Assignment: Building a Data Pipeline in Python

Objective:

Build a simple data pipeline that extracts data from a public API, transforms it for analysis, and loads it into a database.

In [27]:
```
import warnings
import sys
if not sys.warnoptions:
    warnings.simplefilter("ignore")
```

1. Extract Data from an API:

In [29]:
```
import requests
import pandas as pd
```

In [31]:
```
# Step 1: Extract data from the API
api_url = "https://jsonplaceholder.typicode.com/users"
response = requests.get(api_url)
response.raise_for_status()  # Ensure the API request was successful
data = response.json()
```

In [33]:
```
# Convert the data to a pandas DataFrame
df_users = pd.DataFrame(data)
print("Data extracted successfully:")
print(df_users.head())
```

```
Data extracted successfully:
   id              name   username                     email  \
0   1     Leanne Graham       Bret          Sincere@april.biz
1   2      Ervin Howell  Antonette          Shanna@melissa.tv
2   3  Clementine Bauch   Samantha         Nathan@yesenia.net
3   4  Patricia Lebsack   Karianne  Julianne.OConner@kory.org
4   5  Chelsey Dietrich     Kamren   Lucio_Hettinger@annie.ca

                                        address                  phone  \
0  {'street': 'Kulas Light', 'suite': 'Apt. 556',...  1-770-736-8031 x56442
1  {'street': 'Victor Plains', 'suite': 'Suite 87...    010-692-6593 x09125
2  {'street': 'Douglas Extension', 'suite': 'Suit...         1-463-123-4447
3  {'street': 'Hoeger Mall', 'suite': 'Apt. 692',...      493-170-9623 x156
4  {'street': 'Skiles Walks', 'suite': 'Suite 351...          (254)954-1289

         website                                            company
0  hildegard.org  {'name': 'Romaguera-Crona', 'catchPhrase': 'Mu...
1  anastasia.net  {'name': 'Deckow-Crist', 'catchPhrase': 'Proac...
2    ramiro.info  {'name': 'Romaguera-Jacobson', 'catchPhrase': ...
3       kale.biz  {'name': 'Robel-Corkery', 'catchPhrase': 'Mult...
4   demarco.info  {'name': 'Keebler LLC', 'catchPhrase': 'User-c...
```

2. Transform Data:

In [35]:
```python
# Step 2: Transform the data
# Extracting and flattening the address and company columns
df_users['street'] = df_users['address'].apply(lambda x: x['street'])
df_users['city'] = df_users['address'].apply(lambda x: x['city'])
df_users['zipcode'] = df_users['address'].apply(lambda x: x['zipcode'])
df_users['company_name'] = df_users['company'].apply(lambda x: x['name'])

# Selecting the final columns
df_transformed = df_users[['id', 'name', 'email', 'street', 'city', 'zipcode', 'company_name']]

print("Transformed Data:")
print(df_transformed.head())
```

```
Transformed Data:
   id              name                      email             street  \
0   1     Leanne Graham          Sincere@april.biz        Kulas Light
1   2      Ervin Howell          Shanna@melissa.tv      Victor Plains
2   3  Clementine Bauch         Nathan@yesenia.net  Douglas Extension
3   4  Patricia Lebsack  Julianne.OConner@kory.org       Hoeger Mall
4   5  Chelsey Dietrich   Lucio_Hettinger@annie.ca       Skiles Walks

            city     zipcode        company_name
0    Gwenborough  92998-3874     Romaguera-Crona
1    Wisokyburgh  90566-7771        Deckow-Crist
2  McKenziehaven  59590-4157  Romaguera-Jacobson
3    South Elvis  53919-4257       Robel-Corkery
4     Roscoeview       33263         Keebler LLC
```

3. Load Data into a SQLite Database:

In [37]:
```
from sqlalchemy import create_engine

# Step 3: Load the data into SQLite database
engine = create_engine("sqlite:///users_data.db")

# Save the DataFrame to a SQL table
df_transformed.to_sql("users", con=engine, if_exists="replace", index=False)

print("Data loaded into the database successfully!")
```

```
Data loaded into the database successfully!
```

In [39]:
```
# Additional transformation: Extract email domain
df_transformed['email_domain'] = df_transformed['email'].apply(lambda x: x.split('@')[-1])

print("Updated Transformed Data:")
print(df_transformed.head())
```

```
Updated Transformed Data:
   id              name                     email             street  \
0   1     Leanne Graham          Sincere@april.biz        Kulas Light
1   2      Ervin Howell          Shanna@melissa.tv      Victor Plains
2   3  Clementine Bauch         Nathan@yesenia.net  Douglas Extension
3   4  Patricia Lebsack  Julianne.OConner@kory.org        Hoeger Mall
4   5  Chelsey Dietrich   Lucio_Hettinger@annie.ca       Skiles Walks

            city     zipcode        company_name email_domain
0    Gwenborough  92998-3874     Romaguera-Crona    april.biz
1    Wisokyburgh  90566-7771        Deckow-Crist   melissa.tv
2  McKenziehaven  59590-4157  Romaguera-Jacobson  yesenia.net
3    South Elvis  53919-4257       Robel-Corkery     kory.org
4     Roscoeview       33263         Keebler LLC     annie.ca
```

In [ ]: