

Watson Language Translator API

IBM API 사용 가이드부터 읽고 와주세요!

지원 언어 확인하기 -> 이제 지원되지 않습니다.

```
import json
from ibm_watson import LanguageTranslatorV3
from ibm_cloud_sdk_core.authenticators import IAMAuthenticator

authenticator = IAMAuthenticator('발급받은 키 입력')
language_translator = LanguageTranslatorV3(
    version='2018-05-01',
    authenticator=authenticator
)

language_translator.set_service_url('발급받은 url 입력')

languages = language_translator.list_languages().get_result() #지원 언어 리스트 확인 메소드
print(json.dumps(languages, indent=2))
```

결과

AttributeError: 'LanguageTranslatorV3' object has no attribute 'list_languages'

원래는 리스트가 떠야하는데 에러가 발생합니다.

번역하기!

```
import json
from ibm_watson import LanguageTranslatorV3
from ibm_cloud_sdk_core.authenticators import IAMAuthenticator
authenticator = IAMAuthenticator('발급받은 API 키 입력')
language_translator = LanguageTranslatorV3(
    version='2018-05-01',
    authenticator=authenticator
)
language_translator.set_service_url('발급받은 URL 키 입력')
```

.....

인증을 한 후 다음 슬라이드의 방식 중 하나를 통해 번역이 가능합니다. 즉, `model_id`는 `source_target`를 의미하며, `model_id`가 존재할 경우 `source`와 `target`은 `model_id`에 의해 `override`됩니다. 즉, `target`을 `es`로 해두어도 `model_id`가 `en_kr`이면 한국어로 번역됩니다.

※ 다음 슬라이드 코드 중 하나를 에 넣어 주시면 됩니다.

번역하기!

코드	<pre>translation = language_translator.translate(text='Hello, how are you today?', source='en', target='es').get_result() print(json.dumps(translation, indent=2, ensure_ascii=False))</pre>	<pre>translation = language_translator.translate(text='Hello, how are you today?', model_id='en-ko').get_result() print(json.dumps(translation, indent=2, ensure_ascii=False))</pre>	<pre>translation = language_translator.translate(text='Hello, how are you today?', model_id='en-ko', source='en', target='es'.get_result() print(json.dumps(translation, indent=2, ensure_ascii=False))</pre>
결과	<pre>{ "translations": [{ "translation": "Hola, ¿cómo estás hoy?" }], "word_count": 7, "character_count": 25 }</pre>	<pre>{ "translations": [{ "translation": "안녕하세요, 오늘은 어떠세요?" }], "word_count": 7, "character_count": 25 }</pre>	

번역하기!

당연히 한국어 -> 영어로의 번역도 가능합니다.

```
import json
from ibm_watson import LanguageTranslatorV3
from ibm_cloud_sdk_core.authenticators import IAMAuthenticator

authenticator = IAMAuthenticator('발급받은 API키 입력')
language_translator = LanguageTranslatorV3(
    version='2018-05-01',
    authenticator=authenticator
)

language_translator.set_service_url('발급받은 URL 입력')

translation = language_translator.translate(
    text='안녕하세요. 오늘은 어떠세요?',
    model_id='ko-en').get_result()
print(json.dumps(translation, indent=2, ensure_ascii=False))
```

를 수행시 우측의 결과를 얻을 수 있습니다.

```
{
  "translations": [
    {
      "translation": "Hello. How are you feeling today?"
    }
  ],
  "word_count": 9,
  "character_count": 16
}
```

Cusomization은 무료 버전에서 제공하지 않습니다..

원래 `model`을 만들어서 특정 문장은 번역하지 않도록 하는 것이 가능한데
무료버전에서는 제공되지 않습니다.

Error: The given plan does not permit customization requests., Code: 403 , X-global-transaction-id:
3c9df81a22586e3ef813f34093498500

<https://cloud.ibm.com/docs/language-translator?topic=language-translator-customizing#create-your-training-data> 와

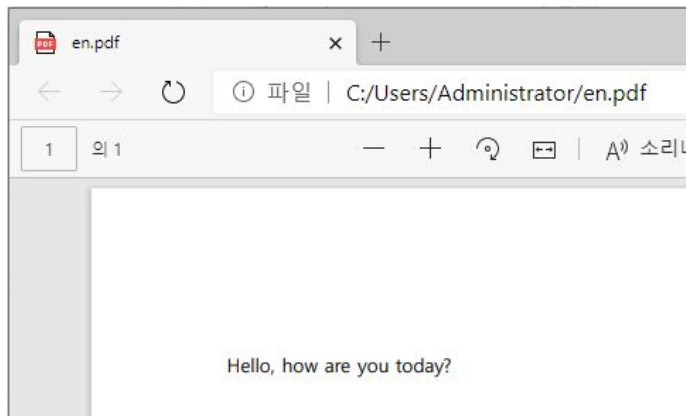
<https://cloud.ibm.com/apidocs/language-translator?code=python#list-models> 를 참고해 주세요.

문서 통채로 번역하기!

순서

1. 번역하려는 문서를 입력
2. 처리 후 리스트로 보관
3. 리스트 중 **document id**를 통해서 번역된 내용을 파일로 받음.
4. 리스트에 보관중인 번역 내용 삭제

문서 통채로 번역하기! - 1. 번역하려는 문서 입력



#아래의 코드는 슬라이드4에 해당하는 부분입니다.

with open('en.pdf', 'rb') as file: #번역하려는 파일을 엽니다. 20mb까지 가능, 절대경로 가능

result = language_translator.translate_document(

file=file,

file_content_type='application/pdf',

#번역할 파일의 종류, 가능한 목록은 다음 슬라이드에

filename='en.pdf',

번역하려는 파일 이름

model_id='en-ko'

source / target으로도 가능

document_id = '{~}' 이전에 번역한거 재활용 하려면 이걸로 하면 됩니다.

).get_result()

print(json.dumps(result, indent=2))

사용 가능한 확장자는 아래와 같습니다.

Allowable values: [application/powerpoint , application/mspowerpoint , application/x-rtf , application/json , application/xml , application/vnd.ms-excel , application/vnd.openxmlformats-officedocument.spreadsheetml.sheet , application/vnd.ms-powerpoint , application/vnd.openxmlformats-officedocument.presentationml.presentation , application/msword , application/vnd.openxmlformats-officedocument.wordprocessingml.document , application/vnd.oasis.opendocument.spreadsheet , application/vnd.oasis.opendocument.presentation , application/vnd.oasis.opendocument.text , application/pdf , application/rtf , text/html , text/json , text/plain , text/richtext , text/rtf , text/xml]

문서 통채로 번역하기! - 2. 번역결과를 리스트로 보관

코드 돌린 결과 입니다. 각 항목의 설명은 우측과 같습니다.

```
{  
  "document_id": "454e8b1e-41bf-4841-b3f4-86df000eb9da",  
  "filename": "en.pdf",  
  "model_id": "en-ko",  
  "source": "en",  
  "target": "ko",  
  "status": "processing",  
  "created": "2020-08-14T08:56:43Z"  
}
```

document_id : 번역된 내용을 가리키는 식별자

filename : 번역한 파일 이름

status : 처리상태 `processing`, `available`, `failed` 로 나뉘어짐

`available` - 번역이 완료되어 추출이 가능한 상태

model_id : 번역에 사용된 모델 아이디

base_mode_id : 이건 커스텀 모델일 경우에만 나옴

source / target : 번역한 언어, 번역된 언어

created : 만들어진 시간

completed : 번역이 완료된 시간

word_count : `available` 상태일때만 나옴, 총 단어 수

character_count : 동일, 총 문자 수

문서 통채로 번역하기! - 2. 번역결과를 리스트로 보관(계속)

아래 명령어를 통해 지금까지 번역된 내용들을 저장하는 리스트를 확인할 수도 있습니다.

```
result = language_translator.list_documents().get_result()
print(json.dumps(result, indent=2))
```

해당 명령어의 결과는 우측과 같습니다.

```
{
  "documents": [
    {
      "document_id":
"454e8b1e-41bf-4841-b3f4-86df000eb9da",
      "filename": "en.pdf",
      "model_id": "en-ko",
      "source": "en",
      "target": "ko",
      "status": "available",
      "created": "2020-08-14T08:56:43Z",
      "completed": "2020-08-14T08:56:46Z",
      "word_count": 5,
      "character_count": 25
    }
  ]
}
```

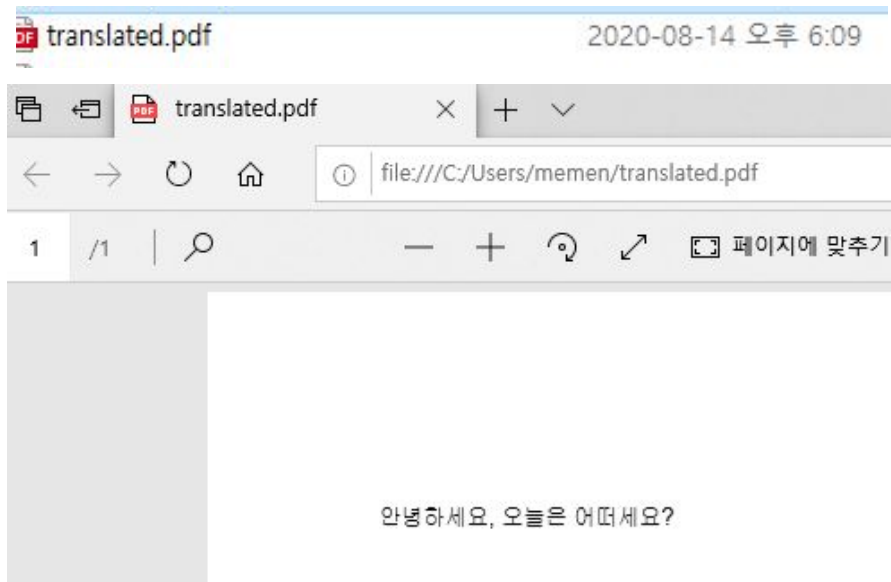
문서 통채로 번역하기! - 3. 번역된 내용을 문서로 내려받기

리스트에 있는 번역 문서 내용을 document id를 통해서 번역된 내용을 파일로 받을 수 있습니다..

```
with open('translated.pdf', 'wb') as f:    #번역본 문서 이름, 절대경로 가능합니다.
    result = language_translator.get_translated_document(
        document_id='454e8b1e-41bf-4841-b3f4-86df000eb9da', #리스트에 저장된 번역된 내용을 가리키는 id
        accept='application/pdf').get_result() #사용가능한 확장자 종류는 다음 슬라이드에 있습니다.
    f.write(result.content)
```

해당 코드를 수행하여
문서 id를 통해 번역 내용을 문서로 내려 받을 수 있습니다.

우측 그림은 해당 코드를 통해 생성된 문서입니다.



문서 통채로 번역하기! - 3. 번역된 내용을 문서로 내려받기(계속)

#번역된 내용은 다양한 파일 형식으로 다운 받을 수 있습니다.

with open('translated.pdf', 'wb') as f: #번역본 문서 이름, 절대경로 가능합니다.

.....
accept='application/pdf').get_result() #사용가능한 확장자 종류는 아래와 같습니다.

f.write(result.content)

단, 처음 번역을 요청한 파일의 형식과 같은 확장자만 가능합니다

이번 예제의 경우, en.pdf를 열어 번역한 후 파일로 저장하는 것이기에 application/pdf 형식으로만 다운 받을 수 있습니다.

만약 다른 형식으로 파일 생성을 시도할 경우 아래와 같은 오류가 발생합니다.

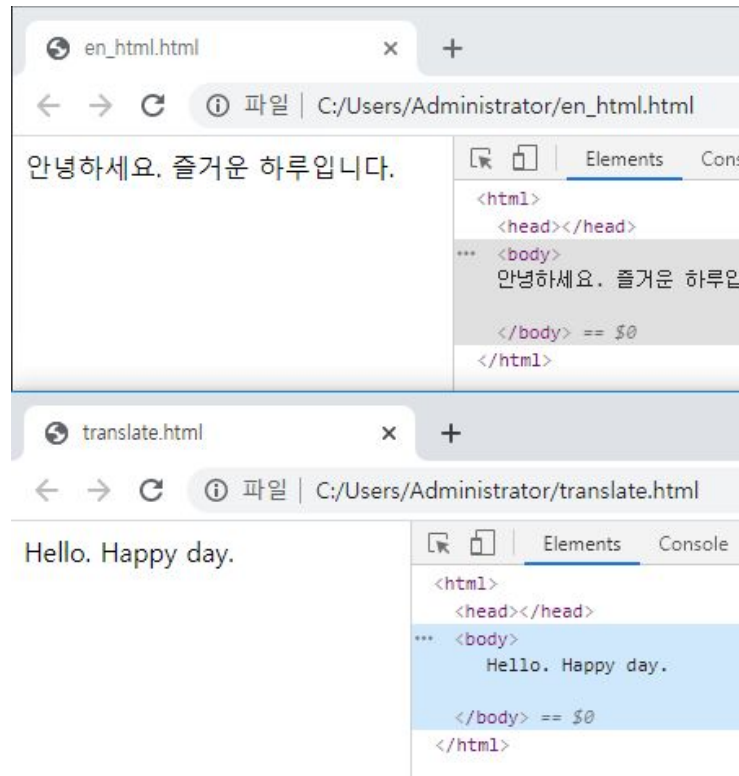
ApiException: Error: Document found but accept header prevents delivery., Code: 406
, X-global-transaction-id: b838400735a45c5cfc8490073db7960d

Allowable values: [application/powerpoint, application/mspowerpoint, application/x-rtf, application/json, application/xml, application/vnd.ms-excel, application/vnd.openxmlformats-officedocument.spreadsheetml.sheet, application/vnd.ms-powerpoint, application/vnd.openxmlformats-officedocument.presentationml.presentation, application/msword, application/vnd.openxmlformats-officedocument.wordprocessingml.document, application/vnd.oasis.opendocument.spreadsheet, application/vnd.oasis.opendocument.presentation, application/vnd.oasis.opendocument.text, application/pdf, application/rtf, text/html, text/json, text/plain, text/richtext, text/rtf, text/xml]

문서 통채로 번역하기! - 3. 번역된 내용을 문서로 내려받기(계속)

그럼, 이번엔 **html** 파일을 열어 번역한 후 다시 다른 **html**파일로 저장해보겠습니다.

```
.....  
with open('en_html.html', 'rb') as file:          #번역하려는 파일을 엽니다.  
    result = language_translator.translate_document(  
        file=file,  
        file_content_type='text/html',  
        filename='en_html.html',  
        model_id='ko-en').get_result()  
    print(json.dumps(result, indent=2))  
  
.....  
with open('translate.html', 'wb') as f:  
    result = language_translator.get_translated_document(  
        document_id='f9bf4e1f-9e74-4f1e-a557-23733619057a',  
        accept='text/html').get_result()  
    f.write(result.content)
```



문서 통채로 번역하기! - 3. 번역된 내용을 문서로 내려받기(계속)

마찬가지로 html문서를 번역한 것을 pdf로 받으려고 할 경우에도 에러가 발생합니다.

```
with open('translate.pdf', 'wb') as f:
    result = language_translator.get_translated_document(
        document_id='f9bf4e1f-9e74-4f1e-a557-23733619057a',
        accept='application/pdf').get_result()
    f.write(result.content)
```

```
c:\users\administrator\appdata\local\programs\python\python38-32\lib\site-
packages\ibm_cloud_sdk_core\base_service.py in send(self, request, **kwar
s)
    226             error_message = 'Unauthorized: Access is denied due to
    227                                     'invalid credentials'
--> 228         raise ApiException(
    229             response.status_code, error_message, http_response=response)
    230     except requests.exceptions.SSLError:
```

ApiException: Error: Document found but accept header prevents delivery., Code: 406
, X-global-transaction-id: b3b8a8f57573c897b7e45e1658b3a4b6

문서 통채로 번역하기! - 4. 리스트 보관 내용 삭제

```
language_translator.delete_document(document_id='454e8b1e-41bf-4841-b3f4-86df000eb9da')
```

처럼 document_id로 해당 내용을 삭제 할 수 있습니다.

응답 결과는 아래와 같습니다.

```
<ibm_cloud_sdk_core.detailed_response.DetailedResponse at 0x72dc808>
```