality=small'
self = {YouTube} <pytube.api.YouTube object at 0x0000018E66906908>
value = {str} 'small'
video = {list} ['type=video%2F3gpp%3B+codecs%3D%22mp4v.20.3%2C+mp4a.40.2%22', 'itag=17', 'fallback_host=tc.v17.cache3.googlevideo.com', 'url=https%3A%2F%2Fr7---sn-ab02a0nfp...
videos = {list} [['type=video%2Fmp4%3B+codecs%3D%22avc1.64001F%2C+mp4a.40.2%22', 'itag=22', 'fallback_host=tc.v17.cache3.googlevideo.com', 'url=https%3A%2F%2Fr7---sn-ab02a0...
Watches
No watches
4: Run 5: Debug 6: TODO Python Console 9: Version Control Terminal
Event Log
301:1 LF UTF-8 Git: master
웹 및 Windows 검색
오후 11:14
2016-01-16

fastcampus - [C:₩Users₩piriri₩PycharmProjects₩fastcampus] - C:₩Users₩piriri₩AppData₩Local₩Programs₩Python₩Python35₩Lib₩site-packages₩pytube₩api.py - PyCharm 5.0.3

File Edit View Navigate Code Refactor Run Tools VCS Window Help

C: Users piriri AppData Local Programs Python Python35 Lib site-packages pytube api.py

hello

Project

fastcampus (C:₩Users₩piriri₩PycharmProjects

- .gitignore
- AhReum_Han.py
- DHCHOI.py
- eunjung.py
- kaien.py
- leejeonghyeon.py
- mhjeon.py
- minsooHan.py
- paulsoh.py
- SeongPil_Lee.py
- seunghyun.py
- seungkwon.py
- Sunbae_Park.py
- yagom.py
- YechanKim.py
- YooDonghyeon.py

External Libraries

hello.p jsinterp.py site-packages₩...₩api.py parse.py __init__.py re.py client.py __init__.py

Match Case Regex Words

돌아와서 드디어 get_video_data가
json_object를 리턴해낸 모습입니다.

```
def get_video_data(self):  self: <pytube.api.YouTube object at 0x0000018E66906908>
    """Gets the page and extracts out the video data."""
    # Reset the filename incase it was previously set.
    self.title = None
    response = urlopen(self.url)  response: <http.client.HTTPResponse object at 0x0000018E6698DE10>
    if not response:
        raise PytubeError("Unable to open url: {0}".format(self.url))

    html = response.read()  html: b'<!DOCTYPE html><html lang="ko" data-cast-api-enabled="true"><head><style name="www-roboto">@font-face{font-family:#'Roboto#';font-style:italic;font-weight:400;src:local(#'Roboto Italic#'),local(#'Roboto-Italic#'),url(
    if isinstance(html, str):
        restriction_pattern = "og:restrictions:age"
    else:
        restriction_pattern = bytes("og:restrictions:age", "utf-8")  restriction_pattern: b'og:restrictions:age'

    if restriction_pattern in html:
        raise AgeRestricted("Age restricted video. Unable to download "
                            "without being signed in.")

    # Extract out the json data from the html response body.
    json_object = self._get_json_data(html)  json_object: {'html5': True, 'url': 'https://s.ytimg.com/yts/swfbin/player-vflcRSkiG/watch_as3.swf', 'params': {'allowfullscreen': 'true', 'bgcolor': '#000000', 'allowscriptaccess': 'always'}, 'url_v9as2': 'h

    # Here we decode the stream map and bundle it into the json object. We
    # do this just so we just can return one object for the video data.
    encoded_stream_map = json_object.get("args", {}).get(  encoded_stream_map: 'type=video%2Fmp4%3B+codecs%3D%22avc1.64001F%2C+mp4a.40.2%22&itag=22&fallback_host=tc.v17.cache3.googlevideo.com&url=https%3A%2F%2Fr7---an-ab02a0nfpgxapox-jmwk.googlevideo.co
        "url_encoded_fmt_stream_map")
    json_object["args"]["stream_map"] = self._parse_stream_map(
        encoded_stream_map)
    return json_object

def _parse_stream_map(self, blob):
    """A modified version of `urlparse.parse_qs` that's able to decode
```

Debug: hello hello hello

Debugger Console

Frames

MainThread

- get_video_data, api.py:277
- from_url, api.py:158
- __init__, api.py:66
- <module>, hello.py:2
- execfile, _pydev_execfile.py:18
- run, pydevd.py:1798
- <module>, pydevd.py:2407

Variables

- encoded_stream_map = {str} 'type=video%2Fmp4%3B+codecs%3D%22avc1.64001F%2C+mp4a.40.2%22&itag=22&fallback_host=tc.v17.cache3.googlevideo.com&url=https%3A%2F%2Fr7--... View
- html = {bytes} b'<!DOCTYPE html><html lang="ko" data-cast-api-enabled="true"><head><style name="www-roboto">@font-face{font-family:#'Roboto#';font-style:italic;font-weight:400;src... View
- json_object = {dict} {'html5': True, 'url': 'https://s.ytimg.com/yts/swfbin/player-vflcRSkiG/watch_as3.swf', 'params': {'allowfullscreen': 'true', 'bgcolor': '#000000', 'allowscriptaccess': 'always'}, 'u... View
- response = {HTTPResponse} <http.client.HTTPResponse object at 0x0000018E6698DE10>
- restriction_pattern = {bytes} b'og:restrictions:age'
- self = {YouTube} <pytube.api.YouTube object at 0x0000018E66906908>

Watches

No watches

4: Run 5: Debug 6: TODO Python Console 9: Version Control Terminal Event Log

277:1 LF UTF-8 Git: master

웹 및 Windows 검색

오후 11:17
2016-01-16

fastcampus - [C:₩Users₩piriri₩PycharmProjects₩fastcampus] - C:₩Users₩piriri₩AppData₩Local₩Programs₩Python₩Python35₩Lib₩site-packages₩pytube₩api.py - PyCharm 5.0.3

File Edit View Navigate Code Refactor Run Tools VCS Window Help

C: Users piriri AppData Local Programs Python Python35 Lib site-packages pytube api.py

hello

Project

fastcampus (C:₩Users₩piriri₩PycharmProjects

- .gitignore
- AhReum_Han.py
- DHCHOI.py
- eunjung.py
- kaien.py
- leejeonghyeon.py
- mhjeon.py
- minsooHan.py
- paulsoh.py
- SeongPil_Lee.py
- seunghyun.py
- seungkwon.py
- Sunbae_Park.py
- yagom.py
- YechanKim.py
- YooDonghyeon.py

External Libraries

hello.py _pydev_execfile.py socket.py jsinterp.py site-packages₩...₩api.py parse.py __init__.py re.py client.py __init__.py

Match Case Regex Words

```
        :param str url:
            The url to the YouTube video.
        """
        self._video_url = url

        # Reset the filename and videos list in case the same instance is
        # reused.
        self._filename = None
        self._videos = []

        # Get the video details.
        video_data = self.get_video_data()

        # Rewrite and add the url to the javascript file, we'll need to fetch
        # this if YouTube doesn't provide us with the signature.
        js_url = "http:" + video_data.get("assets", {}).get("js")

        # Just make these easily accessible as variables.
        stream_map = video_data.get("args", {}).get("stream_map")
        video_urls = stream_map.get("url")

        # For each video url, identify the quality profile and add it to list
        # of available videos.
        for idx, url in enumerate(video_urls):
            log.debug("attempting to get quality profile from url: %s", url)
```

이제 get_video_data메서드가 반환한
json_object를 video_data에 할당합니다.

Debug: hello hello hello

Debugger Console

Frames

MainThread

- from_url, api.py:158
- __init__, api.py:66
- <module>, hello.py:2
- execfile, _pydev_execfile.py:18
- run, pydevd.py:1798
- <module>, pydevd.py:2407

Variables

- self = {YouTube} <pytube.api.YouTube object at 0x0000018E66906908>
- url = {str} 'https://www.youtube.com/watch?v=u_Cp8y-woVM'

Watches

No watches

제목 없음-1 @ 107% (레이어 ...

4: Run 5: Debug 6: TODO Python Console 9: Version Control Terminal Event Log

158:1 LF UTF-8 Git: master

웹 및 Windows 검색

오후 11:19
2016-01-16

fastcampus - [C:₩Users₩piriri₩PycharmProjects₩fastcampus] - C:₩Users₩piriri₩AppData₩Local₩Programs₩Python₩Python35₩Lib₩site-packages₩pytube₩api.py - PyCharm 5.0.3

File Edit View Navigate Code Refactor Run Tools VCS Window Help

C: Users piriri AppData Local Programs Python Python35 Lib site-packages pytube api.py

hello

Project

fastcampus (C:₩Users₩piriri₩PycharmProjects

.gitignore
AhReum_Han.py
DHCHOI.py
eunjung.py
kaien.py
leejeonghyeon.py
mhjeon.py
minsooHan.py
paulsoh.py
SeongPil_Lee.py
seunghyun.py
seungkwon.py
Sunbae_Park.py
yagom.py
YechanKim.py
YooDonghyeon.py
External Libraries

hello.py _pydev_execfile.py socket.py jsinterp.py site-packages₩...₩api.py parse.py __init__.py re.py client.py __init__.py

Match Case Regex Words

```
        self._video_url = url

        # Reset the filename and videos list in case the same instance is
        # reused.
        self._filename = None
        self._videos = []

        # Get the video details
        # Set the title from the title.
        self.title = video_data.get("args", {}).get("title")

        # Rewrite and add the url to the javascript file, we'll need to fetch
        # this if YouTube doesn't provide us with the signature.
        js_url = "http:" + video_data.get("assets", {}).get("js")

        # Just make these easily accessible as variables.
        stream_map = video_data.get("args", {}).get("stream_map")
        video_urls = stream_map.get("url")

        # For each video url, identify the quality profile and add it to list
        # of available videos.
        for idx, url in enumerate(video_urls):
            log.debug("attempting to get quality profile from url: %s", url)
            try:
                itag, quality_profile = self._get_quality_profile_from_url(url)
                if not quality_profile:
```

여기저기 쏙쏙 할당해 줍니다. video_data(딕셔너리형태)안에 값들을 이곳 저곳에 할당합니다.

Debug: hello hello hello

Debugger Console

Frames

MainThread

from_url, api.py:173
__init__, api.py:66
<module>, hello.py:2
execfile, _pydev_execfile.py:18
run, pydevd.py:1798
<module>, pydevd.py:2407

Variables

js_url = {str} 'http://s.ytimg.com/yts/jsbin/player-ko_KR-vflgyVLNF/base.js'
self = {YouTube} <pytube.api.YouTube object at 0x0000018E66906908>
stream_map = {defaultdict} defaultdict(<class 'list'>, {'type': ['video/mp4;+codecs="avc1.64001F,+mp4a.40.2"', 'video/webm;+codecs="vp8.0,+vorbis"', 'video/mp4;+codecs="avc1.42001E,+r... View
url = {str} 'https://www.youtube.com/watch?v=u_Co8v-woVM'
video_data = {dict} {'html5': True, 'url': 'https://s.ytimg.com/yts/swfbin/player-vflcRSkiG/watch_as3.swf', 'params': {'allowfullscreen': 'true', 'bgcolor': '#000000', 'allowscriptaccess': 'always'}, 'ur... View

Watches

No watches

4: Run 5: Debug 6: TODO Python Console 9: Version Control Terminal Event Log

Step to the next line in this file

173:1 LF UTF-8 Git: master

웹 및 Windows 검색

오후 11:23
2016-01-16

fastcampus - [C:₩Users₩piriri₩PycharmProjects₩fastcampus] - C:₩Users₩piriri₩AppData₩Local₩Programs₩Python₩Python35₩Lib₩site-packages₩pytube₩api.py - PyCharm 5.0.3

File Edit View Navigate Code Refactor Run Tools VCS Window Help

C: Users piriri AppData Local Programs Python Python35 Lib site-packages pytube api.py

hello

Project

fastcampus (C:₩Users₩piriri₩PycharmProjects

- .gitignore
- AhReum_Han.py
- DHCHOI.py
- eunjung.py
- kaien.py
- leejeonghyeon.py
- mhjeon.py
- minsooHan.py
- paulsoh.py
- SeongPil_Lee.py
- seunghyun.py
- seungkwon.py
- Sunbae_Park.py
- yagom.py
- YechanKim.py
- YooDonghyeon.py

External Libraries

hello.py _pydev_execfile.py socket.py jsinterp.py site-packages₩...₩api.py parse.py __init__.py re.py client.py __init__.py

Match Case Regex Words

```
        stream_map = video_data.get( args , {}).get( stream_map )  stream_map  defaultdict(<class 'list'>, {'type': ['video/mp4; codecs="avc1.64001F, mp4a.40.2"', 'video/webm; codecs="vp8.0, vorbis"', 'video/mp4; codecs="avc1.42001E, mp4a.40.2"', 'video/
        video_urls = stream_map.get("url")  video_urls  ['https://r7---sn-ab02a0nfpgxapox-jwwk.googlevideo.com/videoplayback?source=youtube&ip=122.36.71.166&pcm2cms=yes&mt=1452950883&mv=m&sparams=dur%2Cid%2Cinitcwndbps%2Cip%2Cipbits%2Citag%2Clmt%2Cmime%2Cmm

        # For each video url, identify the quality profile and add it to list
        # of available videos.
        for idx, url in enumerate(video_urls):
            log.debug("attempting to get quality profile from url: %s", url)
            try:
                itag, quality_profile = self._get_quality_profile_from_url(url)
                if not quality_profile:
                    log.warn("und
                    continue
            except (TypeError, K
                log.exception("p
                continue

            # Check if we have th
            # cipher from the js
            if "signature=" not
                log.debug("signa
                          "ciphe
                signature = self.
                url = "{0}&signat
            self._add_video(url,
        # Clear the cached js, M
        # `from_url()` so we can
        self._js_cache = None
```

['https://r7---sn-ab02a0nfpgxapox-jwwk.googlevideo.com/videoplayback?source=youtube&ip=122.36.71.166&pcm2cms=yes&mt=1452950883&mv=m&sparams=dur%2Cid%2Cinitcwndbps%2Cip%2Cipbits%2Citag%2Clmt%2Cmime%2Cmm%2Cmn%2Cms%2Cmv%2Cpcm2cms%2Cpl%2Cratebypass&signature=493AA7D3F4047F3EB9773AFD7EB0C94A2D8B403D.69043D1B6C798A8DF960FF8611D277CAC0CF5555&id=o-AC8JZmZolSbfyi5nTfM4phO_cdpp6aDgJfMI5Ucxe6n-8&exp=9406994%2C9408211%2C9408377%2C9416126%2C9417299%2C9419451%2C9419817%2C9420452%2C9421931%2C9422596%2C9423291%2C9423454%2C9423662%2C9425737%2C9425750%2C9426409%2C9426725&dur=180.1868&sver=3&upn=aDZJ25nJLi4&lmt=1389902928186614&expire=1452972579&ratebypass=yes&ipbits=0&initcwndbps=2757500&pl=14&key=yt6&mime=video%2Fmp4&itag=22&mn=sn-ab02a0nfpgxapox-jwwk&requiressl=yes', 'https://r7---sn-ab02a0nfpgxapox-jwwk.googlevideo.com/videoplayback?source=youtube&ip=122.36.71.166&pcm2cms=yes&mt=1452950883&mv=m&sparams=dur%2Cid%2Cinitcwndbps%2Cip%2Cipbits%2Citag%2Clmt%2Cmime%2Cmm%2Cmn%2Cms%2Cmv%2Cpcm2cms%2Cpl%2Cratebypass&signature=B02AF1135B3B074D5487606E3F8D20D1BF897AAA.0422AFEF0AE225B8C48B46535A12DACB5FF0A63A&id=o-AC8JZmZolSbfyi5nTfM4phO_cdpp6aDgJfMI5Ucxe6n-8&exp=9406994%2C9408211%2C9408377%2C9416126%2C9417299%2C9419451%2C9419817%2C9420452%2C9421931%2C9422596%2C9423291%2C9423454%2C9423662%2C9425737%2C9425750%2C9426409%2C9426725&dur=0.0008&sver=3&upn=aDZJ25nJLi4&lmt=1357019281888448&expire=1452972579&ratebypass=yes&ipbits=0&initcwndbps=2757500&pl=14&key=yt6&mime=video%2Fwebm&itag=43&mn=sn-ab02a0nfpgxapox-jwwk&requiressl=yes', 'https://r7---sn-ab02a0nfpgxapox-jwwk.googlevideo.com/videoplayback?source=youtube&ip=122.36.71.166&pcm2cms=yes&mt=1452950883&mv=m&sparams=dur%2Cid%2Cinitcwndbps%2Cip%2Cipbits%2Citag%2Clmt%2Cmime%2Cmm%2Cmn%2Cms%2Cmv%2Cpcm2cms%2Cpl%2Cratebypass&signature=1055E08024B9567FEEA8D1966138D1D3F30404DA.14D6E01F0A7519E020BBB9FA55A11E0D78A2DE168&id=o-AC8JZmZolSbfyi5nTfM4phO_cdpp6aDgJfMI5Ucxe6n-8&exp=9406994%2C9408211%2C9408377%2C9416126%2C9417299%2C9419451%2C9419817%2C9420452%2C9421931%2C9422596%2C9423291%2C9423454%2C9423662%2C9425737%2C9425750%2C9426409%2C9426725&dur=180.1868&sver=3&upn=aDZJ25nJLi4&lmt=1389902892206616&expire=1452972579&ratebypass=yes&ipbits=0&initcwndbps=2757500&pl=14&key=yt6&mime=video%2Fmp4&itag=18&mn=sn-ab02a0nfpgxapox-jwwk&requiressl=yes', 'https://r7---sn-ab02a0nfpgxapox-jwwk.googlevideo.com/videoplayback?source=youtube&ip=122.36.71.166&pcm2cms=yes&mt=1452950883&mv=m&sparams=dur%2Cid%2Cinitcwndbps%2Cip%2Cipbits%2Citag%2Clmt%2Cmime%2Cmm%2Cmn%2Cms%2Cmv%2Cpcm2cms%2Cpl%2Crequiressl%2Csource%2Cupn%2Cexpire&ms=au&mm=31&signature=6B8A5ECA064D839183D60C7A176200E1304BEACB.B576DF8200B6F3A13DB3AAFFB63BC7FB1477301F&id=o-AC8JZmZolSbfyi5nTfM4phO_cdpp6aDgJfMI5Ucxe6n-8&exp=9406994%2C9408211%2C9408377%2C9416126%2C9417299%2C9419451%2C9419817%2C9420452%2C9421931%2C9422596%2C9423291%2C9423454%2C9423662%2C9425737%2C9425750%2C9426409%2C9426725&dur=180.1868&sver=3&upn=aDZJ25nJLi4&lmt=1389902878146225&expire=1452972579&ipbits=0&initcwndbps=2757500&pl=14&key=yt6&mime=video%2Fx-flv&itag=5&mn=sn-ab02a0nfpgxapox-jwwk&requiressl=yes', 'https://r7---sn-ab02a0nfpgxapox-jwwk.googlevideo.com/videoplayback?source=youtube&ip=122.36.71.166&pcm2cms=yes&mt=1452950883&mv=m&sparams=dur%2Cid%2Cinitcwndbps%2Cip%2Cipbits%2Citag%2Clmt%2Cmime%2Cmm%2Cmn%2Cms%2Cmv%2Cpcm2cms%2Cpl%2Crequiressl%2Csource%2Cupn%2Cexpire&ms=au&mm=31&signature=4EBA73F9A45ABEB7EA3B04295E381BB7849C64E9]

Debug: hello hello hello

Debugger Console

Frames

MainThread

- from_url, api.py:173
- __init__, api.py:66
- <module>, hello.py:2
- execfile, _pydev_execfile.py:18
- run, pydevd.py:1798
- <module>, pydevd.py:2407

Variables

- js_url = {str} 'http
- self = {YouTube
- stream_map = {
- url = {str} 'https:
- video_data = {di
- video_urls = {list} ['https://r7---sn-ab02a0nfpgxapox-jwwk.googlevideo.com/videoplayback?source=youtube&ip=122.36.71.166&pcm2cms=yes&mt=1452950883&mv=m&sparams=dur%2C... View

Watches

No watches

4: Run 5: Debug 6: TODO Python Console 9: Version Control Terminal Event Log

805 chars 1:3 LF: UTF-8 Git: master:

웹 및 Windows 검색

오후 11:26 2016-01-16