

Práctica API Parsing JSON

Instrucciones

Paso 1) Tomar como base el ejercicio 1.3.3.1 Parsing JSON with Application with Python

- Importar módulos necesarios.
- Cree variables de solicitud de API y construya una URL.
- Agregue funcionalidad de entrada del usuario.
- Agregue una función de salida para que el usuario pueda finalizar la aplicación.
- Muestra información del viaje para tiempo, distancia y uso de combustible.
- Itere a través de los datos JSON para extraer y generar las instrucciones.
- Mostrar mensajes de error para entradas de usuario no válidas.

Paso 2) Visita los siguientes sitios

- [Yandex](#)
- [Apli List](#)
- [Rapid API](#)
- [Public API Dev](#)

Paso 3) Seleccionar al menos tres API's y describirlas

Nombre de la API	Valid Email
Descripción de la API	Llegue a más clientes y mejore el ROI de su correo electrónico con nuestras validaciones de cheques SMTP.
API Portal / Home Page	https://validemail.io/

Nombre de la API	Cats API
Descripción de la API	Inserta y genera razas de gatos y fotografías aleatorias
API Portal / Home Page	https://apilist.fun/api/cats

Nombre de la API	Random Fox
Descripción de la API	Genera fotografías aleatorias de zorros
API Portal / Home Page	https://apilist.fun/api/randomfox

Paso 4) Cerciorarse que las API's no se repitan con tus compañeros de clase

Paso 5) Ve al portal o página de inicio de la API seleccionada.

Paso 6) Regístrate en el portal. Complete el formulario para crear una nueva cuenta para obtener la clave o KEY

Paso 7) Copia la clave de consumidor en el Bloc de notas para uso futuro.

Paso 8) Abra un archivo de script en blanco y guárdalo como practica01_unidad2.py, practica02_unidad2.py, practica03_unidad2.py

Paso 9) Agregar los siguientes datos de comentario

Descripción de la API

Autor

Fecha de creación

Paso 10) Importar los módulos **urllib.parse** y **requests**.

Paso 11) Crea la variable que construye la URL.

```
main_api = "http://api.coronatracker.com/v5/analytics/dailyNewStats?limit="
```

Paso 12) Utiliza la función input para datos de entrada de usuario

Paso 13) Crea la variable que realiza la solicitud

```
json_data = requests.get(url).json()
```

Paso 14) Analiza el dato regresado para obtener el código de estatus regresado e imprímelo.

Paso 15) Utiliza un ciclo infinito con **while True:** para probar tu API; además utiliza una condicional if donde cheque la entrada del usuario **Salir** o **S** para terminar el programa.

Paso 16) Prueba la funcionalidad de salir y con una impresión de pantalla

Agrega tu pantalla de salida con varias corridas

Primera API:

```
linux19@linux19:~/Documentos/otro/Eder$ /usr/bin/python3 /home/linux19/Documentos/otro/Eder/practica01_unidad2.py
Email: edergaelsaldanagalvan11@gmail.com
URL: https://validemail.io/v1/validate?api_key=e6VVXsxthNgYH3tUKfjfsNT0Uqskz6rp&email=edergaelsaldanagalvan11%40gmail.com
Valid
Email: aaaa@hsd.ol
URL: https://validemail.io/v1/validate?api_key=e6VVXsxthNgYH3tUKfjfsNT0Uqskz6rp&email=aaaa%40hsd.ol
Invalid
Email: tonyu705@gmail.com
URL: https://validemail.io/v1/validate?api_key=e6VVXsxthNgYH3tUKfjfsNT0Uqskz6rp&email=tonyu705%40gmail.com
Valid
```

Segunda API:

```
Cat here:
URL: https://api.thecatapi.com/v1/images/0XYvRd7oD
https://cdn2.thecatapi.com/images/0XYvRd7oD.jpg
Cat here: quit
Gracias por amar a los gatitos
```

Tercera API:

```
Fox here:
URL: https://randomfox.ca/floof/?ref=apilist.fun
https://randomfox.ca/images/61.jpg
Fox here:
URL: https://randomfox.ca/floof/?ref=apilist.fun
https://randomfox.ca/images/81.jpg
Fox here: quit
Que un zorro te acompañe siempre
```

Paso 17) Agrega tu código fuente documentado en GitHub y el archivo resultado