

CrediClub – Test de ingeniería analítica

```
In [ ]: from el_process import insert_data, read_excel, create_tables
import sqlite3
import pandas as pd
```

Modelado base de datos

Basado en la estructura de las tablas se propuso el siguiente modelado para la base de datos:



```
CREATE TABLE IF NOT EXISTS "managers" (  
    "GestorID" INTEGER NOT NULL,  
    "OficinaID" INTEGER,  
    PRIMARY KEY ("GestorID")  
);
```

```
CREATE TABLE IF NOT EXISTS "credits" (  
    "CreditoID" INTEGER NOT NULL,  
    "Ciclo" INTEGER,  
    "MontoPendiente" REAL,  
    "GestorID" INTEGER,  
    PRIMARY KEY ("CreditoID"),  
    FOREIGN KEY ("GestorID") REFERENCES "managers" ("GestorID") ON DELETE NO ACTION ON  
UPDATE NO ACTION  
);
```

```
CREATE TABLE IF NOT EXISTS "offices" (  
    "OficinaID" INTEGER NOT NULL,  
    "Oficina" TEXT,  
    PRIMARY KEY ("OficinaID"),  
    FOREIGN KEY ("OficinaID") REFERENCES "managers" ("OficinaID") ON DELETE NO ACTION  
ON UPDATE NO ACTION  
);
```

Extraccion y carga de datos

Creación tablas SQLite

Una vez definido los esquemas de las tablas, vamos a crear un instancia de SQLite y crear las tablas en esta base de datos.

La definición de las funciones se encuentra en el modulo `el_procesos.py`.

```
In [ ]: create_tables()
```

```
Tabla managers creada con exito
Tabla credits creada con exito
Tabla offices creada con exito
```

Leer datos archivo excel

```
In [ ]: data = read_excel()
```

Cargar datos en SQLite

```
In [ ]: for name, df in data.items():
        print(name)
        insert_data(df, name)
```

```
Tabla credits
Registros insertados 25050
Tabla gestores
Registros insertados 179
Tabla oficina
Registros insertados 158
```

Analizar datos

Conexión sqlite

```
In [ ]: con = sqlite3.connect("credits.db")
```

- Número de créditos por oficina

```
In [ ]: q1 = pd.read_sql("""
        SELECT
        o.Oficina,
        COUNT(c.CreditoID) total_credits
        FROM credits c
        LEFT JOIN managers m ON m.GestorID = c.GestorID
        LEFT JOIN offices o ON o.OficinaID = m.OficinaID
        GROUP BY o.Oficina
        ORDER BY total_credits DESC;
        """, con)

q1
```

Out[]:

	Oficina	total_credits
0	CORPORATIVO	3356
1	MEXICALI	354
2	PUEBLA SATELITE	332
3	CUERNAVACA NORTE	331
4	PUEBLA NORTE	313
...
153	MOTUL	34
154	NAVOJOA	27
155	IRAPUATO	19
156	TEPOTZOTLAN	1
157	OAXACA UNIVERSIDAD	1

158 rows × 2 columns

- Monto pendiente acumulado por oficina

In []:

```
q2 = pd.read_sql("""
    SELECT
    o.Oficina,
    SUM(MontoPendiente) MontoPendiente
    FROM credits c
    LEFT JOIN managers m ON m.GestorID = c.GestorID
    LEFT JOIN offices o ON o.OficinaID = m.OficinaID
    GROUP BY o.Oficina
    ORDER BY MontoPendiente DESC;
""", con)
q2
```

Out[]:

	Oficina	MontoPendiente
0	CORPORATIVO	2.270447e+08
1	MEXICALI	5.241719e+07
2	TORREON-MATAMOROS COAHUILA	5.124982e+07
3	REYNOSA JUAREZ	4.760127e+07
4	VILLA AHUMADA-CD. JUAREZ ORIENTE	4.625195e+07
...
153	VALLADOLID	1.699358e+06
154	CALKINI	1.507150e+06
155	NAVOJOA	1.337007e+06
156	OAXACA UNIVERSIDAD	4.883676e+04
157	TEPOTZOTLAN	2.217516e+04

158 rows × 2 columns

- Top 10 gestores por monto pendiente promedio y a que oficina pertenecen (solo de ciclos 1)

```
In [ ]: q3 = pd.read_sql("""
        WITH monto_gestores AS (
            SELECT
                GestorID,
                AVG(MontoPendiente) MontoPendientePromedio
            FROM credits c
            WHERE Ciclo = 1
            GROUP BY GestorID
            ORDER BY MontoPendientePromedio DESC
            LIMIT 10
        )
        SELECT
            mg.GestorID, mg.MontoPendientePromedio, o.Oficina
        FROM monto_gestores mg
        LEFT JOIN managers m ON m.GestorID = mg.GestorID
        LEFT JOIN offices o ON o.OficinaID = m.OficinaID;
        """, con)
q3
```

```
Out[ ]:
```

	GestorID	MontoPendientePromedio	Oficina
0	29258	151702.713474	IRAPUATO
1	16535	141036.094500	CHOLULA
2	22580	127679.160000	MOTUL
3	22509	125072.669674	GUANAJUATO
4	27101	124183.936864	ZACATELCO
5	28809	119502.720094	SAN JUAN DEL RIO QRO
6	15201	118758.527596	TIJUANA OTAY
7	29236	117231.404220	TULTITLAN
8	26710	115688.847750	VALLARTA
9	14523	110918.704312	MEXICALI