The goal will be to construct an ETL pipeline that loads the data, performs some transformations, and writes the data to a database (and Google Cloud!).

- Create a new pipeline, call it `green_taxi_etl`
- Add a data loader block and use Pandas to read data for the final quarter of 2020 (months `10`, `11`, `12`).
  - You can use the same datatypes and date parsing methods shown in the course.
  - `BONUS`: load the final three months using a for loop and `pd.concat`

Question 1. Data Loading

Once the dataset is loaded, what's the shape of the data?

- **266,855 rows x 20 columns**
- 544,898 rows x 18 columns
- 544,898 rows x 20 columns
- 133,744 rows x 20 columns

	VendorID	lpep_pickup_datetime	lpep_dropoff_datetime	store_and_fwd_flag	RatecodeID
0	2	1601512279000	1601512495000	N	1
1	2	1601512932000	1601513031000	N	1
2	2	1601513589000	1601513739000	N	1
3	1	1601511149000	1601511608000	N	1
4	1	1601512358000	1601512982000	N	1
5	2	1601510690000	1601511214000	N	1
6	2	1601513370000	1601514543000	N	1
7	2	1601513675000	1601514223000	N	1
8	2	1601514005000	1601514476000	N	1
9	1	1601513218000	1601513916000	N	1

266855 rows x 20 columns 27.564s

- Add a transformer block and perform the following:
    - Remove rows where the passenger count is equal to 0 *or* the trip distance is equal to zero.
    - Create a new column `lpep_pickup_date` by converting `lpep_pickup_datetime` to a date.
    - Rename columns in Camel Case to Snake Case, e.g. `VendorID` to `vendor_id`.
    - Add three assertions:
        - `vendor_id` is one of the existing values in the column (currently)
        - `passenger_count` is greater than 0
        - `trip_distance` is greater than 0

Question 2. Data Transformation

Upon filtering the dataset where the passenger count is greater than 0 *and* the trip distance is greater than zero, how many rows are left?

- 544,897 rows
- 266,855 rows
- **139,370 rows**
- 266,856 rows

PY TRANSFORMER transform_data 1 parent

	VendorID	lpep_pickup_datetime	lpep_dropoff_datetime	store_and_fwd_flag	RatecodeID
0	2	1601512279000	1601512495000	N	1
1	2	1601512932000	1601513031000	N	1
2	2	1601513589000	1601513739000	N	1
3	1	1601511149000	1601511608000	N	1
4	1	1601512358000	1601512982000	N	1
5	2	1601510690000	1601511214000	N	1
6	2	1601513370000	1601514543000	N	1
7	2	1601513675000	1601514223000	N	1
8	2	1601514005000	1601514476000	N	1
9	1	1601513218000	1601513916000	N	1

139370 rows x 21 columns 3.397s

Question 3. Data Transformation

Which of the following creates a new column `lpep_pickup_date` by converting `lpep_pickup_datetime` to a date?

- `data = data['lpep_pickup_datetime'].date`
- `data('lpep_pickup_date') = data['lpep_pickup_datetime'].date`
- **`data['lpep_pickup_date'] = data['lpep_pickup_datetime'].dt.date`**
- `data['lpep_pickup_date'] = data['lpep_pickup_datetime'].dt().date()`

```python
@transformer
def transform(data, *args, **kwargs):

    df = data[data['passenger_count'] > 0]
    df = df[df['trip_distance'] > 0]
    df['lpep_pickup_date'] = df['lpep_pickup_datetime'].dt.date

    return df
```

Question 4. Data Transformation

What are the existing values of `VendorID` in the dataset?

- 1, 2, or 3
- **1 or 2**
- 1, 2, 3, 4
- 1

PY TRANSFORMER profound_meadow 1 parent

```python
    df = df[df['trip_distance'] > 0]
    df['lpep_pickup_date'] = df['lpep_pickup_datetime'].dt.date
    df.columns = (df.columns.str.replace('(?⇐[a-z])(?=[A-Z])', '_', regex=True).s
    df_list_unique =set(df['vendor_id'])
    return df_list_unique


@test
def test_output(output, *args) -> None:
    """
    Template code for testing the output of the block.
    """
    assert output is not None, 'The output is undefined'
```

```
1/1 tests passed.
```

```
Variable output_0 (no type) for block profound_meadow in pipeline green_taxi_etl
stored in /home/src/mage_data/**************/pipelines/green_taxi_etl/.variable
s/profound_meadow/output_0


[1, 2]
```

Question 5. Data Transformation

How many columns need to be renamed to snake case?

- 3
- 6
- 2
- **4**

```
1/1 tests passed.

VendorID
lpep_pickup_datetime
lpep_dropoff_datetime
store_and_fwd_flag
RatecodeID
PULocationID
DOLocationID
passenger_count
trip_distance
fare_amount
extra
mta_tax
tip_amount
tolls_amount
ehail_fee
improvement_surcharge
total_amount
payment_type
trip_type
congestion_surcharge
```

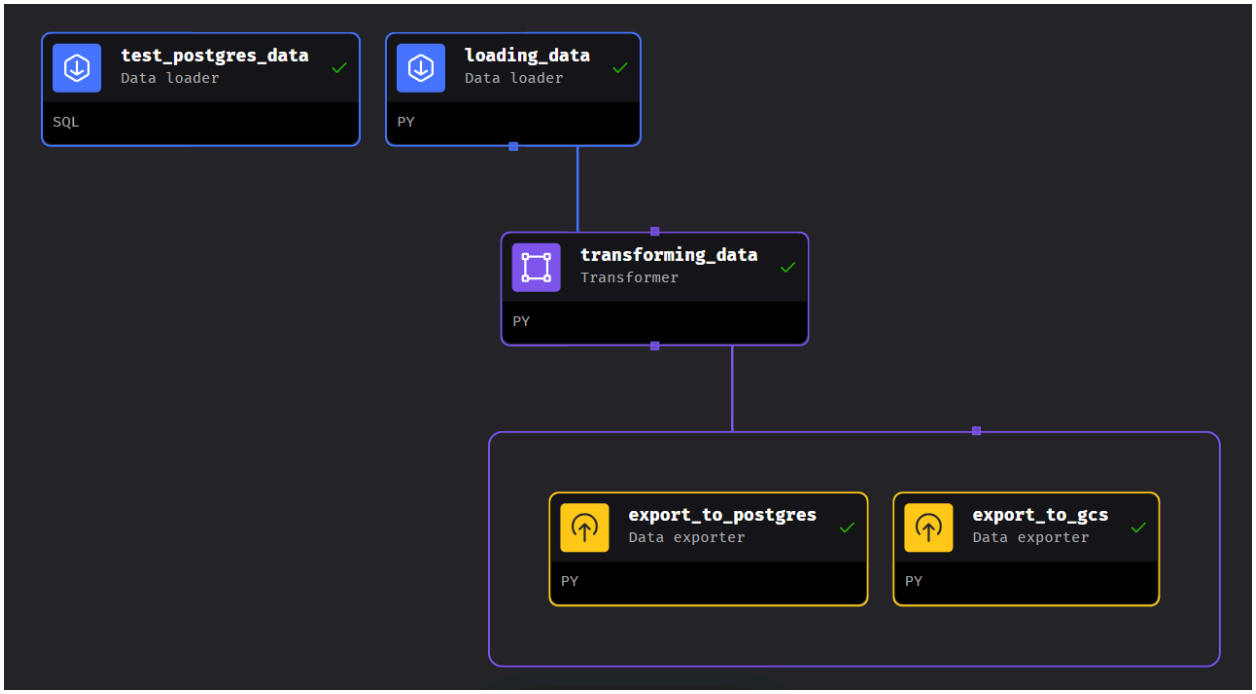
Question 6. Data Exporting

Once exported, how many partitions (folders) are present in Google Cloud?

- **96**
- 56
- 67
- 108

lpep_pickup_date=2020-12-13/	–	Folder	–	–	–	–	–	–	
lpep_pickup_date=2020-12-14/	–	Folder	–	–	–	–	–	–	
lpep_pickup_date=2020-12-15/	–	Folder	–	–	–	–	–	–	
lpep_pickup_date=2020-12-16/	–	Folder	–	–	–	–	–	–	
lpep_pickup_date=2020-12-17/	–	Folder	–	–	–	–	–	–	
lpep_pickup_date=2020-12-18/	–	Folder	–	–	–	–	–	–	
lpep_pickup_date=2020-12-19/	–	Folder	–	–	–	–	–	–	
lpep_pickup_date=2020-12-20/	–	Folder	–	–	–	–	–	–	
lpep_pickup_date=2020-12-21/	–	Folder	–	–	–	–	–	–	
lpep_pickup_date=2020-12-22/	–	Folder	–	–	–	–	–	–	
lpep_pickup_date=2020-12-23/	–	Folder	–	–	–	–	–	–	
lpep_pickup_date=2020-12-24/	–	Folder	–	–	–	–	–	–	
lpep_pickup_date=2020-12-25/	–	Folder	–	–	–	–	–	–	
lpep_pickup_date=2020-12-26/	–	Folder	–	–	–	–	–	–	
lpep_pickup_date=2020-12-27/	–	Folder	–	–	–	–	–	–	
lpep_pickup_date=2020-12-28/	–	Folder	–	–	–	–	–	–	
lpep_pickup_date=2020-12-29/	–	Folder	–	–	–	–	–	–	
lpep_pickup_date=2020-12-30/	–	Folder	–	–	–	–	–	–	
lpep_pickup_date=2020-12-31/	–	Folder	–	–	–	–	–	–	
lpep_pickup_date=2021-01-01/	–	Folder	–	–	–	–	–	–	

Rows per page: 100 1 – 95 of 95


test_postgres_data
Data loader
SQL
loading_data
Data loader
PY
transforming_data
Transformer
PY
export_to_postgres
Data exporter
PY
export_to_gcs
Data exporter
PY