

Google Text to Speech API

0. 사용 설정

- 1) TTS API 사용을 누릅니다.

Cloud Text-to-Speech API를 사용 설정합니다

API 사용 설정

<https://console.cloud.google.com/flows/enableapi?apiid=texttospeech.googleapis.com&hl=ko>

- 2) 다른 API도 사용한 적이 없다면, 서비스 계정키가 없을 것입니다. 서비스 계정키가 없다면 만들어 주세요.
(다른 API를 이용한 적이 있다면, 그때 만든 계정키를 쓰면 됩니다.)

<https://console.cloud.google.com/apis/credentials/serviceaccountkey?hl=ko>

- 3) power shell을들어가서

\$env:GOOGLE_APPLICATION_CREDENTIALS="**[PATH]**"

을 입력해 줍니다. [PATH]대신 2)에서 다운받은 키 json 파일의 경로를 입력해 주면 됩니다

ex) \$env:GOOGLE_APPLICATION_CREDENTIALS="**C:\Users\memen\google_api_key.json**"

가이드에 앞서

- 간단하게 사용하고 싶으신 분들은 간단 설명 슬라이드 4~6만 참조해 주시면 됩니다.
- 이해가 안되거나 자세한 설명이 필요하신 분들은 슬라이드 8 ~ 14 를 참고해주세요.

간편 가이드

1. 텍스트에서 오디오 합성

```
tts_request.json ×
스키마: <선택된 스키마가 없음>

1  {
2    "input": {
3      "text": "Hello-World"
4    },
5    "voice": {
6      "languageCode": "en-gb",
7      "name": "en-GB-Standard-A",
8      "ssmlGender": "FEMALE"
9    },
10   "audioConfig": {
11     "audioEncoding": "MP3"
12   }
13 }
```

합성 언어가 영어라면 여기 내용만 바꿔주세요. 추후 powershell 사용을 쉽게 하시려면 해당 json 파일을 c /사용자/해당사용자 에 저장해 주세요.

2. 요청 전송

아래의 텍스트들을 복사해서 powershell에 붙여주세요.

```
$env:GOOGLE_APPLICATION_CREDENTIALS="C:/Users/memen/google_api_key.json"
$cred = gcloud auth application-default print-access-token
$headers = @{ "Authorization" = "Bearer $cred" }
Invoke-WebRequest `
  -Method POST `
  -Headers $headers `
  -ContentType: "application/json; charset=utf-8"
  -InFile tts_request.json `
  -Uri "https://texttospeech.googleapis.com/v1/text:synthesize" | Select-Object -Expand Content
```

만약 에러가 뜬다면 슬라이드 10 상세가이드 2-1 에러 확인을 참고해주세요.
이상 없이 뭔가 쭉 긴 텍스트들이 powershell에 출력되었다면 다음 슬라이드로 가주세요.

3. 결과 확인 및 처리

정상적으로 작동되었다면 우측의 긴 텍스트들이 반환됩니다.

```
{
  "audioContent" : "~"
}
```

에서 ~에 해당하는 부분(빨간색으로 테두리 쳐진 부분)만을 복사합니다. 큰따옴표는 포함되지 않습니다.

구글 등 검색 브라우저에 base64 to mp3를 검색하여 나오는 사이트 중 하나를 들어가주세요.

예) <https://www.ipvoid.com/base64-to-mp3/>

Enter Base64 String Here

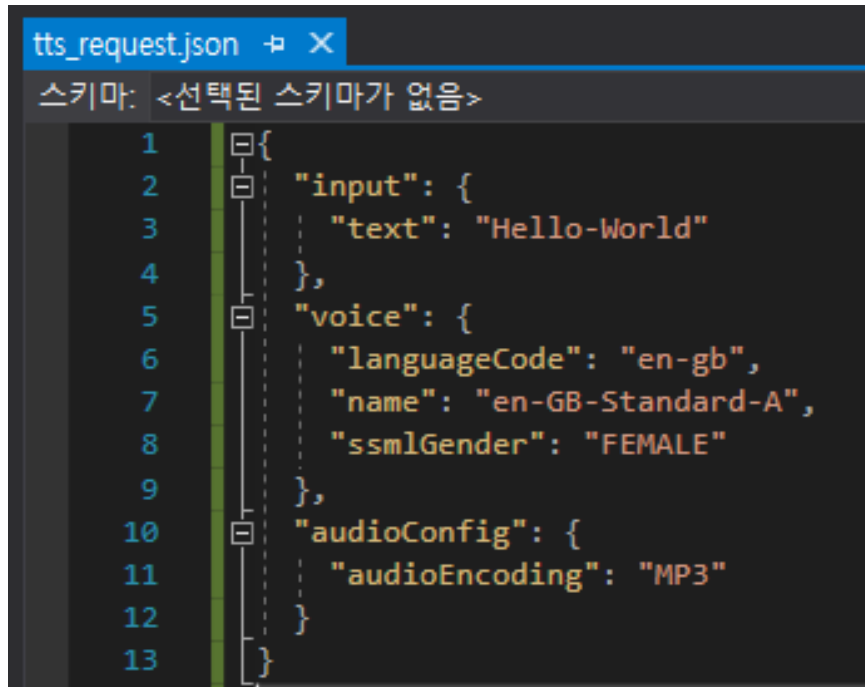
Submit Now

이제 이곳에 복사한 내용을 붙여놓고 버튼을 누르면
mp3 파일이 나옵니다.

[illegible]

상세 가이드

1. 텍스트에서 오디오 합성



The screenshot shows a code editor with a file named 'tts_request.json'. The JSON content is as follows:

```
1 {
2   "input": {
3     "text": "Hello-World"
4   },
5   "voice": {
6     "languageCode": "en-gb",
7     "name": "en-GB-Standard-A",
8     "ssmlGender": "FEMALE"
9   },
10  "audioConfig": {
11    "audioEncoding": "MP3"
12  }
13 }
```

- POST 방식에 request.json을 바디로 하여 보낼 것입니다.
- tts_request.json을 하나 만들어줍니다.
- 1) input :
 - text : 음성으로 듣기를 원하는 문장을 작성해 주시면 됩니다.
- 2) voice :
 - languageCode : 사용하려는 음성의 언어입니다.
 - name : 사용하는 음성의 고유 이름입니다.
 - ssmlGender : 남성 / 여성 목소리임을 설정합니다.
 - 위 항목들은 보이스 리스트를 확인하고 사용해야 합니다.
보이스 리스트는 아래 링크를 통해 확인 가능합니다.
- <https://cloud.google.com/text-to-speech/docs/voices?hl=ko>
- audioConfig : 합성 음성을 어떠한 확장자로 받을것인지 명시합니다.

2. 요청 전송

- power shell을 이용해 서버로 요청을 보냅니다.
- \$cred를 이용해 앞서 등록한 키로 접근 할 수 있는지 확인합니다.

```
PS C:\Users\memen> $cred = gcloud auth application-default print-access-token
```

- \$headers 를 이용해 header에 'Authorization {cred정보}'를 담고있는 변수를 만듭니다..

```
PS C:\Users\memen> $headers = @{ "Authorization" = "Bearer $cred" }
```

- Invoke-WebRequest로 요청을 보냅니다.

- Post방식
- header 내용
- ContentType : application/json
- InFile : Body에 해당하는 내용을 담고있는 파일의 경로.
띄워쓰기가 있는 부분은 " "로 둘러주시면 됩니다.
- Uri : 요청을 보내는 URI

```
PS C:\Users\memen> Invoke-WebRequest `
>> -Method POST `
>> -Headers $headers `
>> -ContentType: "application/json; charset=utf-8" `
>> -InFile C:\Users\memen\OneDrive\바탕 화면\인턴 - 무지개연구소\구글TTS\ tts_request.json `
>> -Uri "https://texttospeech.googleapis.com/v1/text:synthesize" | Select-Object -Expand Content
```

2-1. 에러 확인

아무런 메시지가 뜨지 않았다면 다음 슬라이드로 넘어가 주시고, 에러가 떴다면 아래를 참고해주세요.

```
Invoke-WebRequest : 원격 서버에서 (403) 사용할 수 없음 오류를 반환했습니다.
PS C:\Users\memen> $cred = gcloud auth application-default print-access-token
ERROR: (gcloud.auth.application-default.print-access-token) File C:/Users/memen/wrong_google_api_key.json was not found.
```

이러한 에러시에는 인증키 (GOOGLE_APPLICATION_CREDENTIALS)의 경로와 파일명을 확인해 주세요.

```
Invoke-WebRequest : 'C:\Users\memen\OneDrive\바탕 화면\인턴 - 무지개연구소\구글\TTS\wrong_tts_request.json' 경로는 존재하지 않으므로 찾을 수 없습니다.
```

이러한 에러시에는 요청 json파일의 경로를 다시 한 번 확인해 주세요.

```
Invoke-WebRequest : 원격 서버에서 (400) 잘못된 요청 오류를 반환했습니다.
```

이러한 에러시에는 요청 json파일 내용을 확인해 주세요. 주로 괄호가 잘못 닫히거나, 필요 파라미터 부족 시 이 에러가 발생합니다.

3. 결과 확인

- Base64로 인코딩된 mp3파일을 받게됩니다.

- 이제 이 Base64를 다시 mp3파일로 디코딩(변환)이 필요합니다.



- 먼저 AudioContent의 내용에 해당하는 //NExAASoolIAB ~ 를 복사합니다.



4. 변환

- 이제 Base64를 Mp3로 디코딩이 필요합니다.
- 저는 변환을 우측의 자바 코드로 변환 하였습니다
- 간편히 하고싶은 경우 구글에서 base64 to mp3를 검색해 사이트에서도 변환이 가능합니다.
- 자바 work-space에 성공적으로 변환된 mp3 파일이 있습니다..

```
Decoder.java
1 import java.io.File;
2 import java.io.FileOutputStream;
3 import java.util.Base64;
4
5 public class Decoder {
6     public static void main(String[] args)
7     {
8         byte[] decodedBytes = Base64.getDecoder().decode("//NExAASooIIABhEuTQIAJQM6h07vEI1zGPACQAdRuY8A");
9
10        try
11        {
12            File file = new File("./hello_world.mp3");
13            FileOutputStream fos = new FileOutputStream(file, true);
14            fos.write(decodedBytes);
15            fos.close();
16        }
17        catch (Exception e)
18        {
19
20        }
21    }
22 }
23 }
```

PC > 로컬 디스크 (C:) > 사용자 > memen > eclipse-workspace > Base64 Decoder

이름	수정한 날짜	유형	크기
.settings	2020-08-20 오전 10:21	파일 폴더	
bin	2020-08-20 오전 10:22	파일 폴더	
src	2020-08-20 오전 10:22	파일 폴더	
.classpath	2020-08-20 오전 10:21	CLASSPATH 파일	1KB
.project	2020-08-20 오전 10:21	PROJECT 파일	1KB
hello_world.mp3	2020-08-20 오전 10:27	MP3 파일	5KB



- 한국어는 파일은 정상적으로 만들어지는데 반해
- 음성이 "안녕하세요. ~"가 아닌 "%ysds ~"이런식 으로 코드로 나옴...

```

tts_request2.json
스키마: <선택된 스키마가 없음>
1  {
2    "input": {
3      "text": "안녕하세요. 좋은 하루입니다."
4    },
5    "voice": {
6      "languageCode": "ko-KR",
7      "name": "ko-KR-Standard-A",
8      "ssmlGender": "FEMALE"
9    },
10   "audioConfig": {
11     "audioEncoding": "MP3"
12   }
13 }

```

```

PS C:\Users\memen> Invoke-WebRequest `
>> -Method POST `
>> -Headers $headers `
>> -ContentType: "application/json; charset=utf-8" `
>> -InFile C:\Users\memen\OneDrive\바탕 화면\무지개연구소\구글TTS\tts_request2.json `
>> -Uri "https://texttospeech.googleapis.com/v1/text:synthesize" | Select-Object -Expand Content
{
  "audioContent": "//NExAAREEIMAAhGJGAAEY2YI3cWZwEAQGg/BBQIYnPieCByz6jm9YPygIRP0INwPw+UDEoNeUDEP4Jw/yZQ4YPrB8/U4u8HxwYE5zmsH8k1W/Vc6QwgYsULdE11VTM//NExA4RCHYcAUMYABM4VF3gapDBQqGm0e6nqSPsrEoa8RYhcmKVi5UYq1Aoo0y"
}

```

```

안녕하세요.txt - Windows 메모장
파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)
//NExAAREEIMAAhGJGAAEY2YI3cWZwEAQGg/BBQIYnPieCByz6jm9YPygIRP0INwPw+UDEoNeUDEP4Jw/yZQ4YPrB8/U4u8HxwYE5zmsH8k1W/Vc6QwgYsULdE11VTM//NExA4RCHYcAUMYABM4VF3gapDBQqGm0e6nqSPsrEoa8RYhcmKVi5UYq1Aoo0y

```

```

Decoder.java
1 import java.io.File;
2 import java.io.FileOutputStream;
3 import java.util.Base64;
4
5 public class Decoder {
6     public static void main(String[] args)
7     {
8         byte[] decodedBytes = Base64.getDecoder().decode("//NExAASooIIABhEuTQIAJQM6h07vEI1zGPACQAdRuY8AA");
9
10        try
11        {
12            File file = new File("./안녕하세요.mp3");
13            FileOutputStream fos = new FileOutputStream(file, true);
14            fos.write(decodedBytes);
15            fos.close();
16        }
17        catch (Exception e)
18        {
19        }
20    }
21 }

```

로컬 디스크 (C:) > 사용자 > memen > eclipse-workspace > Base64 Decoder

이름	수정된 날짜	유형	크기
.settings	2020-08-20 오전 10:21	파일 폴더	
bin	2020-08-20 오전 10:22	파일 폴더	
src	2020-08-20 오전 10:22	파일 폴더	
.classpath	2020-08-20 오전 10:21	CLASSPATH 파일	1KB
.project	2020-08-20 오전 10:21	PROJECT 파일	1KB
hello_world.mp3	2020-08-20 오전 10:27	MP3 파일	5KB
안녕하세요.mp3	2020-08-20 오전 10:34	MP3 파일	5KB