Instituto Politécnico Nacional Escuela Superior de Cómputo

Data Mining

Práctica 4: Proceso de ETL, caso "precipitación pluvial en CDMX"

Graciano Herrera Gabriel
Meza Zamora Abraham Manuel

6 de abril de 2022

Índice

Objetivo: Desarrollar una herramienta ETL para procesar archivos de Excel para el caso "Precipitación pluvial (PP), con la técnica de recolección para depósito húmedo (H))", durante el periodo "2010 al 2019"

1. Definición de flujo de datos del ETL

La primera parte del proceso consiste en la definición del modelo de la base de datos, en la cual reducimos el número de columnas, utilizando los datos que nosotros creímos más útiles para desarrollar la práctica, comenzando por el id del detector, el valor registrado, la fecha y la semana.

Posteriormente hicimos la conversión del archivo de excel a csv, esto para poder reutilizar parte del código empleado en la práctica anterior. Y finalmente procesamos la información con un script hecho en python, ya que es bastante sencillo de escribir código por las diferentes bibliotecas que ofrece out of the box.

Comenzamos por procesar el campo de la fecha, aquí partimos el campo en tres cadenas, el día, el mes y el año, al inicio del procesamiento llevamos un contador de la semana, y a medida que procesamos cada fila dentro del archivo, aumentamos el contador de la semana. Por cada medida conocemos su columna y su ubicación, por lo que usando la estructura del archivo generamos una query por cada columna con valores no nulos.

2. Estructura de la tabla de hechos principal y catálogos

2.1. Tabla de hechos principal

idDetector	fecha	valor	dia	mes	anio	semana

2.2. Catálogo de estaciones de monitoreo

id	siglas	nombre	municipio	estado
1	LOM	Lomas	Miguel Hidalgo	CDMX
2	TEC	Cerro del Tepeyac	Gustavo A. Madero	CDMX
3	DIC	Diconsa	Tlalpan	CDMX
4	MCM	Museo de la Ciudad de México	Cuauhtémoc	CDMX
5	TLA	Tlalnepantla	Tlalnepantla de Baz	Edo Mex
6	XAL	Xalostoc	Ecatepec de Morelos	Edo Mex
7	EDL	Ex Convento Desierto de los Leones	Cuajimalpa de Morelos	CDMX
8	IBM	Legaria	Miguel Hidalgo	CDMX
9	NEZ	Nezahualcóyotl	Nezahualcóyotl	Edo Mex
10	MON	Montecillo	Texcoco	Edo Mex
11	EAJ	Ecoguardas Ajusco	Tlalpan	CDMX
12	AJU	Ajusco	Tlalpan	CDMX
13	MPA	Milpa Alta	Milpa Alta	CDMX
14	SNT	San Nicolás Totolapan	La Magdalena Contreras	CDMX
15	COR	CORENA	Xochimilco	CDMX
16	LLA	Laboratorio de Análisis Ambiental	Gustavo A. Madero	CDMX

3. Exploración de los datos integrados

3.1. Cantidad de registros

3.1.1. Totales

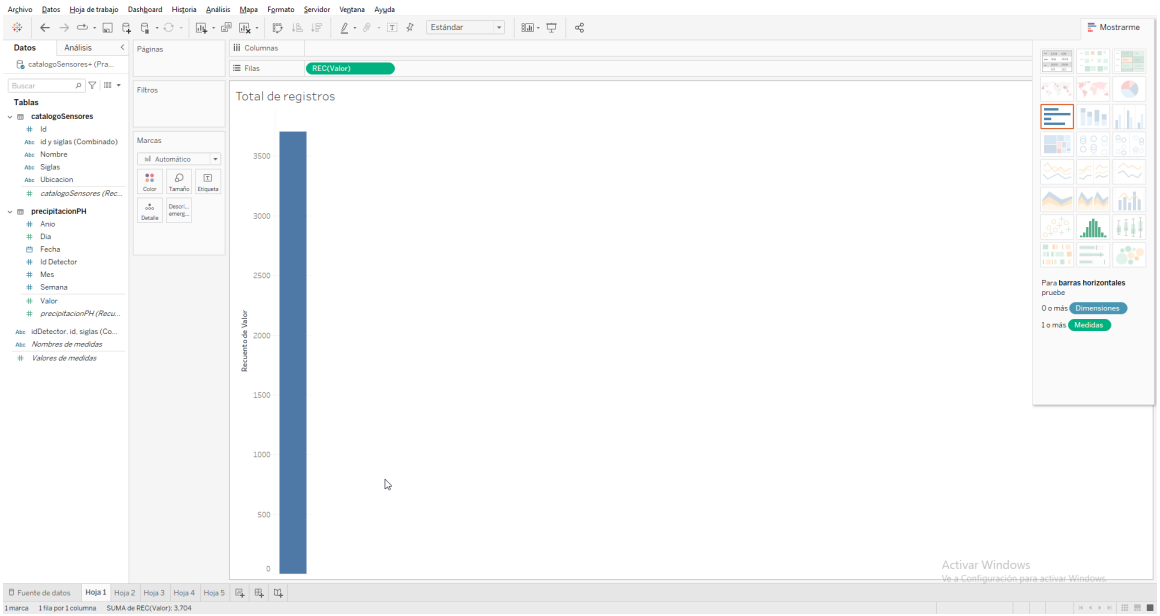

Figura 1: Cantidad de registros totales en Tableau.

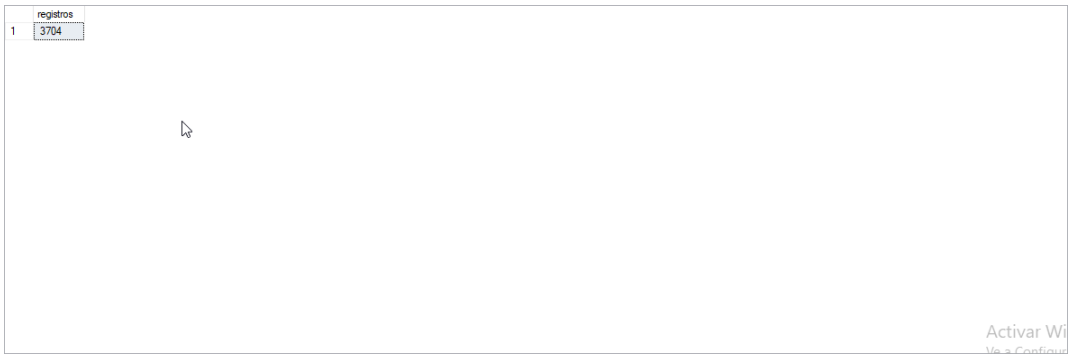

Figura 2: Vista de la cantidad de registros totales.

3.1.2. Por año

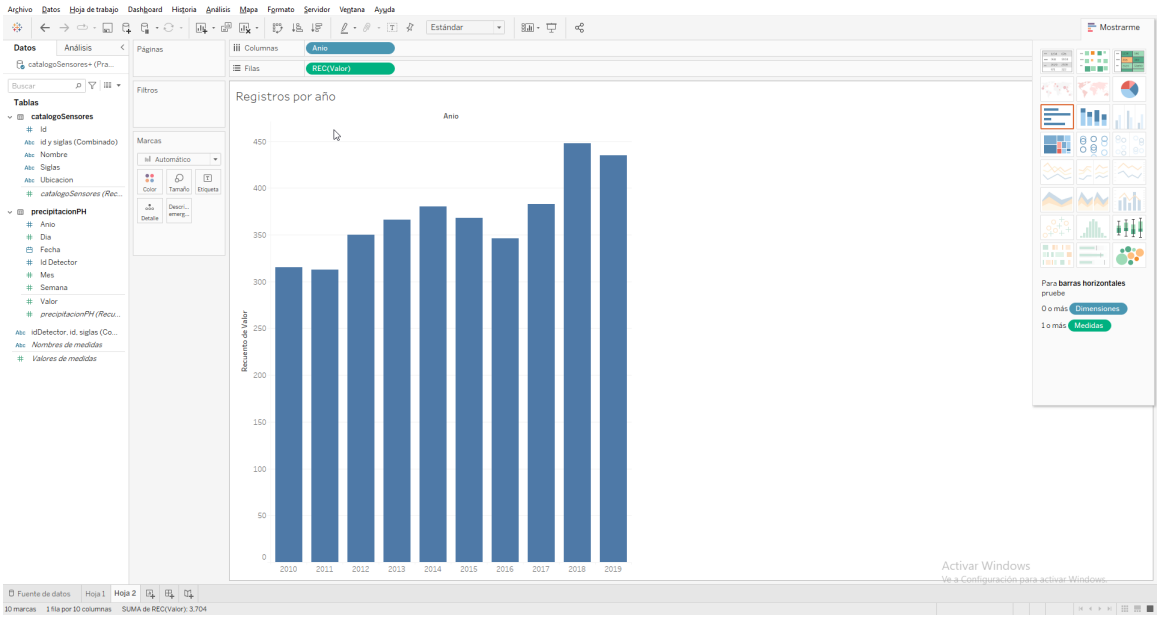

Figura 3: Cantidad de registros por año en Tableau.

	registros	anio
1	315	2010
2	313	2011
3	350	2012
4	366	2013
5	380	2014
6	368	2015
7	346	2016
8	383	2017
9	448	2018
10	435	2019

Figura 4: Vista de la cantidad de registros por año.

3.2. Tendencia en el tiempo de la precipitación pluvial

3.2.1. Semana

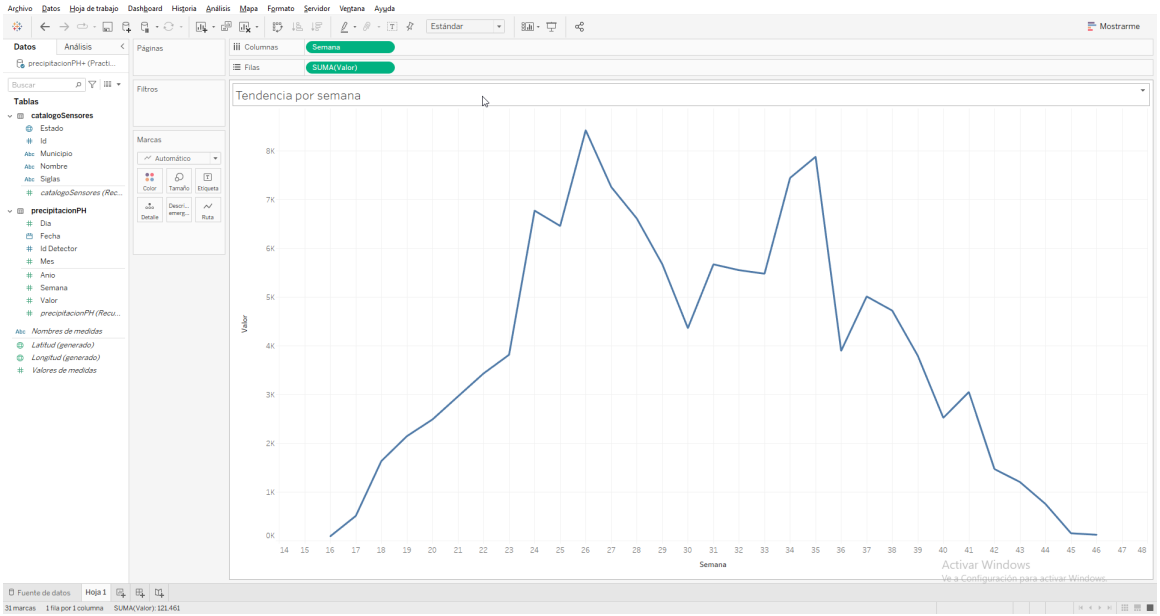

Figura 5: Tendencia precipitación semana en Tableau.

	precipitacion	semana
1	93.7	16
2	512.69	17
3	1638.06	18
4	2148.58	19
5	2490.98	20
6	2964.26	21
7	3435.14	22
8	3818.53	23
9	6776.1	24
10	6461.62	25
11	8425.66	26
12	7261.06	27
13	6615.3	28
14	5675.13	29
15	4365.67	30
16	5674.22	31
17	5554.89	32
18	5482.11	33
19	7447.37	34
20	7881.27	35
21	3899.17	36
22	5015.3	37
23	4724.9	38
24	3799.81	39
25	2525.33	40
26	3054.39	41
27	1474.55	42
28	1207.01	43
29	756.98	44
30	155.29	45
31	125.78	46

Figura 6: Vista de la tendencia precipitación semana.

3.2.2. Mes

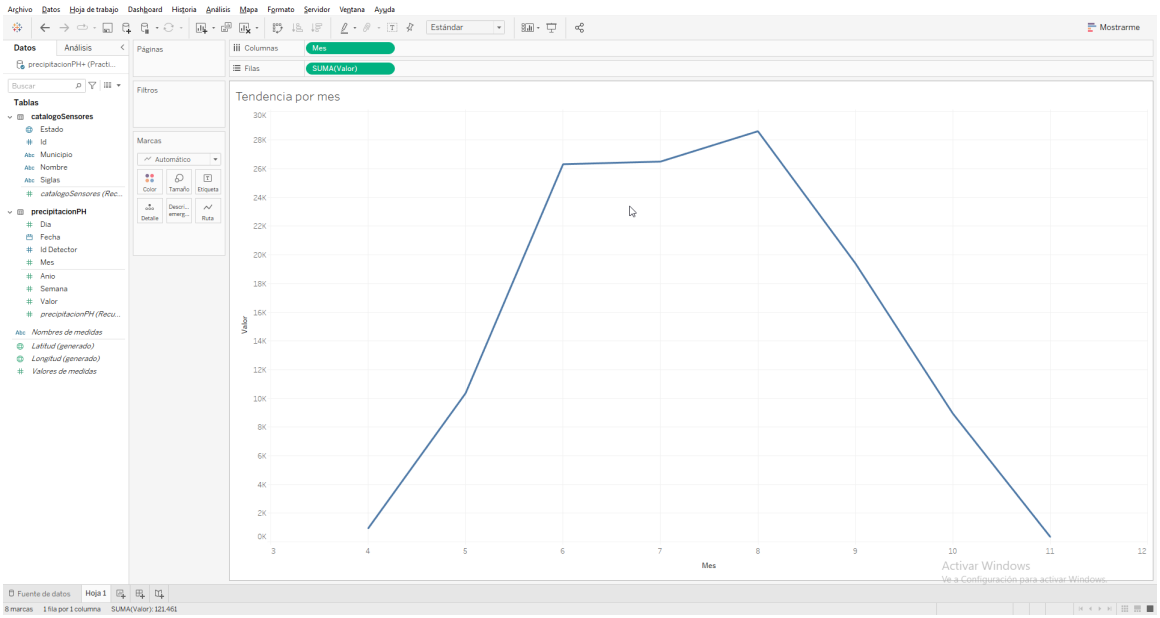

Figura 7: Tendencia precipitación mes en Tableau.

	precipitacion	mes
1	942.69	4
2	10362.86	5
3	26320.35	6
4	26503.71	7
5	28619.28	8
6	19412.63	9
7	8948.15999999999	10
8	351.17	11

Figura 8: Vista de tendencia precipitación mes.

3.2.3. Año

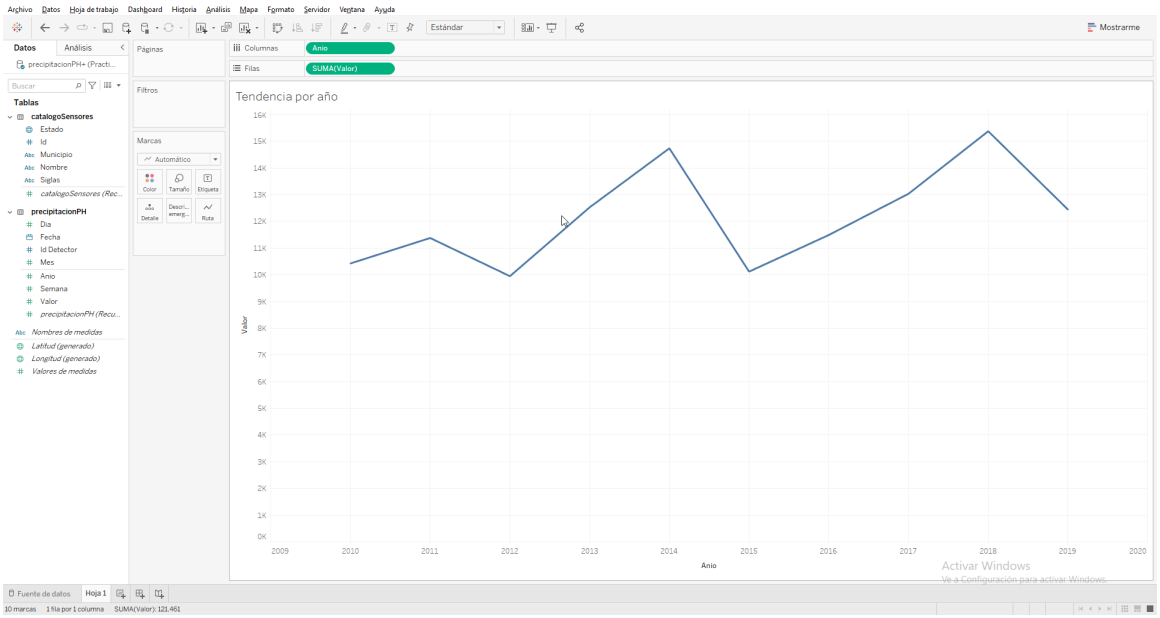

Figura 9: Tendencia precipitación año en Tableau.

	precipitacion	anio
1	10424.6	2010
2	11376.97	2011
3	9946.79	2012
4	12524.19	2013
5	14733.83	2014
6	10118.13	2015
7	11489.15	2016
8	13033.37	2017
9	15373.72	2018
10	12440.1	2019

Figura 10: Vista de tendencia precipitación año.

3.3. Lugares con mayor precipitación durante todo el periodo de estudio (delegación)

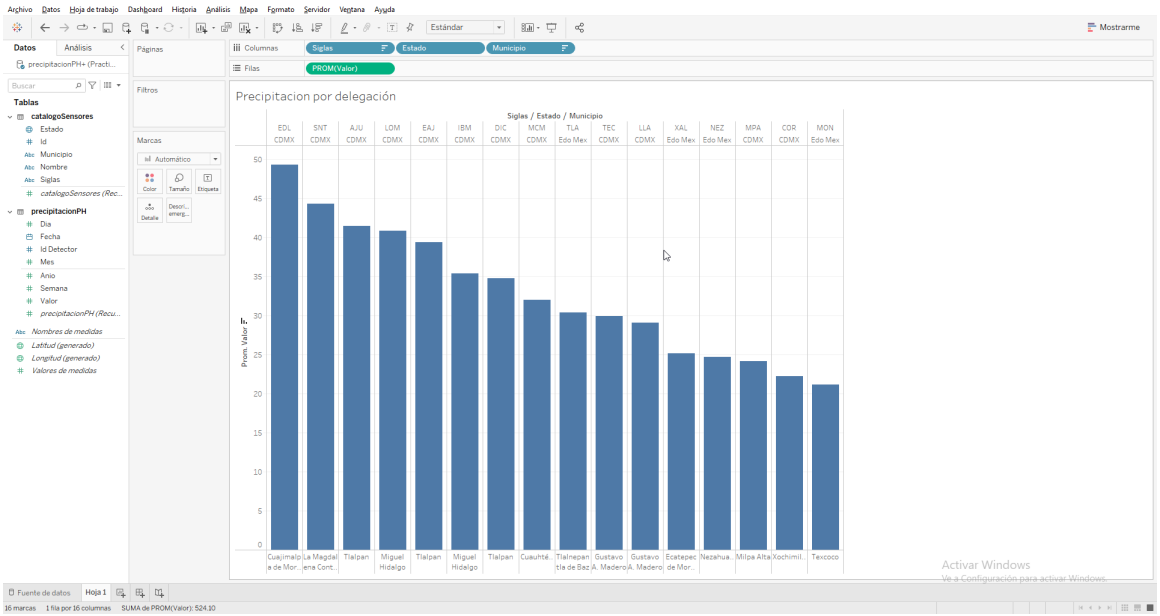

Figura 11: Lugares con mayor precipitación durante todo el periodo de estudio en Tableau.

	precipitacion	detector	nombre	municipio
1	49.2793827160494	EDL	Ex Convento Desierto de los Leones	Cuajimalpa de Morelos
2	44.2960199004975	SNT	San Nicolás Totolapan	La Magdalena Contreras
3	41.4556734693878	AJU	Ajusco	Tlalpan
4	40.7916964285714	LOM	Lomas	Miguel Hidalgo
5	39.3797560975609	EAJ	Ecoguardas Ajusco	Tlalpan
6	35.3612227074236	IBM	Legaria	Miguel Hidalgo
7	34.7892148760331	DIC	Diconsa	Tlalpan
8	31.9897826086956	MCM	Museo de la Ciudad de México	Cuauhtémoc
9	30.3451914893617	TLA	Tlalnepantla	Tlalnepantla de Baz
10	29.9095744680851	TEC	Cerro del Tepeyac	Gustavo A. Madero
11	29.0983116883117	LLA	Laboratorio de Análisis Ambiental	Gustavo A. Madero
12	25.1555042016807	XAL	Xalostoc	Ecatepec de Morelos
13	24.6968691588785	NEZ	Nezahualcóyotl	Nezahualcóyotl
14	24.156835443038	MPA	Milpa Alta	Milpa Alta
15	22.2418222222222	COR	CORENA	Xochimilco
16	21.1502183406114	MON	Montecillo	Texcoco

Figura 12: Vista de lugares con mayor precipitación durante todo el periodo de estudio.

4. Documentación del pseudo-código

Entrada: Archivo de datos `csv`.
Salida: Querys para el sistema gestor de BD.
Tomamos los datos indexados en 0.

1: $semana \leftarrow 1$
2: **for** $i \leftarrow 1, numFilas$ **do** ▷ Ignoramos la primera fila del documento
3: $dia, mes, anio \leftarrow datos[i][0]$ ▷ La fecha siempre se encuentra en la primera columna
4: $fecha \leftarrow anio + mes + dia$ ▷ Armamos la fecha de acuerdo al formato de la BD
5: **for** $j \leftarrow 1, numColumnas$ **do** ▷ Procesamos los datos de cada sensor
6: $valor \leftarrow datos[i][j]$
7: **if** $valor \neq -99$ **then** ▷ Si el valor es distinto de nulo creamos la query.
8: $query \leftarrow j + fecha + valor + dia + mes + anio + semana$

```
        end if
    end for
    semana ← semana + 1
end for
```

4.1. Código fuente usado y capturas de pantallas

```
import csv
import datetime
import calendar

archivos = ['2010PPH.csv', '2011PPH.csv', '2012PPH.csv', '2013PPH.csv',
'2014PPH.csv', '2015PPH.csv', '2016PPH.csv', '2017PPH.csv',
'2018PPH.csv', '2019PPH.csv']
for archivo in archivos:
    with open(archivo, newline='') as File:
        reader = csv.reader(File)
        first = True
        semana = 1
        for row in reader:
            if not first:#Omitimos la descripci n de columna
                dia, mes, ano = row[0].split('/')
                dia, mes, ano= int(dia), int(mes), int(ano)
                fecha = datetime.date(ano, mes, dia)
                for i in range(1,17):#Recorrido de columnas
                    valor = row[i]
                    if valor == '-99':#Si el valor es nulo omitimos ese dato
                        continue
                    #Creaci n de la query
                    query = f"INSERT INTO precipitacionPH values ({i},
                    '{fecha}', {valor}, {dia}, {mes}, {ano}, {semana})"
                    print(query)
                semana += 1
            first = False
```

```
dataset ➤ ls
2010PPH.csv    2013PPH.csv    2016PPH.csv    2019PPH.csv
2011PPH.csv    2014PPH.csv    2017PPH.csv    extraer_csv.py
2012PPH.csv    2015PPH.csv    2018PPH.csv
dataset ➤ python3 extraer_csv.py > querys.sql
dataset ➤ ls
2010PPH.csv    2013PPH.csv    2016PPH.csv    2019PPH.csv
2011PPH.csv    2014PPH.csv    2017PPH.csv    extraer_csv.py
2012PPH.csv    2015PPH.csv    2018PPH.csv    querys.sql
dataset ➤
```

Figura 13: Ejecución del script, podemos ver que el resultado se guarda en el archivo de querys mediante una tubería.

```
  1 INSERT INTO precipitacionPH values (5, '2010-05-03', 5.59, 3, 5, 2010, 18)
  2 INSERT INTO precipitacionPH values (6, '2010-05-03', 7.06, 3, 5, 2010, 18)
  3 INSERT INTO precipitacionPH values (7, '2010-05-03', 17.76, 3, 5, 2010, 18)
  4 INSERT INTO precipitacionPH values (11, '2010-05-03', 9.98, 3, 5, 2010, 18)
  5 INSERT INTO precipitacionPH values (12, '2010-05-03', 0.31, 3, 5, 2010, 18)
  6 INSERT INTO precipitacionPH values (14, '2010-05-03', 2.61, 3, 5, 2010, 18)
  7 INSERT INTO precipitacionPH values (16, '2010-05-03', 0.71, 3, 5, 2010, 18)
  8 INSERT INTO precipitacionPH values (2, '2010-05-10', 7.58, 10, 5, 2010, 19)
  9 INSERT INTO precipitacionPH values (5, '2010-05-10', 5.53, 10, 5, 2010, 19)
 10 INSERT INTO precipitacionPH values (6, '2010-05-10', 5.89, 10, 5, 2010, 19)
 11 INSERT INTO precipitacionPH values (7, '2010-05-10', 53.18, 10, 5, 2010, 19)
 12 INSERT INTO precipitacionPH values (8, '2010-05-10', 10.75, 10, 5, 2010, 19)
 13 INSERT INTO precipitacionPH values (9, '2010-05-10', 5.37, 10, 5, 2010, 19)
 14 INSERT INTO precipitacionPH values (11, '2010-05-10', 17.46, 10, 5, 2010, 19
 15 INSERT INTO precipitacionPH values (12, '2010-05-10', 15.66, 10, 5, 2010, 19
 16 INSERT INTO precipitacionPH values (13, '2010-05-10', 8.48, 10, 5, 2010, 19)
 17 INSERT INTO precipitacionPH values (14, '2010-05-10', 26.44, 10, 5, 2010, 19
 18 INSERT INTO precipitacionPH values (15, '2010-05-10', 2.69, 10, 5, 2010, 19)
 19 INSERT INTO precipitacionPH values (16, '2010-05-10', 8.44, 10, 5, 2010, 19)
 20 INSERT INTO precipitacionPH values (1, '2010-05-17', 17.96, 17, 5, 2010, 20)
 21 INSERT INTO precipitacionPH values (2, '2010-05-17', 20.72, 17, 5, 2010, 20)
 22 INSERT INTO precipitacionPH values (3, '2010-05-17', 1.75, 17, 5, 2010, 20)
 23 INSERT INTO precipitacionPH values (5, '2010-05-17', 1.50, 17, 5, 2010, 20)
 24 INSERT INTO precipitacionPH values (6, '2010-05-17', 2.15, 17, 5, 2010, 20)
 25 INSERT INTO precipitacionPH values (7, '2010-05-17', 6.98, 17, 5, 2010, 20)
 26 INSERT INTO precipitacionPH values (8, '2010-05-17', 4.61, 17, 5, 2010, 20)
"querys.sql" 3704L, 284430B written
```

Figura 14: Contenido del archivo generado por la ejecución del script.

4.2. Modelo de datos

4.2.1. Tabla de hechos principal

```
CREATE TABLE precipitacionPH (
    idDetector int NOT NULL,
    fecha date NOT NULL,
    valor float NOT NULL,
    dia int NOT NULL,
    mes int NOT NULL,
    anio int NOT NULL,
    semana int NOT NULL,
    FOREIGN KEY(idDetector) REFERENCES catalogoSensores(id),
    PRIMARY KEY(idDetector,fecha)
);
```

4.2.2. Catálogos

```
CREATE TABLE precipitacionPH (
    idDetector int NOT NULL,
    fecha date NOT NULL,
    valor float NOT NULL,
    dia int NOT NULL,
    mes int NOT NULL,
    anio int NOT NULL,
    semana int NOT NULL,
    FOREIGN KEY(idDetector) REFERENCES catalogoSensores(id),
    PRIMARY KEY(idDetector,fecha)
);
```

4.2.3. Vistas

```
CREATE VIEW a1
AS
select count(*)as registros from precipitacionPH;
select * from a1;

CREATE VIEW a2
AS
SELECT COUNT(*) as registros,anio from precipitacionPH
group by anio;
select * from a2 order by anio;

create view b
as
select sum(valor) as precipitacion, semana from precipitacionPH
group by semana;
select * from b order by semana;

create view b1
as
select sum(valor) as precipitacion, mes from precipitacionPH
group by mes;
select * from b1 order by mes;

create view b2
as
select sum(valor) as precipitacion, anio from precipitacionPH
group by anio;
select * from b2 order by anio;

create view c
as
select AVG(valor) as precipitacion, siglas as
detector,nombre,municipio from precipitacionPH
INNER join catalogoSensores on idDetector = id
group by nombre,municipio,siglas;
select * from c order by precipitacion desc;
```

5. Conclusiones

- Graciano Herrera Gabriel
  Esta práctica me ayudo a reforzar el proceso de ETL visto en la práctica anterior, fue un proceso similar al de la práctica anterior ya que la BD que utilizamos tenia una estructura un tanto similar, lo que cambiaba era la dimensión del tiempo, pero no fue difícil adaptarla a lo que se nos pedía. Me doy cuenta que el paso más importante es la realización de un análisis previo de la información para poder lograr lo esperado conforme el conocimiento que se busca sobre los datos.

- Meza Zamora Abraham Manuel

Esta práctica me ayudó a comprender mejor el proceso de ETL ya que es necesario hacer un análisis previo de la información que se nos está proporcionando, ya que puede requerir de una reestructuración de la misma, esto con el objetivo de procesar los datos de manera que sea más fácil para nosotros poder interpretarlos.