

Assignment REST API Topics

- Get different types of response from API endpoint (CRUD)
- Check for Response Code Eg (404, 403, 200)
- Visualize response (JSON/XML)
- Difference Between api versions
- Store Useful Data in tabular form (dataframe)

For this assignment I will be using quran.com api to extract data and store it in Dataframe

▼ Get different types of response from API endpoint (CRUD)

requests

Requests is a python library used for making HTTP request and store response either in JSON , XML or binary

Endpoint

In an API, an endpoint is a specific URL or URI that represents a resource or functionality provided by the API. It serves as the target location for accessing or manipulating data through HTTP requests.

<https://api.quran.com/api/{endpoint}>

<https://api.quran.com/api/v3/chapters/3>

```
1 import requests
2
3 url = "https://api.quran.com/api/v3/chapters/3"
4
```

▼ Check for Response Code Eg (404, 403, 200)

- response = 404 - Not Found
- response = 403 - Forbidden
- response = 200 - OK

```
1 response = requests.get(url)
2 print("Response Cocde: ",response.status_code)
```

```
Response Cocde: 200
```

▼ Visualize response (JSON/XML)

```

1 response.json()

{'chapter': {'id': 3,
  'revelation_place': 'madinah',
  'revelation_order': 89,
  'bismillah_pre': True,
  'name_simple': "Ali 'Imran",
  'name_complex': 'Āli `Imrān',
  'name_arabic': 'آل عمران',
  'verses_count': 200,
  'pages': [50, 76],
  'chapter_number': 3,
  'translated_name': {'language_name': 'english', 'name': 'Family of Imran'}}}

```

▼ Difference Between api versions

1. <https://api.quran.com/api/v3/chapters/3> (here **v3** means version-3 of this api)
there can be multiple versions of same api
2. For instance **v4** of this api is also available

```

1 print("*****")
2 print("[INFO] Version-3 API response")
3 print("*****\n")
4
5 print(requests.get("https://api.quran.com/api/v3/chapters/3").json())
6
7 print("\n\n*****")
8 print("[INFO] Version-4 API response")
9 print("*****\n")
10
11 print(requests.get("https://api.quran.com/api/v4/chapters/3").json())

*****
[INFO] Version-3 API response
*****

{'chapter': {'id': 3, 'revelation_place': 'madinah', 'revelation_order': 89, 'bismillah_pre': True, 'name_simple': "Ali 'Imran", 'name_complex': 'Āli `Imrān', 'name_arabic': 'آل عمران', 'verses_count': 200, 'pages': [50, 76], 'chapter_number': 3, 'translated_name': {'language_name': 'english', 'name': 'Family of Imran'}}}

*****
[INFO] Version-4 API response
*****

{'chapter': {'id': 3, 'revelation_place': 'madinah', 'revelation_order': 89, 'bismillah_pre': True, 'name_simple': "Ali 'Imran", 'name_complex': 'Āli `Imrān', 'name_arabic': 'آل عمران', 'verses_count': 200, 'pages': [50, 76], 'chapter_number': 3, 'translated_name': {'language_name': 'english', 'name': 'Family of Imran'}}}

```

▼ Store Useful Data in tabular form (dataframe)

```

1 import requests
2
3 # {'id': 3,
4 #   'revelation_place': 'madinah',
5 #   'revelation_order': 89,
6 #   'bismillah_pre': True,
7 #   'name_simple': "Ali 'Imran",

```

```

8 # 'name_complex': 'Āli `Imrān',
9 # 'name_arabic': 'آل عمران',
10 # 'verses_count': 200,
11 # 'pages': [50, 76],
12 # 'chapter_number': 3,
13 # 'translated_name': {'language_name': 'english', 'name': 'Family of Imran'}}
14
15
16 def get_data() :
17     try:
18
19         extracted_data = {}
20         for i in range(1, 115):
21             url = f"https://api.quran.com/api/v3/chapters/{i}"
22             response = requests.get(url)
23             chapter_info = response.json()
24
25             extracted_data[i] = {
26                 "id": chapter_info["chapter"]["id"],
27                 "name_simple": chapter_info["chapter"]["name_simple"],
28                 "revelation_city": chapter_info["chapter"]["revelation_place"],
29                 "verses_count": chapter_info["chapter"]["verses_count"],
30                 "translated_name": chapter_info["chapter"]["translated_name"]["name"],
31             }
32
33         return extracted_data
34
35     except requests.exceptions.HTTPError as e:
36         print(e.respose.text)
37
38
39 raw_data = get_data()

```

▼ Making Dataframe pandas with specific columns

```

1 import pandas as pd
2
3 df = pd.DataFrame.from_dict(raw_data, orient='index', columns=['id', 'name_simple', 'r

1 df.head()

```

	id	name_simple	revelation_city	verses_count	translated_name
1	1	Al-Fatihah	makkah	7	The Opener
2	2	Al-Baqarah	madinah	286	The Cow
3	3	Ali 'Imran	madinah	200	Family of Imran
4	4	An-Nisa	madinah	176	The Women
5	5	Al-Ma'idah	madinah	120	The Table Spread

1

Colab paid products - Cancel contracts here