

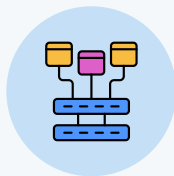
WEEKLY ASSIGNMENT #5

CREATE FLOW DATA FROM
ARTIFICIAL BANKSIM DATASETS

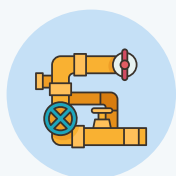


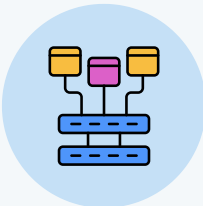
CONTENT

1. Data Model

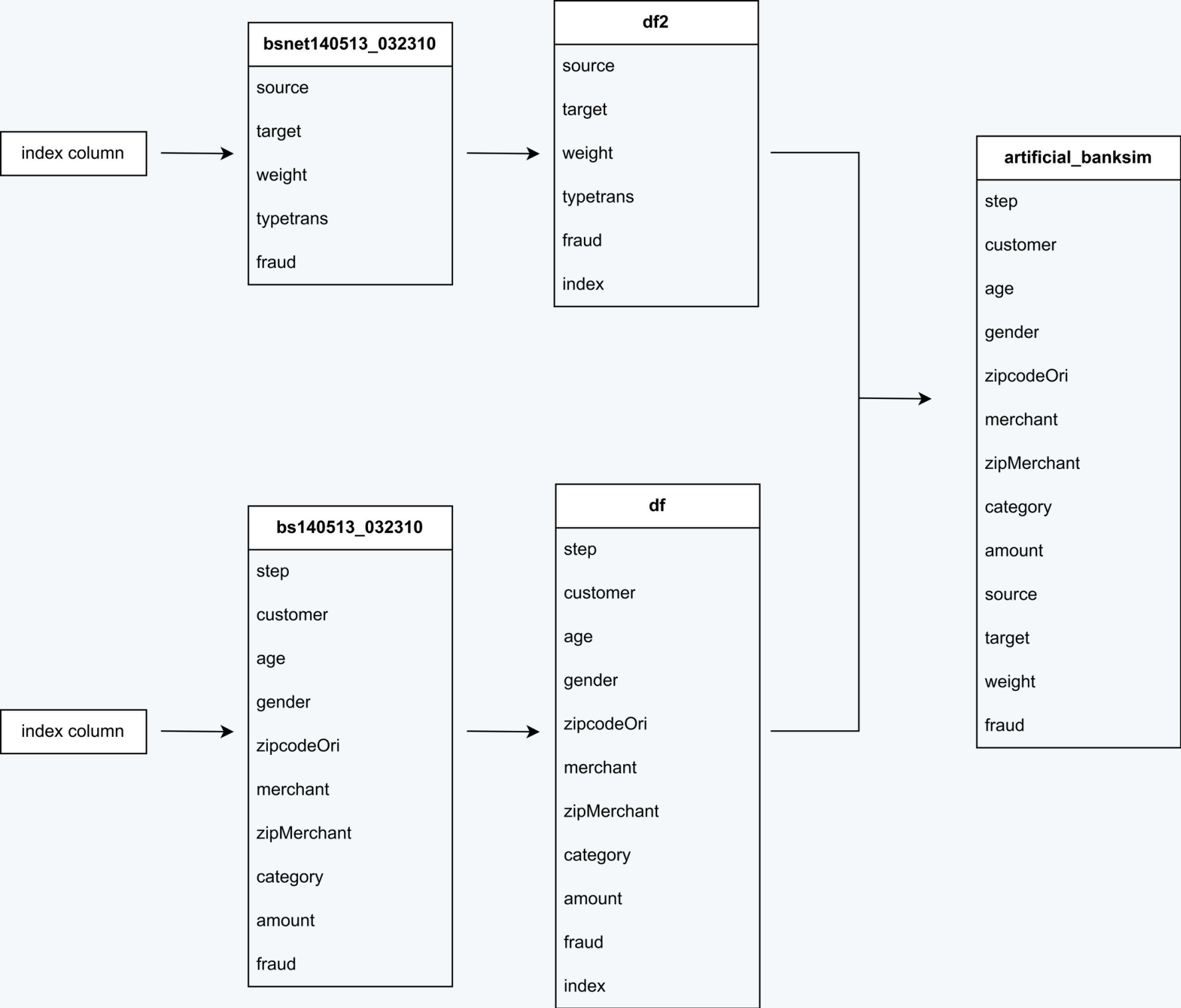


2. Data Pipeline

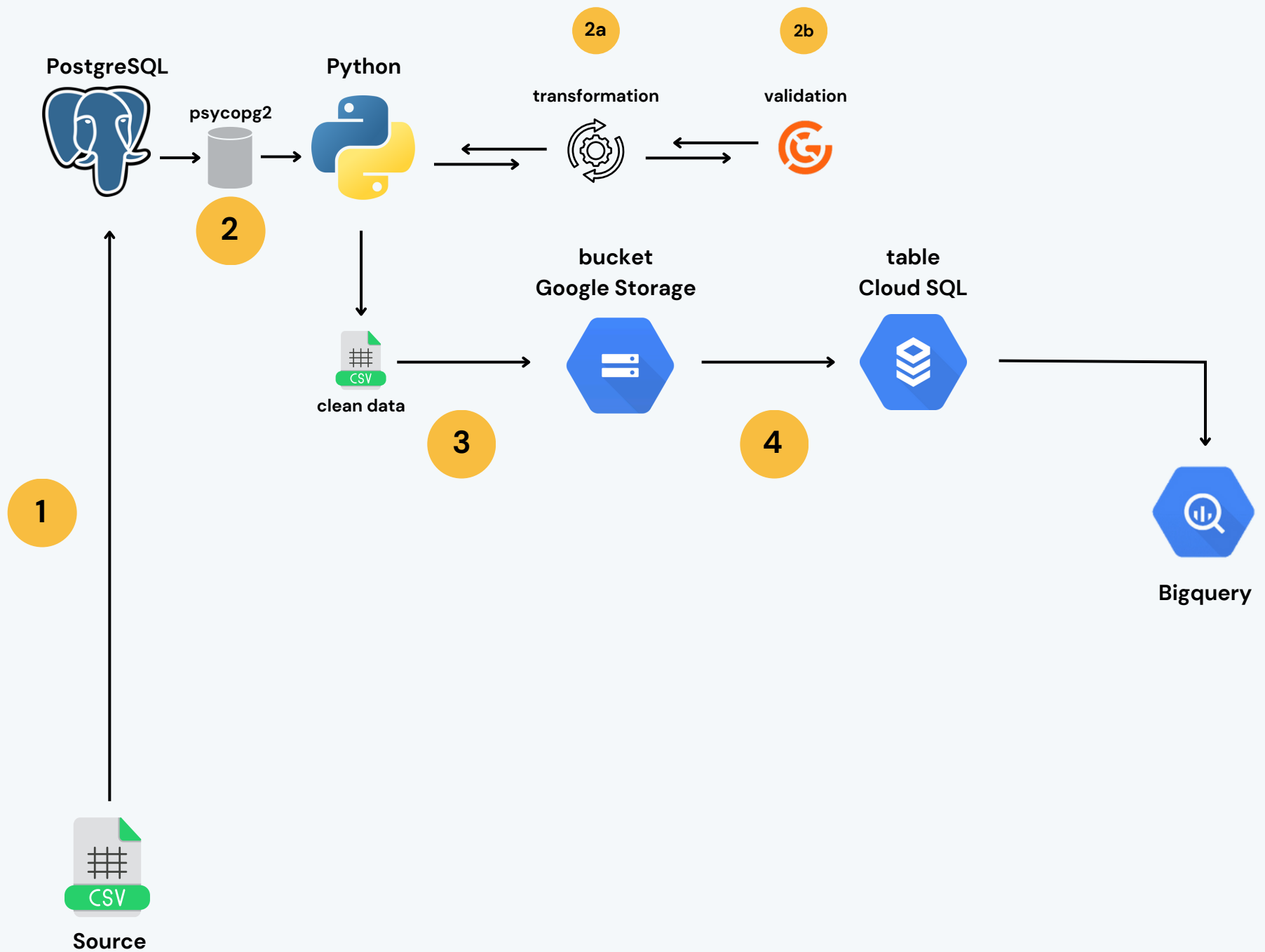
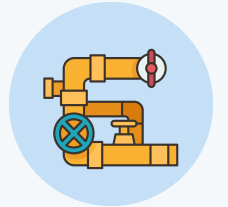




Data Model



Data Pipeline



Data Pipeline Explanation

1. The **CSV** data source is **loaded into PostgreSQL**.
2. Using **psycopg2** we can establish **connections to postgresQL**, execute queries, manage transactions, retrieve and store data, and handle exceptions **through Python**.
 - a. Based on the data, we have to carry out **transformations** so that the resulting data is good. The transformations used include:
 - **Join df1 and df2** based on index column.
 - **Drop one of the two columns** that have the **same value** (fraud in df1 and df2 as well as category and typetrans)
 - **Remove the character** (') in the customer, age, gender, zipcodeori, merchant, zipmerchant, category, source, target, and typetrans columns.
 - **Changed the inappropriate value** in the age column in df1 to '0'. That way the data type can be changed to integer.
 - **Changed the data type** of the zipcodeori zipmerchant column in df1 to integer.

Documentation:

```
Server Operations Using Python's Psycopg2

import pandas as pd
import psycopg2

CONNECT_DB = "host=localhost port=5432 dbname=datafellowship12 user=datafellowship12 password=datafellowship12"

(1) ✓ 12s Python
```

```
# join df and df2 based on index column
merged_df = df.merge(df2, on='index')

✓ 0.2s
```

Data Pipeline Explanation

b. **Data validation** was carried out **using Great Expectations** to ensure data quality before proceeding to the next process. The assumptions used are:

- The value in the customer column and source column cannot be NaN.
- The value in the age column cannot be '0', so we **drop** the data **age='0'** (the value dropped is **only 0.61%**).
- In real life, gender only exists F and M. In the gender column df1, values other than F and M are found. But we try to hold them first.
- Even though there are still values in the weight column in df2 that are '0', but we try to hold them first.

Documentation:

```
# create expectations
validator.expect_column_values_to_not_be_null("customer")
✓ 0.0s

{
  "success": true,
  "result": {
    "element_count": 594643,
    "unexpected_count": 0,
    "unexpected_percent": 0.0,
    "unexpected_percent_total": 0.0,
    "partial_unexpected_list": []
  },
  "meta": {},
  "exception_info": {
    "raised_exception": false,
    "exception_traceback": null,
    "exception_message": null
  }
}
```

```
# Count the occurrences of '0' in the 'age' column
count_zero = merged_df['age'].value_counts().get(0, 0)

# Count the total number of rows in the DataFrame
total_rows = merged_df.shape[0]

# Calculate the percentage of '0' values in the 'age' column
percentage_zero = (count_zero / total_rows) * 100

print("Percentage of '0' values in the 'age' column: {:.2f}%".format(percentage_zero))
✓ 0.0s

Percentage of '0' values in the 'age' column: 0.61%

# Drop rows where 'age' column has value 0
merged_df = merged_df[merged_df['age'] != 0]
✓ 0.0s
```

Data Pipeline Explanation

- The transformation data in python is saved to CSV. This file is then uploaded to a bucket on Google Storage.
- In Cloud SQL, we create a table first. Then import the data in the bucket that was created previously.

Documentation:

```
C:\Users\ACER>gsutil mb gs://weekly_assignment_1
Creating gs://weekly_assignment_1/...

C:\Users\ACER>gsutil ls
gs://datafellowship12/
gs://df12/
gs://df12_testing_bucket_python_client/
gs://weekly_assignment1/
gs://weekly_assignment_1/
```

3

```
C:\Users\ACER>gsutil ls gs://weekly_assignment_1/
gs://weekly_assignment_1/artificial_banksim.csv
```

```
C:\Users\ACER>gsutil cp "C:\Folder Aisyah\Weekly Assignment - Week 1\artificial_banksim.csv" gs://weekly_assignment_1/
Copying file://C:\Folder Aisyah\Weekly Assignment - Week 1\artificial_banksim.csv [Content-Type=application/vnd.ms-excel]...
| [1 files][ 57.0 MiB/ 57.0 MiB]
Operation completed over 1 objects/57.0 MiB.
```

Source

Choose a file to import from. Make sure you have read access first. [Learn more](#)

bucket-name/file-name *

☒ weekly_assignment_1/artificial_banksim.csv

BROWSE

Browse for a Cloud Storage file or enter the path to one (bucket/folder/file)

File format

☐ SQL

A plain text file with a sequence of SQL commands, like the output of pg_dump

☒ CSV

If your Cloud Storage file is a CSV file, select CSV. The CSV file should be a plain text file with one line per row and comma-separated fields.

Destination

Choose the database and table in your instance for this file to import into. [Learn more](#)

Database *

weekly_assignment1

Table *

artificial_banksim

Enter the name of an existing table in the database to house your CSV file

SHOW USER OPTIONS

When you import, a Cloud SQL service account will be granted read access to the selected file and bucket, which will be reflected in your permissions.

IMPORT

CANCEL

4

```
weekly_assignment1=> select * from artificial_banksim limit 10;
```

step	customer	age	gender	zipcodeori	merchant	zipmerchant	category	amount	source	target	weight	fraud
0	C1093826151	4	M	28007	M348934600	28007	es_transportation	4.55	C1093826151	M348934600	4.55	False
0	C352968107	2	M	28007	M348934600	28007	es_transportation	39.68	C352968107	M348934600	39.68	False
0	C2054744914	4	F	28007	M1823072687	28007	es_transportation	26.89	C2054744914	M1823072687	26.89	False
0	C1760612790	3	M	28007	M348934600	28007	es_transportation	17.25	C1760612790	M348934600	17.25	False
0	C757503768	5	M	28007	M348934600	28007	es_transportation	35.72	C757503768	M348934600	35.72	False
0	C1315400589	3	F	28007	M348934600	28007	es_transportation	25.81	C1315400589	M348934600	25.81	False
0	C765155274	1	F	28007	M348934600	28007	es_transportation	9.1	C765155274	M348934600	9.1	False
0	C202531238	4	F	28007	M348934600	28007	es_transportation	21.17	C202531238	M348934600	21.17	False
0	C105845174	3	M	28007	M348934600	28007	es_transportation	32.4	C105845174	M348934600	32.4	False

(10 rows)

Data Pipeline Explanation

5. Load CSV to Bigquery

Documentation:

5

df-12-419104

weekly_assignment

week1

SHOW MORE

SHOW MORE

SUMMARY

week1

df-12-419104 weekly_assignment

Last modified Apr 9, 2024, 11:51:00 AM UTC+7

Data US

location

Field name	Type	Mode	Key	Collation	Default Value	Policy Tags
step	INTEGER	NULLABLE	-	-	-	-
customer	STRING	NULLABLE	-	-	-	-
age	INTEGER	NULLABLE	-	-	-	-
gender	STRING	NULLABLE	-	-	-	-
zipcodeori	INTEGER	NULLABLE	-	-	-	-
merchant	STRING	NULLABLE	-	-	-	-
zipmerchant	INTEGER	NULLABLE	-	-	-	-
category	STRING	NULLABLE	-	-	-	-
amount	FLOAT	NULLABLE	-	-	-	-
source	STRING	NULLABLE	-	-	-	-
target	STRING	NULLABLE	-	-	-	-
weight	FLOAT	NULLABLE	-	-	-	-
fraud	BOOLEAN	NULLABLE	-	-	-	-

EDIT SCHEMA

VIEW ROW ACCESS POLICIES

for more:
<https://github.com/aisyahputami/week5>

Thank You!

