



Cofinanciado por
la Unión Europea



MINISTERIO
DE TRABAJO
Y ECONOMÍA SOCIAL



Fondos Europeos



Ejemplo PokeApi

Este archivo se encargará de recibir el nombre del Pokémon desde el formulario, hacer la petición a la PokeAPI, traer la imagen y poderes (si existe el Pokémon), y mostrar el resultado en la página web.

Antes de nada tenemos que **instalar flask**:

```
pip install flask
```

Ahora llega lo divertido, vamos a jugar ¡!!!!!!

```
from flask import Flask, render_template, request, jsonify
import requests

app = Flask(__name__)

# Ruta para el formulario
@app.route('/', methods=['GET', 'POST'])
def buscar_pokemon():
    if request.method == 'POST':
        # Obtener el nombre del pokemon desde el formulario
        pokemon_nombre = request.form['pokemon']

        # URL de la PokeAPI para buscar el pokemon por nombre
        url =
f"https://pokeapi.co/api/v2/pokemon/{pokemon_nombre.lower()}"

        try:
            # Realizamos la solicitud a la PokeAPI
            respuesta = requests.get(url)

            # Si la respuesta es exitosa (status code 200)
            if respuesta.status_code == 200:
                # Convertir la respuesta a formato JSON
                datos_pokemon = respuesta.json()

                # Extraer la imagen del pokemon y sus poderes
                imagen_pokemon =
datos_pokemon['sprites']['front_default']
                poderes_pokemon = [habilidad['ability']['name'] for
habilidad in datos_pokemon['abilities']]

                # Pasar los datos a la plantilla para mostrar
```



REGISTRO NACIONAL DE ASOCIACIONES N°611922
DECLARADA ENTIDAD DE UTILIDAD PÚBLICA ESTATAL
AGENCIA DE COLOCACIÓN: ID 0100000017

CENTRO EN MÁLAGA
C/DOS ACERAS 23, 29012
MÁLAGA | (+34) 952 300 500
ARRABAL@ARRABALEMPLEO.ORG

CENTRO EN CÁDIZ
TR.º ALAMEDA DE SOLANO, 32, 11130
CHICLANA DE LA FRONTERA | (+34) 956 900 312
CHICLANA@ARRABALEMPLEO.ORG



Cofinanciado por
la Unión Europea



MINISTERIO
DE TRABAJO
Y ECONOMÍA SOCIAL



Fondos Europeos



Junta de Andalucía
Fondo de Empleo Juvenil
Ayuda al Empleo Juvenil



Ayuda
en Acción
Impulsa Empleo Joven

```

        return render_template('index2.html',
pokemon_nombre=pokemon_nombre,
                                imagen=imagen_pokemon,
poderes=poderes_pokemon)
    else:
        # Si el pokemon no existe, mostramos un mensaje de error
        mensaje_error = "Pokémon no encontrado. Intenta con otro
nombre."
        return render_template('index2.html',
error=mensaje_error)

except requests.exceptions.RequestException as e:
    # Si ocurre un error al hacer la petición, mostrar un error
    mensaje_error = f"Error al intentar obtener los datos: {e}"
    return render_template('index2.html', error=mensaje_error)

# Si es un GET (cuando se carga la página por primera vez), solo
mostramos el formulario vacío
return render_template('index2.html')

if __name__ == '__main__':
    app.run(debug=True)

```

No olvidemos instalar:

```
PS C:\Users\Empleo1\Desktop\Ejercicios\Python\Flask\BD> pip install requests
```

Explicación del código:

1. **Flask y requests:** Importamos las librerías necesarias para Flask y requests para hacer las peticiones HTTP.
2. **Ruta '/':** Creamos la ruta /, que acepta tanto GET como POST:
 - Si es un **GET**, mostramos el formulario vacío.
 - Si es un **POST**, recibimos el nombre del Pokémon, hacemos la petición a la PokeAPI y procesamos la respuesta.
3. **Petición a la PokeAPI:** Usamos la URL https://pokeapi.co/api/v2/pokemon/{pokemon_nombre} para buscar el Pokémon. Si la API responde correctamente, extraemos la imagen y las habilidades del Pokémon.
4. **Manejo de errores:** Si el Pokémon no existe o ocurre algún error en la petición, mostramos un mensaje de error en la misma página.
5. **render_template:** Usamos render_template para enviar los resultados (nombre, imagen y habilidades) o el mensaje de error a la plantilla index2.html.



REGISTRO NACIONAL DE ASOCIACIONES N°611922
DECLARADA ENTIDAD DE UTILIDAD PÚBLICA ESTATAL
AGENCIA DE COLOCACIÓN: ID 0100000017

CENTRO EN MÁLAGA
C/DOS ACERAS 23, 29012
MÁLAGA | (+34) 952 300 500
ARRABAL@ARRABALEMPLEO.ORG

CENTRO EN CÁDIZ
TR.º ALAMEDA DE SOLANO, 32, 11130
CHICLANA DE LA FRONTERA | (+34) 956 900 312
CHICLANA@ARRABALEMPLEO.ORG



Cofinanciado por
la Unión Europea



MINISTERIO
DE TRABAJO
Y ECONOMÍA SOCIAL



Fondos Europeos



Ayuda
en Acción
Impulsa Empleo Joven

Ahora vamos con el index2.html que debe estar situado en la carpeta templates para que flask lo encuentre.

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Buscar Pokémon</title>
</head>
<body>
  <h1>Buscar Pokémon</h1>

  <!-- Formulario para ingresar el nombre del Pokémon -->
  <form method="POST">
    <label for="pokemon">Nombre del Pokémon:</label>
    <input type="text" id="pokemon" name="pokemon" required>
    <button type="submit">Buscar</button>
  </form>

  {% if pokemon_nombre %}
    <!-- Si el Pokémon se encuentra, mostrar los detalles -->
    <h2>Información de {{ pokemon_nombre.capitalize() }}</h2>
    
    <h3>Habilidades:</h3>
    <ul>
      {% for poder in poderes %}
        <li>{{ poder }}</li>
      {% endfor %}
    </ul>
  {% elif error %}
    <!-- Si hay un error, mostrar el mensaje de error -->
    <p style="color: red;">{{ error }}</p>
  {% endif %}
</body>
</html>
```



REGISTRO NACIONAL DE ASOCIACIONES N°611922
DECLARADA ENTIDAD DE UTILIDAD PÚBLICA ESTATAL
AGENCIA DE COLOCACIÓN: ID 0100000017

CENTRO EN MÁLAGA
C/DOS ACERAS 23, 29012
MÁLAGA | (+34) 952 300 500
ARRABAL@ARRABALEMPLEO.ORG

CENTRO EN CÁDIZ
TR.ª ALAMEDA DE SOLANO, 32, 11130
CHICLANA DE LA FRONTERA | (+34) 956 900 312
CHICLANA@ARRABALEMPLEO.ORG



Cofinanciado por
la Unión Europea



MINISTERIO
DE TRABAJO
Y ECONOMÍA SOCIAL



Fondos Europeos



Explicación de la plantilla index2.html:

1. **Formulario:** El formulario permite al usuario ingresar el nombre de un Pokémon y hacer la petición al servidor con el método POST.
2. **Mostrar resultados:** Si el nombre del Pokémon (pokemon_nombre) es recibido y existe, se muestran su imagen y habilidades.
3. **Manejo de errores:** Si hay un error (como que el Pokémon no exista o algún problema con la API), se muestra un mensaje de error en rojo.

Cómo funciona:

- El usuario ingresa el nombre del Pokémon en el formulario.
- Flask recibe ese nombre, hace la petición a la PokeAPI, obtiene la imagen y las habilidades del Pokémon y las pasa a la plantilla HTML.
- Si el Pokémon no se encuentra, se muestra un mensaje de error.

Vamos a ejecutar y ver que funciona, veremos algo así:

```
PS C:\Users\Empleo1\Desktop\Ejercicios\Python\Flask\BD> python app2.py
>>
* Serving Flask app 'app2'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 303-743-524
```

Abrimos en nuestro navegador la dirección ip local indicada y ya podemos probar que todo funciona.



REGISTRO NACIONAL DE ASOCIACIONES N°611922
DECLARADA ENTIDAD DE UTILIDAD PÚBLICA ESTATAL
AGENCIA DE COLOCACIÓN: ID 0100000017

CENTRO EN MÁLAGA
C/DOS ACERAS 23, 29012
MÁLAGA | (+34) 952 300 500
ARRABAL@ARRABALEMPLEO.ORG

CENTRO EN CÁDIZ
TR.ª ALAMEDA DE SOLANO, 32, 11130
CHICLANA DE LA FRONTERA | (+34) 956 900 312
CHICLANA@ARRABALEMPLEO.ORG



Cofinanciado por
la Unión Europea



MINISTERIO
DE TRABAJO
Y ECONOMÍA SOCIAL



Fondos Europeos

ASOCIACIÓN
Arrabal



Ayuda
en Acción
Impulsa Empleo Joven

← ↻ ⓘ 127.0.0.1:5000

Buscar Pokémon

Nombre del Pokémon:

Información de Pikachu:



Habilidades:

- static
- lightning-rod

Aplicar estilos en Flask:

Si queremos aplicar estilos a nuestro html:

En una aplicación Flask, la forma estándar de organizar los archivos estáticos, como las *hojas de estilo (CSS)*, imágenes o archivos JavaScript, es utilizando la carpeta **static**.

Luego vinculamos el css en nuestro html con la siguiente línea en la cabecera:

```
<link rel="stylesheet" href="{{ url_for('static', filename='css/styles.css') }}">
```



REGISTRO NACIONAL DE ASOCIACIONES N°611922
DECLARADA ENTIDAD DE UTILIDAD PÚBLICA ESTATAL
AGENCIA DE COLOCACIÓN: ID 0100000017

CENTRO EN MÁLAGA
C/DOS ACERAS 23, 29012
MÁLAGA | (+34) 952 300 500
ARRABAL@ARRABALEMPLEO.ORG

CENTRO EN CÁDIZ
TR.º ALAMEDA DE SOLANO, 32, 11130
CHICLANA DE LA FRONTERA | (+34) 956 900 312
CHICLANA@ARRABALEMPLEO.ORG

Tabla resumen de Jinja:

Estructura	Descripción	Ejemplo en Jinja
Mostrar datos	Mostrar una variable o valor en el HTML.	<code>{{ variable }}</code> Ejemplo: <code>{{ nombre }}</code> muestra el valor de la variable nombre.
Condicional IF	Condicional básico: se ejecuta si la condición es verdadera.	<code>{% if condicion %}...{% endif %}</code> Ejemplo: <code>{% if edad >= 18 %}</code> Eres mayor de edad <code>{% endif %}</code>
Condicional IF-ELSE	Condicional con una rama alternativa si la condición es falsa.	<code>{% if condicion %}...{% else %}...{% endif %}</code> Ejemplo: <code>{% if edad >= 18 %}</code> Adulto <code>{% else %}</code> Menor <code>{% endif %}</code>
Condicional IF-ELIF-ELSE	Condicional con múltiples opciones, similar a un "if-else if-else" en otros lenguajes.	<code>{% if condicion1 %}...{% elif condicion2 %}...{% else %}...{% endif %}</code> Ejemplo: <code>{% if edad >= 18 %}</code> Adulto <code>{% elif edad >= 12 %}</code> Adolescente <code>{% else %}</code> Niño <code>{% endif %}</code>
Bucle FOR	Bucle para iterar sobre una lista o un rango.	<code>{% for item in lista %}...{% endfor %}</code> Ejemplo: <code>{% for nombre in nombres %}</code> <code><p>{{ nombre }}</p></code> <code>{% endfor %}</code>
Bucle FOR con índice	Bucle con índice para acceder a la posición de los elementos.	<code>{% for item in lista %} {{ loop.index }}: {{ item }} {% endfor %}</code> Ejemplo: <code>{% for i, nombre in enumerate(nombres) %} {{ i }}: {{ nombre }} {% endfor %}</code>
Bucle con rango	Bucle que itera sobre un rango de números.	<code>{% for i in range(1, 5) %}...{% endfor %}</code> Ejemplo: <code>{% for i in range(1, 6) %}</code> <code><p>{{ i }}</p></code> <code>{% endfor %}</code>
Verificar si la lista no está vacía	Comprobar si una lista o conjunto tiene elementos.	<code>{% if lista %}...{% endif %}</code> Ejemplo: <code>{% if nombres %}</code> <code><p>Hay nombres</p></code> <code>{% else %}</code> <code><p>No hay nombres</p></code> <code>{% endif %}</code>
Mostrar el valor de un diccionario	Acceder y mostrar un valor de un diccionario usando claves.	<code>{{ diccionario['clave'] }}</code> Ejemplo: <code>{{ persona['nombre'] }}</code> muestra el valor asociado a 'nombre'.
Condicional con is	Condicional para verificar tipos o si una variable es nula.	<code>{% if variable is none %}...{% endif %}</code> Ejemplo: <code>{% if nombre is not none %}</code> <code><p>Nombre: {{ nombre }}</p></code> <code>{% endif %}</code>



Cofinanciado por
la Unión Europea



MINISTERIO
DE TRABAJO
Y ECONOMÍA SOCIAL



Fondos Europeos



Ayuda
en Acción
Impulsa Empleo Joven

Notas:

POST: Se usa cuando queremos **enviar datos** al servidor y, generalmente, modificar algo (por ejemplo, realizar una búsqueda y mostrar resultados).

GET: Se usa cuando simplemente queremos **recuperar** datos del servidor sin cambiar nada.

(Si usáramos method="GET" en nuestro formulario html, los datos se enviarían como parámetros en la URL, lo cual no es tan seguro ni adecuado para este tipo de operación.)



REGISTRO NACIONAL DE ASOCIACIONES Nº611922
DECLARADA ENTIDAD DE UTILIDAD PÚBLICA ESTATAL
AGENCIA DE COLOCACIÓN: ID 0100000017

CENTRO EN MÁLAGA

C/DOS ACERAS 23, 29012
MÁLAGA | (+34) 952 300 500
ARRABAL@ARRABALEMPLEO.ORG

CENTRO EN CÁDIZ

TR.ª ALAMEDA DE SOLANO, 32, 11130
CHICLANA DE LA FRONTERA | (+34) 956 900 312
CHICLANA@ARRABALEMPLEO.ORG