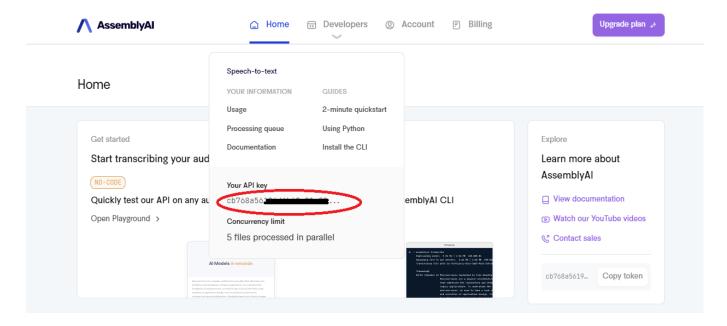
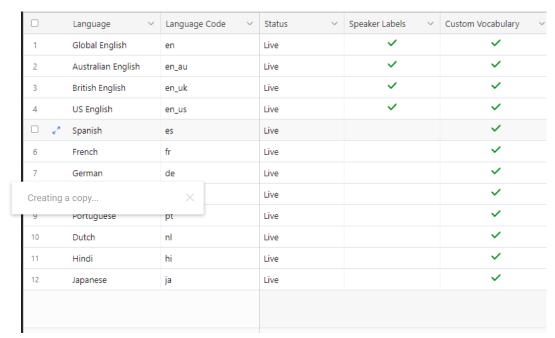
▼ AssemblyAi





```
with open('assemblyai.txt') as f:
    API_KEY = f.read()

import requests

# upload
filename = 'f2bjrop1.0.wav'
def read_file(filename,chunk_size=5242880):
    with open(filename,'rb') as f:
    while True:
    data = f.read(chunk_size)
    if not data:
        break
    yield data
```

```
neaders = { authorization :API_KEY}
response = requests.post('https://api.assemblyai.com/v2/upload',headers = headers,data=read_file(filename))
response
     <Response [200]>
print(response.json())
     {'upload_url': 'https://cdn.assemblyai.com/upload/dc099cfa-0498-449a-ad9b-d7a619e547c6'}
#transcript
endpoint = "https://api.assemblyai.com/v2/transcript"
json = {'audio_url':response.json()['upload_url']}
headers = {'authorization':API_KEY, 'content_type':'application/json'}
response = requests.post(endpoint,headers = headers,json=json)
response
     <Response [200]>
response.json()
#poll
polling_endpoint = endpoint+'/'+str(response.json()['id'])
polling_response = requests.get(polling_endpoint,headers=headers)
print(polling_response.json())
     {'id': '66a91shr1g-522f-45de-812f-6cc7574e3c7b', 'language_model': 'assemblyai_default', 'acoustic_model': 'assemblyai_default', 'langua
print(polling_response.json()['text'])
     Wanted chief justice of the Massachusetts Supreme Court. In April, the SJC's current leader, Edward Hennessy, reaches the mandatory reti
    4
while True:
  polling_response = requests.get(polling_endpoint,headers=headers)
  if polling_response.json()['status']=='completed':
    nrint(nolling resnonse ison()['text'])
 Creating a copy...
                                    us']=='error':
    print('error')
   break
  else:
    pass
     Wanted chief justice of the Massachusetts Supreme Court. In April, the SJC's current leader, Edward Hennessy, reaches the mandatory reti
    4
```

• ×

Creating a copy... ×