

API Shortener

Alumno: Víctor Manuel Ramírez Reyes

Codigo

```
1  '''
2  * Nombre: Víctor Manuel Ramírez Reyes
3  * Fecha: 20/11/2022
4  * API: URL Shortener
5  * Función: Crea una URL corta como alias de la larga. La URL corta será redirigida a la original
6  '''
7
8  import requests
9
10 #se crea la función que acepta la clave API, la URL y un alias opcional como parametro
11 def acortar_url(api_key, url, alias=None):
12     api_url = "https://url-shortener23.p.rapidapi.com/shorten"
13     headers = {
14         "content-type": "application/json",
15         "X-RapidAPI-Key": api_key,
16         "X-RapidAPI-Host": "url-shortener23.p.rapidapi.com"
17     }
18
19     payload = {"url": url}
20     if alias:
21         payload["alias"] = alias
22
23     response = requests.post(api_url, json=payload, headers=headers)
24
25     return response.json()
26
```

```
27 #la funcion main solicita al usuario la URL que desea recortar y un alias opcional
28 def main():
29     api_key = "d907b7d440msh51e2842354fafa6p10e270jsn5d94ef792f5f"
30
31     while True:
32         url_para_recortar = input("Ingrese la URL que desea recortar (si desea salir escriba \"fin\"): ")
33
34         # Verificar si el usuario desea salir del bucle
35         if url_para_recortar.lower() == 'fin':
36             break
37         alias = input("Ingrese un alias personalizado (opcional): ")
38
39         #se llama a la función shorten_url con la entrada del usuario
40         result = acortar_url(api_key, url_para_recortar, alias)
41         #se muestra el resultado
42         print(result)
43
44 #verifica si el script esta siendo ejecutado directamente
45 if __name__ == "__main__":
46     #se ejecuta la función main
47     main()
```

Evidencia de la función de la API

```
• vreyes@vreyes:~/Documentos/Recuperacion1$ /bin/python3 /home/vreyes/Documentos/Recuperacion1/shortURL.py
Ingrese la URL que desea recortar (si desea salir escriba "fin"): https://www.youtube.com/results?search_query=request+api+python
Ingrese un alias personalizado (opcional):
{'short_url': 'https://gbits.me/lpGatSJE'}
Ingrese la URL que desea recortar (si desea salir escriba "fin"): https://github.com/vicvapurub/programacion-redes-2023/tree/main/Unidad2/unidad2
Ingrese un alias personalizado (opcional): mirepositorio
{'short_url': 'https://gbits.me/mirepositorio'}
Ingrese la URL que desea recortar (si desea salir escriba "fin"): fin
• vreyes@vreyes:~/Documentos/Recuperacion1$
```

API Covid-19 Tracking

Función: Esta API Freemium rastrea información real sobre el coronavirus (COVID-19) sobre personas infectadas y sometidas a pruebas en varios países.

Codigo

```
1  """
2      Nombre: Víctor Manuel Ramírez Reyes
3      Fecha: 17/10/2023
4      API: Covid-19-tracking
5      Función: Esta API Freemium rastrea información real sobre el coronavirus (COVID-19) sobre personas infectadas y sometidas a prue
6  """
7
8  import requests
9
10 url = "https://covid-19-tracking.p.rapidapi.com/v1/"
11
12 headers = {
13     "X-RapidAPI-Key": "29a251fa7fmsh5974211lee70d7fp1cfef5jsna94453741981",
14     "X-RapidAPI-Host": "covid-19-tracking.p.rapidapi.com"
15 }
16
17 def obtener_datos_por_pais(api_url, api_headers, pais):
18     endpoint = pais
19     response = requests.get(api_url + endpoint, headers=api_headers)
20     return response
21
```

```
21
22 def imprimir_datos(response):
23     if response.status_code == 200:
24         data = response.json()
25
26         if isinstance(data, list):
27             # Si la respuesta es una lista, asumimos que contiene datos de varios países
28             for pais_info in data:
29                 print("Datos del país:")
30                 print("País: " + pais_info.get("Country_text", "N/A"))
31                 print("Casos Confirmados: " + pais_info.get("Total Cases_text", "N/A"))
32                 print("Muertes: " + pais_info.get("Total Deaths_text", "N/A"))
33                 print("Recuperados: " + pais_info.get("Total Recovered_text", "N/A"))
34                 print("Fecha de Actualización: " + pais_info.get("Last Update", "N/A"))
35                 print("-----")
36         elif isinstance(data, dict):
37             # Si la respuesta es un diccionario, asumimos que contiene datos de un solo país
38             print("Datos del país:")
39             print("País: " + data.get("Country_text", "N/A"))
40             print("Casos Confirmados: " + data.get("Total Cases_text", "N/A"))
41             print("Muertes: " + data.get("Total Deaths_text", "N/A"))
42             print("Recuperados: " + data.get("Total Recovered_text", "N/A"))
43             print("Fecha de Actualización: " + data.get("Last Update", "N/A"))
44         else:
45             print("No se pudieron obtener datos para el país especificado.")
46
```

```

47     else:
48         print(f"Error en la solicitud: {response.status_code}")
49         try:
50             error_message = response.json().get("message", "No se proporcionó un mensaje de error.")
51             print(f'Mensaje de error: {error_message}')
52         except:
53             print('No se pudo analizar el mensaje de error.')
54
55 if __name__ == "__main__":
56     while True:
57         # Solicitar al usuario que ingrese un país
58         pais_ingresado = input("Introduce el nombre del país (o escribe 'fin' para salir): ").strip()
59
60         # Verificar si el usuario desea salir del bucle
61         if pais_ingresado.lower() == 'fin':
62             break
63
64         # Obtener datos para el país ingresado
65         response = obtener_datos_por_pais(url, headers, pais_ingresado)
66
67         # Imprimir los datos obtenidos
68         imprimir_datos(response)
69

```

Evidencia de la función de la API
(Los nombre de los paises deben de estar en ingles)

```

vreyes@vreyes:~/Documentos/Recuperacion1$ /bin/python3 /home/vreyes/Documentos/Recuperacion1/01_app_parse01.py
Introduce el nombre del país (o escribe 'fin' para salir): spain
Datos del país:
País: Spain
Casos Confirmados: 13,914,811
Muertes: 121,760
Recuperados: 13,762,417
Fecha de Actualización: 2023-11-23 23:05
Introduce el nombre del país (o escribe 'fin' para salir): australia
Datos del país:
País: Australia
Casos Confirmados: 11,691,400
Muertes: 22,899
Recuperados: 11,651,856
Fecha de Actualización: 2023-11-23 23:05
Introduce el nombre del país (o escribe 'fin' para salir): taiwan
Datos del país:
País: Taiwan
Casos Confirmados: 10,241,523
Muertes: 19,005
Recuperados: 10,222,518
Fecha de Actualización: 2023-11-23 23:05
Introduce el nombre del país (o escribe 'fin' para salir): mexico
Datos del país:
País: Mexico
Casos Confirmados: 7,649,199
Muertes: 334,472
Recuperados: 6,899,865
Fecha de Actualización: 2023-11-23 23:05
Introduce el nombre del país (o escribe 'fin' para salir): fin
vreyes@vreyes:~/Documentos/Recuperacion1$ /bin/python3 /home/vreyes/Documentos/Recuperacion1/trackingcovid19.py

```

API Random Word

Función: selecciona aleatoriamente palabras de todo el diccionario de inglés con muchos parámetros.

Codigo:

```
1  '''
2  *   Nombre: Víctor Manuel Ramírez Reyes
3  *   Fecha: 23/11/2022
4  *   API: randomword api
5  *   Función: selecciona aleatoriamente palabras de todo el diccionario de inglés con muchos parámetros.
6  '''
7  import requests
8
9  url = "https://random-word-api.p.rapidapi.com/S/"
10
11 headers = {
12     "X-RapidAPI-Key": "29a251fa7fmsh5974211lee70d7fp1cfef5jsna94453741981",
13     "X-RapidAPI-Host": "random-word-api.p.rapidapi.com"
14 }
15
16 def obtener_palabras_por_letras(api_url, api_headers, letras):
17     endpoint = letras
18     response = requests.get(api_url + endpoint, headers=api_headers)
19     return response
20
21 def imprimir_palabras(response):
22     if response.status_code == 200:
23         data = response.json()
24         palabra = data.get("word")
25
```

```
26     if palabra:
27         print("Palabra encontrada:", palabra)
28     else:
29         print("No se encontraron palabras para las letras especificadas.")
30 else:
31     print(f"Error en la solicitud: {response.status_code}")
32     try:
33         error_message = response.json().get("message", "No se proporcionó un mensaje de error.")
34         print(f'Mensaje de error: {error_message}')
35     except:
36         print('No se pudo analizar el mensaje de error.')
37
38 if __name__ == "__main__":
39     while True:
40         # Solicitar al usuario que ingrese letras
41         letras_ingresadas = input("Introduce letras para completar una palabra en inglés aleatoriamente (o escribe 'q' para salir):")
42
43         # Verificar si el usuario desea salir del bucle
44         if letras_ingresadas.lower() == 'q':
45             break
46
47         # Obtener palabras para las letras ingresadas
48         response = obtener_palabras_por_letras(url, headers, letras_ingresadas)
49
50         # Imprimir la palabra obtenida
51         imprimir_palabras(response)
```

Evidencia de la función de la API

```
● vreyes@vreyes:~/Documentos/Recuperacion1$ /bin/python3 /home/vreyes/Documentos/Recuperacion1/randomwordapi.py
Introduce letras para completar una palabra en inglés aleatoriamente (o escribe 'q' para salir): car
Palabra encontrada: cardiaplegia
Introduce letras para completar una palabra en inglés aleatoriamente (o escribe 'q' para salir): teac
Palabra encontrada: teacherhood
Introduce letras para completar una palabra en inglés aleatoriamente (o escribe 'q' para salir): neig
Palabra encontrada: neighed
Introduce letras para completar una palabra en inglés aleatoriamente (o escribe 'q' para salir): sa
Palabra encontrada: saccharomycetales
Introduce letras para completar una palabra en inglés aleatoriamente (o escribe 'q' para salir): q
○ vreyes@vreyes:~/Documentos/Recuperacion1$
```