



Estás empezando a trabajar en el laboratorio del profesor Oak y te ha asignado una tarea crucial: diseñar la infraestructura técnica para analizar los datos de la Pokédex.

El Desafío: Debes consumir la Pokeapi (<https://pokeapi.co/>) para listar los primeros 50 Pokémon. La vista principal debe incluir las columnas:

- ID
- Nombre
- Tipo(s)
- Altura
- Peso

Teniendo la tabla, el profesor Oak necesita lo siguiente:

- 1) Filtro de Peso: Identificar todos los Pokémon que pesen más de 30 y menos de 80
- 2) Filtro de Tipo: Identificar todos los Pokémon tipo “grass”
- 3) Filtro Combinado: Identificar todos los Pokémon tipo “flying” que midan más de 10
- 4) Transformación de datos: Crear una nueva columna con todos los nombres de los Pokémon invertidos, por ejemplo, “bulbasaur” → “ruasablub”

Especificaciones Técnicas:

- Backend: Debes utilizar **Python** y **Django** para la visualización y lógica de servidor.
- Arquitectura: No te limites a una llamada simple a la API. Se evaluará como manejas la persistencia de datos.
- DevOps: La solución debe estar contenerizada con **Docker**.
- Código: Se espera uso de variables de entorno y una estructura de carpetas profesional.

Rubrica:

Estructura y Patrones: Uso correcto de modelos, vistas y servicios en Django.

Legibilidad y Rigor: Código documentado, tipado y manejo de errores.

Funcionalidad: Cumplimiento exacto de los filtros de Oak.

UX/UI: Calidad visual del dashboard y facilidad de navegación.

Autonomía DevOps: Facilidad para desplegar el entorno con Docker.

¡Buena Suerte!