ETL Project (Magic the Gathering ETL)

Gibran Ahmad   ·   Paul Dean   ·   Soobin Hwang   ·   Seunghwa Jun

---

EXTRACT   (Original Data Sources and Formats)

- **Magic The Gathering (MTG, collectable card games) from Kaggle**
    - Source : https://www.kaggle.com/mylesoneill/magic-the-gathering-cards
    - Format : JSON file ("./AllCards.json")
- **Card Price (War of the Spark)**
    - Source : https://www.mtggoldfish.com/index/XLN_GRN#paper
    - Format : HTML file (pd.read_html)

TRANSFORM   (Data Cleaning or Transformation Process)

- **① Use 'pd.read_json' to create a DataFrame of the cardinfo data**
    - json_file = "./AllCards.json"
    - mtg_df = pd.read_json(json_file)

- **② Tanspose the data, so that the card names go down vertically and into the index**
    - mtg_df = pd.DataFrame(mtg_df.transpose())

- **③ Clean up the dataframe, select and rename columns**
    - clean_mtg_df = mtg_df[["index","convertedManaCost", "text", "type"]]
    - clean_mtg_df = clean_mtg_df.rename(columns={"index":"cardname"})

- **④ Use pd.read_html to create a Dataframe of the price information data**
    - price_url = 'https://www.mtggoldfish.com/index/XLN_GRN#paper'
    - price = pd.read_html(price_url)

- **⑤ Remove the first two rows of the dataframe as they are not relevant**
    - price_table = price[2:].copy()
    - df = pd.DataFrame(price_table[0])

- **⑥ Select columns that are relevant and set ID.**
    - clean_price_df = df[['Card', 'Price']]
    - clean_price_df.index.name = 'id'

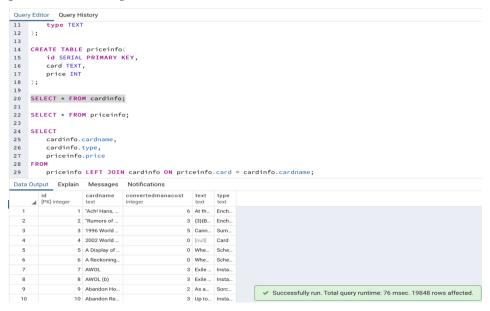## **L**OAD   (Final Database and Tables)

- **Create SQLite database and Push Data inti it**

  - Import sqlalchemy and connecting to DBS
  - Import methods needed to abstract classes into tables
  - Declare column types

- **Set up a connection to the database**

- **Create the tables associated with the classes**

- **Push the pandas dataframe into the SQLite database**

  - clean_mtg_df.to_sql('cardinfo', con=engine, if_exists='replace')
  - clean_price_df.to_sql('priceinfo', con=engine, if_exists='replace')

- **Create another engine for postgres and rename table names**

  - Rename the colums to all lower case and appending cardinfo table
  - Rename columns to all lower case and appending priceinfo table '

- **Joining Queries to do trend analysis and to uncover opportunities for profit**

  - As MTG card players, we want to be able to analyze card price trends based on various metrics, such as converted mana cost and type. Combining these two databases would allow us to do trend analysis on these metrics, potentially allowing us to uncover opportunities for profit. There is a large "second-hand" market for sale and purchase of MTG card singles – we would be able to get a better sense of which cards are under- and over-priced.

Captured Images (Postgres Tables)

[Cardinfo table]


```
Query Editor    Query History
11       type TEXT
12   );
13
14   CREATE TABLE priceinfo(
15       id SERIAL PRIMARY KEY,
16       card TEXT,
17       price INT
18   );
19
20   SELECT * FROM cardinfo;
21
22   SELECT * FROM priceinfo;
23
24   SELECT
25       cardinfo.cardname,
26       cardinfo.type,
27       priceinfo.price
28   FROM
29       priceinfo LEFT JOIN cardinfo ON priceinfo.card = cardinfo.cardname;
```

Data Output   Explain   Messages   Notifications

	id [PK] integer	cardname text	convertedmanacost integer	text text	type text	
1	1	"Ach! Hans, …	6	At th…	Ench…	
2	2	"Rumors of …	3	{3}{B…	Ench…	
3	3	1996 World …	5	Cann…	Sum…	
4	4	2002 World …	0	[null]	Card	
5	5	A Display of …	0	Whe…	Sche…	
6	6	A Reckoning…	0	Whe…	Sche…	
7	7	AWOL	3	Exile …	Insta…	
8	8	AWOL (b)	3	Exile …	Insta…	
9	9	Abandon Ho…	2	As a…	Sorc…	✔ Successfully run. Total query runtime: 76 msec. 19848 rows affected.
10	10	Abandon Re…	3	Up to…	Insta…	

[Priceinfo table]


```
Query Editor    Query History
11       type TEXT
12   );
13
14   CREATE TABLE priceinfo(
15       id SERIAL PRIMARY KEY,
16       card TEXT,
17       price INT
18   );
19
20   SELECT * FROM cardinfo;
21
22   SELECT * FROM priceinfo;
23
24   SELECT
25       cardinfo.cardname,
26       cardinfo.type,
27       priceinfo.price
28   FROM
29       priceinfo LEFT JOIN cardinfo ON priceinfo.card = cardinfo.cardname;
```

Data Output   Explain   Messages   Notifications

	id [PK] integer	card text	price integer	
1	0	Arcli…	21	
2	1	Assa…	9	
3	2	Aggr…	6	
4	3	Fires…	6	
5	4	Vras…	5	
6	5	Ral, …	4	
7	6	Serra…	4	
8	7	Skall…	4	
9	8	Grav…	3	✔ Successfully run. Total query runtime: 77 msec. 367 rows affected.
10	9	Jace,…	3	

[Joining Query]

Query Editor   Query History

```sql
11        type TEXT
12    );
13
14    CREATE TABLE priceinfo(
15        id SERIAL PRIMARY KEY,
16        card TEXT,
17        price INT
18    );
19
20    SELECT * FROM cardinfo;
21
22    SELECT * FROM priceinfo;
23
24    SELECT
25        cardinfo.cardname,
26        cardinfo.type,
27        priceinfo.price
28    FROM
29        priceinfo LEFT JOIN cardinfo ON priceinfo.card = cardinfo.cardname;
```

Data Output   Explain   Messages   Notifications

	cardname text	type text	price integer
1	Admiral Bec…	Lege…	0
2	Admiral's Or…	Insta…	0
3	Aggressive …	Creat…	6
4	Ajani's Influe…	Sorc…	2
5	Ajani's Last …	Ench…	0
6	Ajani, Adver…	Lege…	0
7	Alpine Moon	Ench…	1
8	Amulet of S…	Artif…	0
9	Angrath's M…	Creat…	0
10	Angrath, the …	Lege…	0