m08_01_insert_data

May 25, 2021

1 Curso de Python do DS ao DEV

Comunidade DS - Meigarom Lopes

2 Modulo 08 - SQL

3 Inserir dados em Tabela do Banco de Dados

3.1 1.0. Connect to database

[13]:
```
import pandas as pd
from sqlalchemy import create_engine
```

[14]:
```
db = create_engine( 'sqlite:///db_olist.sqlite', echo=False )
conn = db.connect()
```

3.2 2.0. Loading dataset

[16]:
```
# dataset
df_customer = pd.read_csv( 'olist/olist_customers_dataset.csv' )

schema_customer = """
CREATE TABLE customer(
    customer_id                 TEXT,
    customer_unique_id          TEXT,
    customer_zip_code_prefix    INTEGER,
    customer_city               TEXT,
    customer_state              TEXT
)
"""
# crate schema
#conn.execute( schema_customer )

# insert data into table
df_customer.to_sql( 'customer', con=conn, if_exists='append', index=False )
```

[33]:
```
# dataset
df_geolocation = pd.read_csv( 'olist/olist_geolocation_dataset.csv' )

schema_geolocation = """
CREATE TABLE geolocation(
    geolocation_zip_code_prefix    INTEGER,
    geolocation_lat                REAL,
    geolocation_lng                REAL,
    geolocation_city               TEXT,
    geolocation_state              TEXT
)
"""
# crate schema
conn.execute( schema_geolocation )

# insert data into table
df_geolocation.to_sql( 'geolocation', con=conn, if_exists='append', index=False␣
 ↪)
```

[37]:
```
# dataset
df_order_items = pd.read_csv( 'olist/olist_order_items_dataset.csv' )

schema_order_items = """
CREATE TABLE order_items(
    order_id               TEXT,
    order_item_id          INTEGER,
    product_id             TEXT,
    seller_id              TEXT,
    shipping_limit_date    TEXT,
    price                  REAL,
    freight_value          REAL
)
"""
# crate schema
conn.execute( schema_order_items )

# insert data into table
df_order_items.to_sql( 'order_items', con=conn, if_exists='append', index=False␣
 ↪)
```

[41]:
```
# dataset
df_order_payments = pd.read_csv( 'olist/olist_order_payments_dataset.csv' )

schema_order_payments = """
CREATE TABLE order_payments(
    order_id                TEXT,
    payment_sequential      INTEGER,
```

```
    payment_type            TEXT,
    payment_installments    INTEGER,
    payment_value           REAL
)
"""
# crate schema
conn.execute( schema_order_payments )

# insert data into table
df_order_payments.to_sql( 'order_payments', con=conn, if_exists='append',␣
 ↪index=False )
```

[45]:
```
# dataset
df_order_reviews = pd.read_csv( 'olist/olist_order_reviews_dataset.csv' )

schema_order_reviews = """
CREATE TABLE order_reviews(
    review_id                   TEXT,
    order_id                    TEXT,
    review_score                INTEGER,
    review_comment_title        TEXT,
    review_comment_message      TEXT,
    review_creation_date        TEXT,
    review_answer_timestamp     TEXT
)
"""
# crate schema
conn.execute( schema_order_reviews )

# insert data into table
df_order_reviews.to_sql( 'order_reviews', con=conn, if_exists='append',␣
 ↪index=False )
```

[49]:
```
# dataset
df_orders = pd.read_csv( 'olist/olist_orders_dataset.csv' )

schema_orders = """
CREATE TABLE orders(
    order_id                        TEXT,
    customer_id                     TEXT,
    order_status                    TEXT,
    order_purchase_timestamp        TEXT,
    order_approved_at               TEXT,
    order_delivered_carrier_date    TEXT,
    order_delivered_customer_date   TEXT,
    order_estimated_delivery_date   TEXT
)
```

```
"""
# crate schema
conn.execute( schema_orders )

# insert data into table
df_orders.to_sql( 'orders', con=conn, if_exists='append', index=False )
```

[53]:
```
# dataset
df_products = pd.read_csv( 'olist/olist_products_dataset.csv' )

schema_products = """
CREATE TABLE products(
    product_id                  TEXT,
    product_category_name       TEXT,
    product_name_lenght         REAL,
    product_description_lenght  REAL,
    product_photos_qty          REAL,
    product_weight_g            REAL,
    product_length_cm           REAL,
    product_height_cm           REAL,
    product_width_cm            REAL
)
"""
# crate schema
conn.execute( schema_products )

# insert data into table
df_products.to_sql( 'products', con=conn, if_exists='append', index=False )
```

[57]:
```
# dataset
df_sellers = pd.read_csv( 'olist/olist_sellers_dataset.csv' )

schema_sellers = """
CREATE TABLE sellers(
    seller_id               TEXT,
    seller_zip_code_prefix  INTEGER,
    seller_city             TEXT,
    seller_state            TEXT
)
"""
# crate schema
conn.execute( schema_sellers )

# insert data into table
df_sellers.to_sql( 'sellers', con=conn, if_exists='append', index=False )
```

[61]:
```
# dataset
df_product_category_name = pd.read_csv( 'olist/
 ↪product_category_name_translation.csv' )

schema_product_category_name = """
CREATE TABLE product_category_name(
    product_category_name            TEXT,
    product_category_name_english    TEXT
)
"""
# crate schema
conn.execute( schema_product_category_name )

# insert data into table
df_product_category_name.to_sql( 'product_category_name', con=conn,␣
 ↪if_exists='append', index=False )
```

3.3 3.0. Check Database

[62]:
```
query = """
    SELECT name
    FROM sqlite_master
    WHERE type = 'table'
"""

#query = """
#    SELECT *
#    FROM customer
#"""

table = pd.read_sql_query( query, conn )
table
```

[62]:

	name
0	customer
1	geolocation
2	order_items
3	order_payments
4	order_reviews
5	orders
6	products
7	sellers
8	product_category_name