

# Caso - JSON y CSV en Base de Datos

## Contexto

El equipo de Seguridad Informática de Mercado Libre, se encarga de hacer las reválidas anuales del proceso de clasificación de la información. Sabemos por el feedback del año pasado, que generar reuniones presenciales para validar esto, es un poco molesto para el usuario, más aún cuando las bases no son muy críticas. Por eso y dado que estamos cerca de la fecha de reválida de bases de datos, este año queremos hacerlo de manera automática. Pensamos pedirle a los managers de las bases más críticas que nos den el OK por mail.

## Características

Disponemos de un archivo de tipo JSON con la información de la clasificación de las bases de datos y un archivo CSV con información de usuarios y su manager.

El archivo JSON puede tener campos incompletos. En dicho caso será necesario encontrar alguna solución para procesarlo.

El archivo CSV tiene la siguiente forma:

row_id,	user_id,	user_state,	user_manager
---------	----------	-------------	--------------

## Objetivo

Generar un programa que, a partir de los archivos dados, guarde su contenido en una base de datos y por cada registro guardado, en donde la clasificación sea alta (high), envíe un email al manager del owner pidiendo su OK respecto de la clasificación.

Se espera tener una base de datos con la siguiente información almacenada:

- nombre de la base de datos
- el email del owner
- el email del manager
- la clasificación de la misma

## Entregables

Se espera:

- Código fuente (en zip o url al repositorio)
- Instrucciones para la ejecución de la aplicación (incluida cualquier aplicación o librería a instalar para el correcto funcionamiento del programa)
- Descripción de la aplicación realizada, supuestos, problemas y soluciones con los que se encontró al realizar la misma.

## Consideraciones generales

A tener en cuenta:

- Preferentemente el challenge debe ser realizado en Python o Golang.
- Se podrán crear todas las funciones complementarias que se consideren necesarias para un correcto funcionamiento de la aplicación.
- Se recomienda modularizar y aplicar buenas prácticas de programación para un mejor entendimiento del código.

Bonus (opcional):

- Base de datos esté normalizada
- Tener el código en Github

Ultra Bonus (opcional)

- Dockerizar la aplicación