

## Prueba

Para esta prueba tienes **48 Hrs.** para escribir un simple proceso ETL (Extract-Transform-Load), transformar el archivo **clientes.csv** que se encuentra dentro del **.rar** adjunto y cargarlo a una base de datos SQLite, la contraseña del archivo es: **v9fZ^8jfVVN3**

Deberás generar un repositorio en github con la siguiente estructura:

```
python-test
├── output
│   └── .gitkeep
├── .gitignore
├── database.db3
├── main.py
├── README.md
├── requirements.txt
└── setup.py
```

En **main.py** es donde escribirás tu script, este al ejecutarlo debe solicitar la ruta de un archivo en el disco, leer ese archivo, transformarlo, guardar los archivos transformados en el directorio **output**, y cargarlos en la base de datos **database.db3** que se encuentra en el directorio.

La estructura del archivo **clientes.csv** es la siguiente:

clientes.csv	
Campo	Descripción
fiscal_id	Rut del cliente con dígito verificador
first_name	Primer nombre del cliente
last_name	Apellido del cliente
gender	Genero del cliente
fecha_nacimiento	Fecha de nacimiento del cliente
fecha_vencimiento	Fecha de vencimiento de la deuda
deuda	Deuda a pagar

direccion	Dirección de facturación
correo	Email del cliente
estatus_contacto	Estatus del medio de contacto (valido, invalido)
prioridad	
telefono	

Utilizando el archivo mencionado anteriormente, el proceso deberá generar los siguientes archivos

clientes.xlsx		
Campo	Tipo de dato	Columna del archivo original
fiscal_id	STRING	{fiscal_id}
first_name	STRING	{first_name}
last_name	STRING	{last_name}
gender	STRING	{gender}
birth_date	DATE	{fecha_nacimiento}
age	INT	-
age_group	INT	-
due_date	DATE	{fecha_vencimiento}
delinquency	INT	{current-date} days - {due_date} days
due_balance	INT	{deuda}
address	STRING	direccion

emails.xlsx		
Campo	Tipo de dato	Columna del archivo original
fiscal_id	STRING	{fiscal_id}
email	STRING	{correo}
status	STRING	{estatus_contacto}
priority	INT	{prioridad}

phones.xlsx		
Campo	Tipo de dato	Columna del archivo original
fiscal_id	STRING	{fiscal_id}
phone	STRING	{telefono}
status	STRING	{estatus_contacto}
priority	INT	{prioridad}

Para calcular el campo **age\_group** se siguen las siguientes reglas.

Grupo	Regla
1	Edad Menor o igual a 20
2	Edad entre 21 a 30
3	Edad entre 31 a 40
4	Edad entre 41 a 50
5	Edad entre 51 a 60
6	Edad Mayor a 60

Una vez que tengas los tres archivos ya transformados y validados, el script debe cargar la información en la base de datos **database.db3**, en tres tablas (customers, emails, phones) las cuales deben tener el mismo formato de los archivos antes mencionados (es decir, mismas columnas y tipos de datos).

Por último deberás enviar el enlace de tu repositorio al siguiente correo electrónico: **henry@colectia.com** para su revisión

Consideraciones:

1. El código debe correr en python 3.8.
2. La lectura y transformación del archivo debe realizarse con pandas.
3. Indentación de 2 espacios
4. Debes utilizar el archivo **clientes.csv** para realizar tu script, ya que este será el archivo que utilizaremos para evaluar la prueba.
5. Los datos deben estar normalizados, por ejemplo, todos los strings deben almacenarse en mayúscula, los valores enteros deben guardarse como enteros.
6. Para nosotros es muy importante el performance del código, es por esto que debes utilizar las mejores prácticas de programación en pandas para el manejo de memoria.