

```

In [5]: from selenium.webdriver.support.ui import WebDriverWait
        from selenium.webdriver.support import expected_conditions as EC
        from selenium import webdriver
        from selenium.webdriver.chrome.service import Service
        from selenium.webdriver.common.by import By
        from selenium.webdriver.common.keys import Keys
        import time

        service = Service(executable_path= "chromedriver.exe")
        driver = webdriver.Chrome()
        driver.get("https://www.google.co.kr/imghp?hl=ko&ogbl")

        input_element = driver.find_element(By.CLASS_NAME, "gLfyf")
        input_element.send_keys("sunrise sky orange" + Keys.ENTER)

        #Scroll Down
        elem = driver.find_element(By.TAG_NAME, 'body')

        for i in range(100):
            elem.send_keys(Keys.PAGE_DOWN)
            time.sleep(0.1)

        #View More

        try:
            view_more_button = WebDriverWait(driver, 10).until(EC.element_to_be_clickable((
            view_more_button.click()
            for i in range(140):
                elem.send_keys(Keys.PAGE_DOWN)
                time.sleep(0.1)

        except:
            pass

```

```

In [6]: # 이미지 다운로드 하기
        images = driver.find_elements(By.CSS_SELECTOR, ".YQ4gaf")
        links = [image.get_attribute('src') for image in images if image.get_attribute('src')
        print('찾은 이미지의 개수: ', len(links))

        # 이미지 경로 설정하기

        import urllib.request

        for k, i in enumerate(links):
            url = i
            urllib.request.urlretrieve(url, r'C:\AutoCoder\sun' + str(k) + '.jpg')

        print('다운로드 완료')

```

찾은 이미지의 개수: 1875

```

-----
HTTPError                                Traceback (most recent call last)
Cell In[6], line 13
    11 for k, i in enumerate(links):
    12     url = i
--> 13     urllib.request.urlretrieve(url, r'C:\AutoCoder\sun' + str(k) + '.jpg')
    15 print('다운로드 완료')

File ~\anaconda3\Lib\urllib\request.py:240, in urlretrieve(url, filename, reporthook, data)
    223 """
    224 Retrieve a URL into a temporary location on disk.
    225
    (...)
    236 data file as well as the resulting HTTPMessage object.
    237 """
    238 url_type, path = _splittype(url)
--> 240 with contextlib.closing(urlopen(url, data)) as fp:
    241     headers = fp.info()
    243     # Just return the local path and the "headers" for file://
    244     # URLs. No sense in performing a copy unless requested.

File ~\anaconda3\Lib\urllib\request.py:215, in urlopen(url, data, timeout, cafile, capath, cadefault, context)
    213 else:
    214     opener = _opener
--> 215 return opener.open(url, data, timeout)

File ~\anaconda3\Lib\urllib\request.py:521, in OpenerDirector.open(self, fullurl, data, timeout)
    519 for processor in self.process_response.get(protocol, []):
    520     meth = getattr(processor, meth_name)
--> 521     response = meth(req, response)
    523 return response

File ~\anaconda3\Lib\urllib\request.py:630, in HTTPErrorProcessor.http_response(self, request, response)
    627 # According to RFC 2616, "2xx" code indicates that the client's
    628 # request was successfully received, understood, and accepted.
    629 if not (200 <= code < 300):
--> 630     response = self.parent.error(
    631         'http', request, response, code, msg, hdrs)
    633 return response

File ~\anaconda3\Lib\urllib\request.py:559, in OpenerDirector.error(self, proto, *args)
    557 if http_err:
    558     args = (dict, 'default', 'http_error_default') + orig_args
--> 559     return self._call_chain(*args)

File ~\anaconda3\Lib\urllib\request.py:492, in OpenerDirector._call_chain(self, chain, kind, meth_name, *args)
    490 for handler in handlers:
    491     func = getattr(handler, meth_name)
--> 492     result = func(*args)
    493     if result is not None:

```

```
494         return result
```

```
File ~\anaconda3\Lib\urllib\request.py:639, in HTTPDefaultErrorHandler.http_error_de  
fault(self, req, fp, code, msg, hdrs)
```

```
638 def http_error_default(self, req, fp, code, msg, hdrs):  
--> 639     raise HTTPError(req.full_url, code, msg, hdrs, fp)
```

```
HTTPError: HTTP Error 404: Not Found
```

In [ ]:

In [ ]: