

Fecha Inicial: 27/05/25

Grupo De Trabajo: Luis Tinoco y Matías Benavides

Fecha:	27/05/25
Nombre:	Matias Benavides
¿Qué he hecho?	Hice lo de agregar pokemon
¿Qué haré a continuación?	Continuar con la siguiente funcion
¿En qué ocupo ayuda?	Ver que avances hemos hecho y avanzar

## Evidencias:

```
#===== 12. Agregar Pokemon =====
def ejecutarAgregar(pokepad):
    """
    Permite agregar un nuevo Pokémon al diccionario si su ID no está registrado aún.
    """
    ventanaAgregar = tk.Toplevel(pokepad.ventana)
    ventanaAgregar.title("Agregar Pokémon")
    ventanaAgregar.geometry("320x180")

    ttk.Label(ventanaAgregar, text="Ingrese el ID del Pokémon a agregar:").pack(pady=10)
    entradaID = ttk.Entry(ventanaAgregar)
    entradaID.pack(pady=5)

    def obtenerInfoPokemon(idPoke):
        url = f"https://pokeapi.co/api/v2/pokemon/{idPoke}"
        try:
            respuesta = requests.get(url, timeout=8)
            datos = respuesta.json()

            idApi = datos['id']
            nombre = datos['name']
            esShiny = random.choice([True, False])
            peso = datos['weight'] * 10 # en gramos
            altura = datos['height'] * 10 # en centímetros

            # Estadísticas ordenadas
            statsOrden = ['hp', 'attack', 'defense', 'special-attack', 'special-defense', 'speed']
            statsValores = []
            for statNombre in statsOrden:
                for stat in datos['stats']:
                    if stat['stat']['name'] == statNombre:
                        statsValores.append(stat['base_stat'])
                        break

            totalEstats = sum(statsValores)
            statsTupla = tuple(statsValores)

            # Obtener los tipos del Pokémon desde la estructura de datos
            tipos = []
            for tipo_info in datos['types']:
                tipo_nombre = tipo_info['type']['name']
                tipos.append(tipo_nombre)
            tipos = tuple(tipos) # Convertimos la lista a una tupla para que sea inmutable

            # Determinar la URL de la imagen según si el Pokémon es shiny o no
            if esShiny:
                urlImagen = datos['sprites']['front_shiny']
            else:
                urlImagen = datos['sprites']['front_default']

            return {
                idApi: [
                    nombre,
                    (esShiny, peso, altura),
                    [totalEstats, statsTupla],
                    tipos,
                    urlImagen
                ]
            }

        except Exception as e:
            print(f"Error al obtener info de Pokémon {idPoke}: {e}")
            return None

    def leerIDsDiccionario():
        ids = set()
        .
```

```

def generarHtml(pagina, numeroPagina):
    nombreArchivo = f"Shinys{numeroPagina}.html"

    archivo = open(nombreArchivo, "w", encoding="utf-8")

    encabezado = """<!DOCTYPE html>
<html>
<head>
    <title>Pokémon Shinys</title>
    <style>
        table { border-collapse: collapse; width: 100%; }
        th, td { padding: 8px; text-align: left; border: 1px solid #ddd; }
        th { background-color: #ffcc00; color: black; }
        tr:nth-child(even) { background-color: #081a41; }
        img { width: 80px; }
    </style>
</head>
<body>
    <h1>Lista de Pokémon Shiny</h1>
    <table>
        <tr>
            <th>ID</th>
            <th>Nombre</th>
            <th>Shiny</th>
            <th>Peso (kg)</th>
            <th>Altura (cm)</th>
            <th>Estadísticas</th>
            <th>Total</th>
            <th>Tipos</th>
            <th>Imagen</th>
        </tr>
"""
    archivo.write(encabezado)
    for idPoke, datos in pagina:
        nombre = datos[0]
        tupla = datos[1]
        shiny = tupla[0]
        peso = tupla[1]
        altura = tupla[2]

        estadisticas = datos[2]
        total_stats = estadisticas[0]
        stats = estadisticas[1]
        tiposLista = datos[3]
        tipos = ', '.join(tiposLista)
        url = datos[4]

        filaHtml = f"""
        <tr>
            <td>{idPoke}</td>
            <td>{nombre}</td>
            <td>{shiny}</td>
            <td>{peso}</td>
            <td>{altura}</td>
            <td>{stats}</td>
            <td>{total_stats}</td>
            <td>{tipos}</td>
            <td></td>
        </tr>
"""
        archivo.write(filaHtml)
    cierre = """
</table>
</body>
</html>
"""

```