

MOOTOD PROJECT



WITH



END-TO-END

DATA PLATFORM



ມູຕົດ ຕິດ ພັບ

MOOTOD

Presented by Gang Look Moo

MEMBERS



SENSE

6420412001



SARAN

6410412010



FAH

6420412011



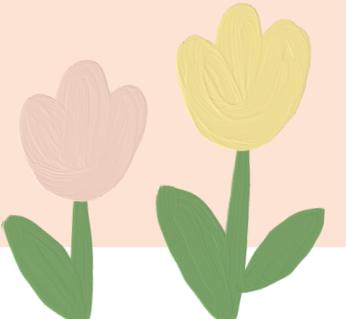
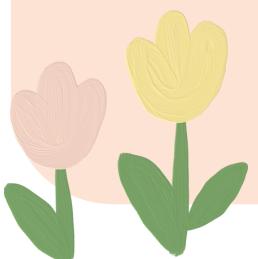
MINNIE

6420412013

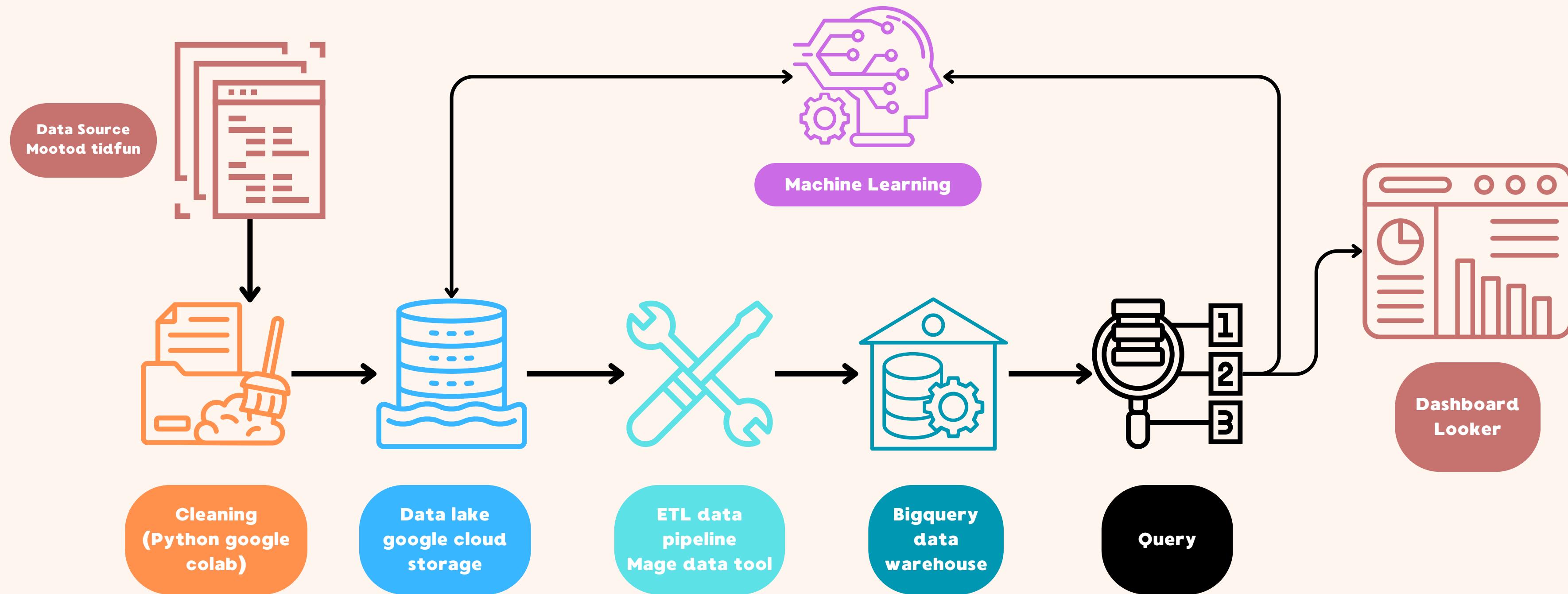


BOON

6420412014



PROJECT OVERVIEW





Data Source Mootod tidfun



Cleaning (Python google colab)



Data lake google cloud storage



ETL data pipeline Mage data tool



Bigquery data warehouse



Query



Dashboard Looker



Machine Learning

Data Source



**Data Source
Mootod tidfun**

ref_no	order	time	cashier_name	bill_type	pickup_type	payment_method	balance	discount	taxes_svc	payment	tip	change	cashback
3005535	1	27/8/2022 20:00	นุน	ท่านที่ร้าน	-	โอนเงิน	฿248	฿0	฿0	฿248	฿0	฿0	฿0
3005534	2	27/8/2022 19:45	นุน	ท่านที่ร้าน	-	โอนเงิน	฿87	฿0	฿0	฿87	฿0	฿0	฿0
3005533	3	27/8/2022 19:35	นุน	ท่านที่ร้าน	-	โอนเงิน	฿239	฿0	฿0	฿239	฿0	฿0	฿0
3005532	4	27/8/2022 19:22	นุน	ท่านที่ร้าน	-	เงินสด	฿464	฿0	฿0	฿1,000	฿0	฿536	฿0
3005531	5	27/8/2022 19:04	นุน	ท่านที่ร้าน	-	โอนเงิน	฿75	฿0	฿0	฿75	฿0	฿0	฿0
3005530	6	27/8/2022 18:46	นุน	ท่านที่ร้าน	-	โอนเงิน	฿304	฿0	฿0	฿304	฿0	฿0	฿0
3005529	7	27/8/2022 18:39	นุน	สั่งกลับบ้าน	Line Man	อีนๆ	฿279	฿0	฿0	฿279	฿0	฿0	฿0
3005528	8	27/8/2022 18:35	นุน	ท่านที่ร้าน	-	โอนเงิน	฿75	฿0	฿0	฿75	฿0	฿0	฿0
3005527	9	27/8/2022 18:34	นุน	สั่งกลับบ้าน	ลูกค้ารับเองที่ร้าน	เงินสด	฿30	฿0	฿0	฿30	฿0	฿0	฿0
3005526	10	27/8/2022 18:34	นุน	สั่งกลับบ้าน	ลูกค้ารับเองที่ร้าน	เงินสด	฿130	฿0	฿0	฿200	฿0	฿70	฿0
3005525	11	27/8/2022 18:31	นุน	ท่านที่ร้าน	-	เงินสด	฿97	฿0	฿0	฿100	฿0	฿3	฿0
3005524	12	27/8/2022 18:31	นุน	สั่งกลับบ้าน	ลูกค้ารับเองที่ร้าน	โอนเงิน	฿55	฿0	฿0	฿55	฿0	฿0	฿0
3005523	13	27/8/2022 18:27	นุน	สั่งกลับบ้าน	ลูกค้ารับเองที่ร้าน	โอนเงิน	฿65	฿0	฿0	฿65	฿0	฿0	฿0
3005522	14	27/8/2022 18:03	นุน	สั่งกลับบ้าน	Line Man	อีนๆ	฿135	-฿15	฿0	฿135	฿0	฿0	฿0
3005521	15	27/8/2022 17:58	นุน	สั่งกลับบ้าน	GrabFood	อีนๆ	฿469	฿0	฿0	฿469	฿0	฿0	฿0
3005520	16	27/8/2022 17:34	นุน	ท่านที่ร้าน	-	โอนเงิน	฿326	฿0	฿0	฿326	฿0	฿0	฿0
3005519	17	27/8/2022 17:23	นุน	สั่งกลับบ้าน	Line Man	อีนๆ	฿75	฿0	฿0	฿75	฿0	฿0	฿0
3005518	18	27/8/2022 17:03	นุน	สั่งกลับบ้าน	Line Man	อีนๆ	฿135	-฿15	฿0	฿135	฿0	฿0	฿0
3005517	19	27/8/2022 16:59	นุน	สั่งกลับบ้าน	Line Man	อีนๆ	฿420	฿0	฿0	฿420	฿0	฿0	฿0
3005516	20	27/8/2022 15:55	นุน	สั่งกลับบ้าน	GrabFood	อีนๆ	฿349	฿0	฿0	฿349	฿0	฿0	฿0
3005515	21	27/8/2022 13:12	นุน	สั่งกลับบ้าน	Line Man	อีนๆ	฿160	-฿15	฿0	฿160	฿0	฿0	฿0
3005514	22	27/8/2022 13:01	นุน	ท่านที่ร้าน	-	โอนเงิน	฿414	฿0	฿0	฿414	฿0	฿0	฿0
3005513	23	27/8/2022 12:39	นุน	สั่งกลับบ้าน	Line Man	อีนๆ	฿175	฿0	฿0	฿175	฿0	฿0	฿0
3005512	24	27/8/2022 12:34	นุน	สั่งกลับบ้าน	Line Man	อีนๆ	฿425	฿0	฿0	฿425	฿0	฿0	฿0
3005511	25	27/8/2022 12:27	นุน	สั่งกลับบ้าน	ลูกค้ารับเองที่ร้าน	โอนเงิน	฿215	฿0	฿0	฿215	฿0	฿0	฿0
3005510	26	27/8/2022 12:18	นุน	สั่งกลับบ้าน	GrabFood	อีนๆ	฿175	฿0	฿0	฿175	฿0	฿0	฿0
3005509	27	27/8/2022 11:44	นุน	สั่งกลับบ้าน	Line Man	อีนๆ	฿90	฿0	฿0	฿90	฿0	฿0	฿0



Data Source Mootod tidfun



Cleaning (Python google colab)



Data lake google cloud storage



ETL data pipeline Mage data tool



Bigquery data warehouse



Query



Dashboard Looker



Machine Learning

Cleaning

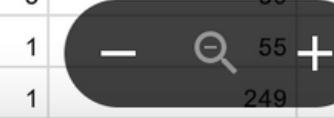
ref_no	order	time	cashier_name	bill_type	pickup_type	payment_method	balance	discount	taxes_svc	payment	tip	change	cashback
3005535	1	27/8/2022 20:00	นุน	ท่านที่ร้าน	-	โอนเงิน	B248	B0	B0	B248	B0	B0	B0
3005534	2	27/8/2022 19:45	นุน	ท่านที่ร้าน	-	โอนเงิน	B87	B0	B0	B87	B0	B0	B0
3005533	3	27/8/2022 19:35	นุน	ท่านที่ร้าน	-	โอนเงิน	B239	B0	B0	B239	B0	B0	B0
3005532	4	27/8/2022 19:22	นุน	ท่านที่ร้าน	-	เงินสด	B464	B0	B0	B1,000	B0	B536	B0
3005531	5	27/8/2022 19:04	นุน	ท่านที่ร้าน	-	โอนเงิน	B75	B0	B0	B75	B0	B0	B0
3005530	6	27/8/2022 18:46	นุน	ท่านที่ร้าน	-	โอนเงิน	B304	B0	B0	B304	B0	B0	B0
3005529	7	27/8/2022 18:39	นุน	สั่งกลับบ้าน	Line Man	อีนๆ	B279	B0	B0	B279	B0	B0	B0
3005528	8	27/8/2022 18:35	นุน	ท่านที่ร้าน	-	โอนเงิน	B75	B0	B0	B75	B0	B0	B0
3005527	9	27/8/2022 18:34	นุน	สั่งกลับบ้าน	ลูกค้ารับเองที่ร้าน	เงินสด	B30	B0	B0	B30	B0	B0	B0
3005526	10	27/8/2022 18:34	นุน	สั่งกลับบ้าน	ลูกค้ารับเองที่ร้าน	เงินสด	B130	B0	B0	B200	B0	B70	B0
3005525	11	27/8/2022 18:31	นุน	ท่านที่ร้าน	-	เงินสด	B97	B0	B0	B100	B0	B3	B0
3005524	12	27/8/2022 18:31	นุน	สั่งกลับบ้าน	ลูกค้ารับเองที่ร้าน	โอนเงิน	B55	B0	B0	B55	B0	B0	B0
3005523	13	27/8/2022 18:27	นุน	สั่งกลับบ้าน	ลูกค้ารับเองที่ร้าน	โอนเงิน	B65	B0	B0	B65	B0	B0	B0
3005522	14	27/8/2022 18:03	นุน	สั่งกลับบ้าน	Line Man	อีนๆ	B135	-B15	B0	B135	B0	B0	B0
3005521	15	27/8/2022 17:58	นุน	สั่งกลับบ้าน	GrabFood	อีนๆ	B469	B0	B0	B469	B0	B0	B0
3005520	16	27/8/2022 17:34	นุน	ท่านที่ร้าน	-	โอนเงิน	B326	B0	B0	B326	B0	B0	B0
3005519	17	27/8/2022 17:23	นุน	สั่งกลับบ้าน	Line Man	อีนๆ	B75	B0	B0	B75	B0	B0	B0
3005518	18	27/8/2022 17:03	นุน	สั่งกลับบ้าน	Line Man	อีนๆ	B135	-B15	B0	B135	B0	B0	B0
3005517	19	27/8/2022 16:59	นุน	สั่งกลับบ้าน	Line Man	อีนๆ	B420	B0	B0	B420	B0	B0	B0
3005516	20	27/8/2022 15:55	นุน	สั่งกลับบ้าน	GrabFood	อีนๆ	B349	B0	B0	B349	B0	B0	B0
3005515	21	27/8/2022 13:12	นุน	สั่งกลับบ้าน	Line Man	อีนๆ	B160	-B15	B0	B160	B0	B0	B0
3005514	22	27/8/2022 13:01	นุน	ท่านที่ร้าน	-	โอนเงิน	B414	B0	B0	B414	B0	B0	B0
3005513	23	27/8/2022 12:39	นุน	สั่งกลับบ้าน	Line Man	อีนๆ	B175	B0	B0	B175	B0	B0	B0
3005512	24	27/8/2022 12:34	นุน	สั่งกลับบ้าน	Line Man	อีนๆ	B425	B0	B0	B425	B0	B0	B0
3005511	25	27/8/2022 12:27	นุน	สั่งกลับบ้าน	ลูกค้ารับเองที่ร้าน	โอนเงิน	B215	B0	B0	B215	B0	B0	B0
3005510	26	27/8/2022 12:18	นุน	สั่งกลับบ้าน	GrabFood	อีนๆ	B175	B0	B0	B175	B0	B0	B0
3005509	27	27/8/2022 11:44	นุน	สั่งกลับบ้าน	Line Man	อีนๆ	B90	B0	B0	B90	B0	B0	B0



Cleaning



ref_no	time	vendor	bill_type	pickup_type	payment_method	sale	payment	tip	change
3001012	7/3/2022 11:39		2	2	7	3	400	400	0
3001013	7/3/2022 11:43		2	2	4	3	300	300	0
3001014	7/3/2022 11:52		2	2	2	2	65	100	0
3001015	7/3/2022 11:56		2	2	6	3	180	180	0
3001016	7/3/2022 11:56		2	2	6	3	180	180	0
3001017	7/3/2022 12:20		2	1	1	1	75	75	0
3001018	7/3/2022 13:27		2	2	7	3	325	325	0
3001019	7/3/2022 14:10		2	2	4	3	75	75	0
3001020	7/3/2022 15:12		2	2	7	3	80	80	0
3001021	7/3/2022 15:16		2	2	3	3	150	150	0
3001022	7/3/2022 15:21		2	2	2	2	65	65	0
3001023	7/3/2022 15:40		2	2	3	3	423	423	0
3001024	7/3/2022 15:57		2	2	2	1	165	165	0
3001025	7/3/2022 16:35		2	2	2	2	65	65	0
3001026	7/3/2022 18:08		2	2	2	2	205	210	5
3001027	7/3/2022 18:22		2	2	4	3	190	190	0
3001028	7/3/2022 19:03		2	2	6	3	225	225	0
3001029	7/3/2022 20:07		2	1	1	1	445	445	0
3001030	7/3/2022 21:05		2	2	4	3	469	469	0
3001031	7/3/2022 21:23		2	1	1	2	237	500	263
3001032	8/3/2022 11:18		2	2	2	2	65	100	0
3001033	8/3/2022 11:27		2	2	3	3	165	165	0
3001034	8/3/2022 11:30		2	2	2	2	55	100	45
3001035	8/3/2022 12:09		2	2	4	3	315	315	0
3001036	8/3/2022 12:28		2	2	7	3	80	80	0
3001037	8/3/2022 12:32		2	2	2	1	55	0	0
3001038	8/3/2022 12:40		2	1	1	1	249	249	0





Data Source Mootod tidfun



Cleaning (Python google colab)



Data lake google cloud storage



ETL data pipeline Mage data tool



Bigquery data warehouse



Query



Dashboard Looker



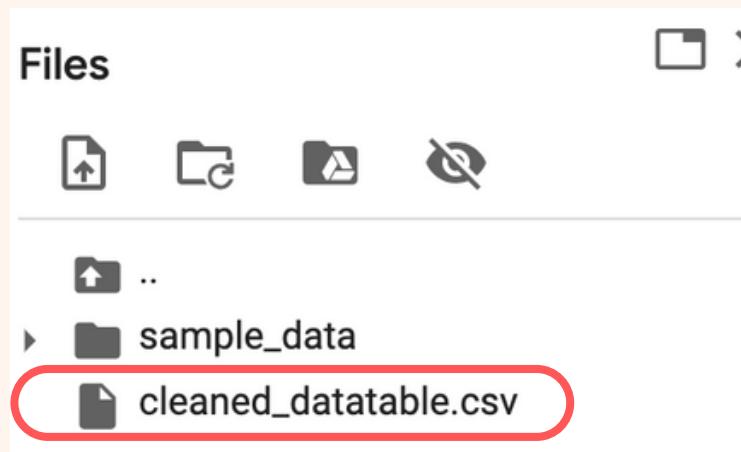
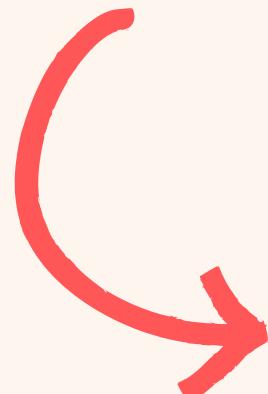
Machine Learning

Loading data from google drive (Data source)

```
✓ 13s   1 from pydrive.auth import GoogleAuth
       2 from pydrive.drive import GoogleDrive
       3 from google.colab import auth
       4 from oauth2client.client import GoogleCredentials
       5 # Authenticate and create the PyDrive client.
       6
       7 auth.authenticate_user()

✓ [2] 1 !pip install pydrive

✓ 1s    1 gauth = GoogleAuth()
       2 gauth.credentials = GoogleCredentials.get_application_default()
       3 drive = GoogleDrive(gauth)
       4
       5 link_file = 'https://drive.google.com/file/d/1Pej\_RqvngXB24WFV0j10njP-hm5-TaeR/view?usp=sharing'
       6 _,_,_,_,id,_ = link_file.split('/')
       7 downloaded = drive.CreateFile({'id':id})
       8 downloaded.GetContentFile('cleaned_datatable.csv')
```



Ingesting data to storage bucket (Data lake)



Google Cloud

```
0s [4] 1 from google.cloud import storage
2
3 project_id = 'projectmadt' ←
4 # project_id = 'your-project-id'
5
6 bucket_name = 'mootod_data_project' ←
7 # bucket_name = 'your-bucket-name'
8 bucket = storage.Client(project=project_id).bucket(bucket_name)

2s [ ] 1 bucket = storage.Client(project=project_id).bucket(bucket_name)
2 local_file_path = '/content/cleaned_datatable.csv' ←
3 blob = bucket.blob('cleaned_datatable.csv')
4 # local_file_path = '/path/to/local/file'
5 # blob = bucket.blob('path/to/remote/file')
6 blob.upload_from_filename(local_file_path)
```

Bucket details

mootod_data_project

Location	Storage class	Public access	Protection
asia-southeast1 (Singapore)	Standard	Subject to object ACLs	None

OBJECTS CONFIGURATION PERMISSIONS PROTECTION LIFECYCLE OBSERVABILITY NEW INVENTORY REPORTS NEW

Buckets > mootod_data_project

UPLOAD FILES UPLOAD FOLDER CREATE FOLDER TRANSFER DATA MANAGE HOLDS DOWNLOAD DELETE

Filter by name prefix only Filter objects and folders

<input type="checkbox"/> Name	Size	Type	Created	Storage class	Last modified
<input type="checkbox"/> cleaned_datatable.csv	519.2 KB	text/csv	Jun 20, 2023, 12:33:51 PM	Standard	Jun 20, 2023, 12:33:51 PM
<input type="checkbox"/> sale-predict-model.pkl	3 MB	application/octet-stream	Jun 14, 2023, 3:05:35 PM	Standard	Jun 14, 2023, 3:05:35 PM



Data Source Mootod tidfun



Cleaning (Python google colab)



Data lake google cloud storage



ETL data pipeline Mage data tool



Bigquery data warehouse



Query



Dashboard Looker



Machine Learning

ETL data pipeline

Compute Engine

VM instances CREATE INSTANCE IMPORT VM REFRESH

Virtual machines

- VM instances
- Instance templates
- Sole-tenant nodes
- Machine images
- TPUs
- Committed use discounts

INSTANCES OBSERVABILITY INSTANCE SCHEDULES

Instance "mootod-data-project" is underutilized. You can save an estimated \$63 per month by switching to the machine type: e2-custom. [Learn more](#) DISMISS

VM instances

Status	Name	Zone	Recommendations	In use by	Internal IP	External IP	Connect
<input checked="" type="checkbox"/>	mootod-data-project	asia-southeast1-b	Save \$63 / mo		10.148.0.2 (nic0)	34.126.138.28 (nic0)	SSH

ssh.cloud.google.com/v2/ssh/projects/projectmadt/zones/asia-southeast1-b/instances/mootod-data-project?authuser=2&hl=en_GB&proj...

ssh.cloud.google.com/v2/ssh/projects/projectmadt/zones/asia-southeast1-b/instances/mootod-data-project?authuser=2&hl=en_GB...

SSH-in-browser

UPLOAD FILE DOWNLOAD FILE

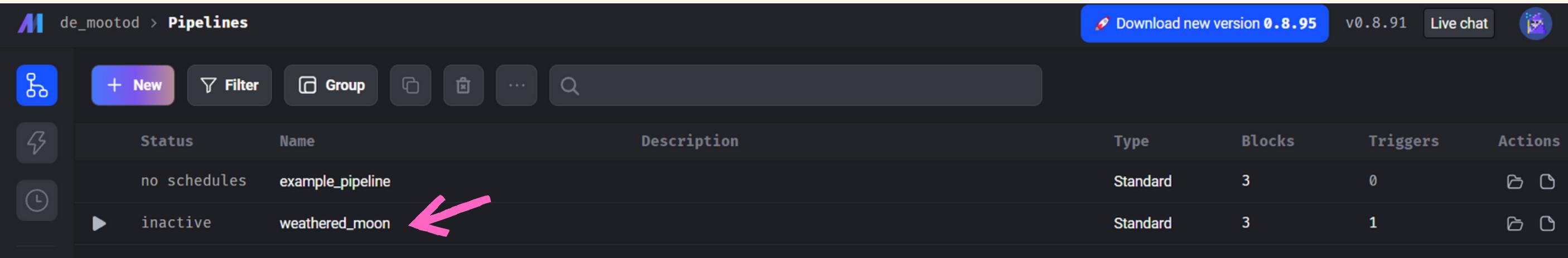
```
Linux mootod-data-project 5.10.0-22-cloud-amd64 #1 SMP Debian 5.10.178-3 (2023-04-22) x86_64
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Jun 19 15:21:03 2023 from 35.235.240.161
csaran111@mootod-data-project:~$ mage start de_mootod
```

INFO [remote/RuntimeMigration] will assume
Checking port 6789...

34.126.138.28:6789

ETL data pipeline



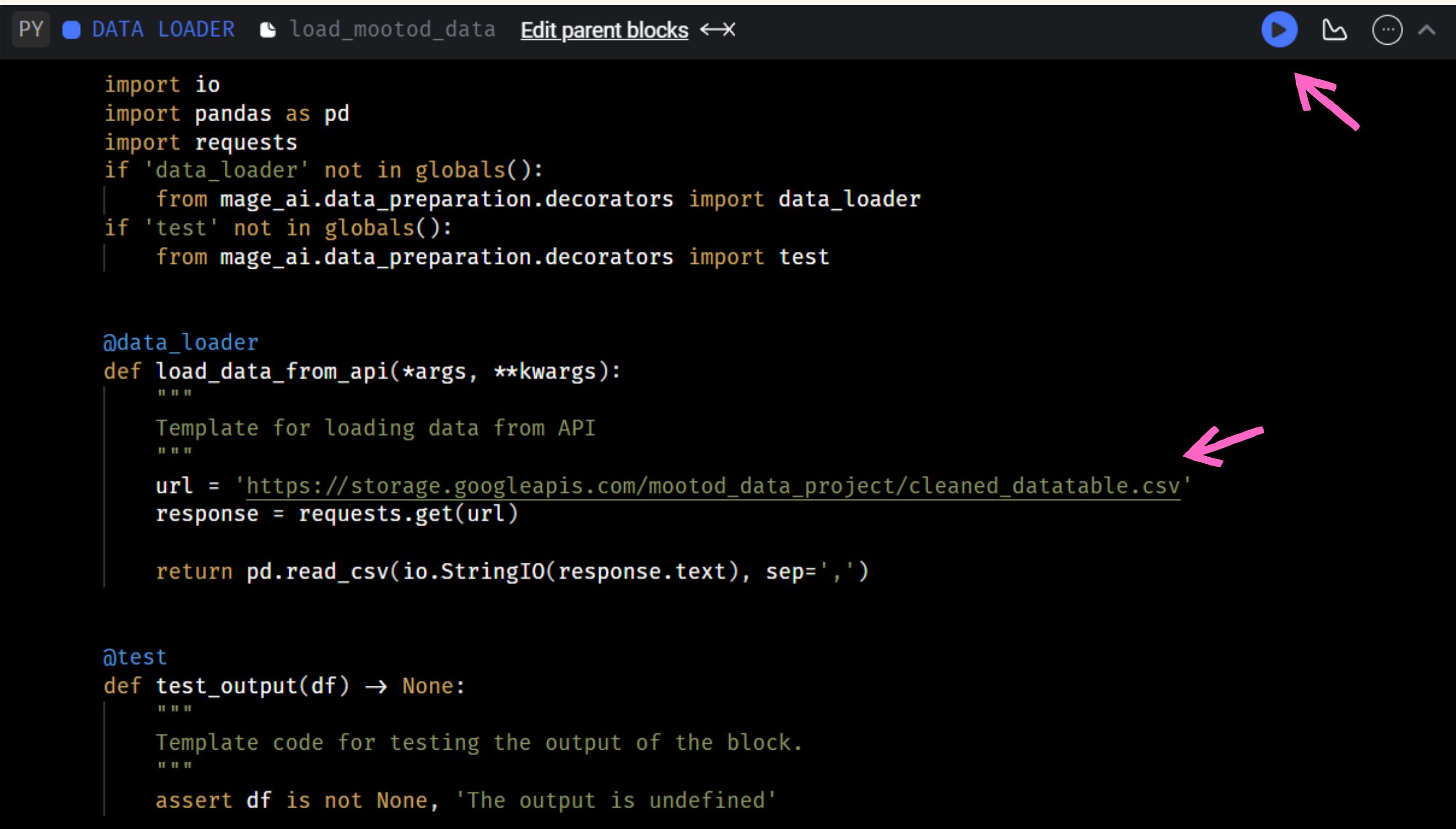
The screenshot shows the MAGE Pipelines interface. At the top, there's a navigation bar with the project name "de_mootod > Pipelines". On the right side of the header, there are buttons for "Download new version 0.8.95" (v0.8.91), "Live chat", and a user icon. Below the header is a toolbar with icons for "New", "Filter", "Group", and other actions. The main area is a table listing pipelines:

Status	Name	Description	Type	Blocks	Triggers	Actions
no schedules	example_pipeline		Standard	3	0	Edit Delete
inactive	weathered_moon		Standard	3	1	Edit Delete

A pink arrow points to the "weathered_moon" pipeline in the list.

ETL data pipeline

loader



```
PY DATA LOADER load_mootod_data Edit parent blocks ↗
```

```
import io
import pandas as pd
import requests
if 'data_loader' not in globals():
    from mage_ai.data_preparation.decorators import data_loader
if 'test' not in globals():
    from mage_ai.data_preparation.decorators import test

@data_loader
def load_data_from_api(*args, **kwargs):
    """
    Template for loading data from API
    """
    url = 'https://storage.googleapis.com/mootod_data_project/cleaned_datatable.csv'
    response = requests.get(url)

    return pd.read_csv(io.StringIO(response.text), sep=',')

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

ETL data pipeline

PY DATA LOADER load_mootod_data [Edit parent blocks](#) ↗

```

24     template code for testing the output of the block.
25     """
26     assert df is not None, 'The output is undefined'
27
28

```

1/1 tests passed.

	ref_no	time	vendor	bill_type	pickup_type	payment_method	sale
0	3001012	7/3/2022 11:39	2	2	7	3	400
1	3001013	7/3/2022 11:43	2	2	4	3	300
2	3001014	7/3/2022 11:52	2	2	2	2	65
3	3001015	7/3/2022 11:56	2	2	6	3	180
4	3001016	7/3/2022 11:56	2	2	6	3	180
5	3001017	7/3/2022 12:20	2	1	1	1	75
6	3001018	7/3/2022 13:27	2	2	7	3	325
7	3001019	7/3/2022 14:10	2	2	4	3	75
8	3001020	7/3/2022 15:12	2	2	7	3	80
9	3001021	7/3/2022 15:16	2	2	3	3	150

▲ 10260 rows x 10 columns ✓ Expand table ↗

ETL data pipeline

Transform

PY TRANSFORMER mootod_tranformation 1 parent block ←○

```

import pandas as pd
if 'transformer' not in globals():
    from mage_ai.data_preparation.decorators import transformer
if 'test' not in globals():
    from mage_ai.data_preparation.decorators import test


@transformer
def transform(df, *args, **kwargs):
    """
    Template code for a transformer block.

    Add more parameters to this function if this block has multiple parent blocks.
    There should be one parameter for each output variable from each parent block.

    Args:
        data: The output from the upstream parent block
        args: The output from any additional upstream blocks (if applicable)

    Returns:
        Anything (e.g. data frame, dictionary, array, int, str, etc.)
    """
    # Specify your transformation logic here
    df=df.drop_duplicates().reset_index(drop=True)
    df['order_id']=df.index
    #####

```

ETL data pipeline

Transform

PY TRANSFORMER mootod_tranformation 1 parent block ←○

```

#####
df['time'] = pd.to_datetime(df['time'], format='%d/%m/%Y %H:%M')
datetime_dim = df[['time']].reset_index(drop=True)
datetime_dim['sale_time']=datetime_dim['time'].dt.time
datetime_dim['sale_day']=datetime_dim['time'].dt.day
datetime_dim['sale_month']=datetime_dim['time'].dt.month
datetime_dim['sale_year']=datetime_dim['time'].dt.year

datetime_dim['datetime_id']=datetime_dim.index
datetime_dim = datetime_dim[['datetime_id','time','sale_time','sale_day','sale_month','sale_year']]
#####
vendor_name={
    1:"Noon",
    2:"Minnie"
}
vendor_dim=df[['vendor']].reset_index(drop=True)
vendor_dim['vendor_id']=vendor_dim.index
vendor_dim['vendor_name']=vendor_dim['vendor'].map(vendor_name)
vendor_dim=vendor_dim[['vendor_id','vendor','vendor_name']]
#####

bill_type_name={
    1:"For_here",
    2:"To_go"
}
bill_type_dim=df[['bill_type']].reset_index(drop=True)
bill_type_dim['bill_type_id']=bill_type_dim.index
bill_type_dim['bill_type_name']=bill_type_dim['bill_type'].map(bill_type_name)
bill_type_dim=bill_type_dim[['bill_type_id','bill_type','bill_type_name']]
#####

```

ETL data pipeline

Transform

```

PY ● TRANSFORMER mootod_transformation 1 parent block ←○
54     #####  

55     #####  

56     pickup_type_name={  

57         1:"served",  

58         2:"pickup_by_themselves",  

59         3:"LineMan",  

60         4:"GrabFood",  

61         5:"FoodPanda",  

62         6:"ShopeeFood",  

63         7:"Robinhood"  

64     }  

65     pickup_type_dim=df[['pickup_type']].reset_index(drop=True)  

66     pickup_type_dim['pickup_type_id']=pickup_type_dim.index  

67     pickup_type_dim['pickup_type_name']=pickup_type_dim['pickup_type'].map(pickup_type_name)  

68     pickup_type_dim=pickup_type_dim[['pickup_type_id','pickup_type','pickup_type_name']]  

69     #####  

70     payment_method_name={  

71         1:"transfer",  

72         2:"cash",  

73         ...3:"other"  

74     }  

75     payment_method_dim=df[['payment_method']].reset_index(drop=True)  

76     payment_method_dim['payment_method_id']=payment_method_dim.index  

77     payment_method_dim['payment_method_name']=payment_method_dim['payment_method'].map(payment_method_name)  

78     payment_method_dim=payment_method_dim[['payment_method_id','payment_method','payment_method_name']]  

79     #####  

80     fact_table = df.merge(datetime_dim, left_on='order_id', right_on='datetime_id') \  

81             .merge(vendor_dim, left_on='order_id', right_on='vendor_id') \  

82             .merge(bill_type_dim, left_on='order_id', right_on='bill_type_id') \  

83             .merge(pickup_type_dim, left_on='order_id', right_on='pickup_type_id') \  

84             .merge(payment_method_dim, left_on='order_id', right_on='payment_method_id') \  

85             [['order_id','ref_no', 'datetime_id', 'vendor_id',  

86              'bill_type_id', 'pickup_type_id', 'payment_method_id',  

87              'sale', 'payment', 'tip', 'change',]]  

88     #####

```

ETL data pipeline

Transform

```
89
90     return {"datetime_dim":datetime_dim.to_dict(orient='dict'),
91             "vendor_dim":vendor_dim.to_dict(orient='dict'),
92             "bill_type_dim":bill_type_dim.to_dict(orient='dict'),
93             "pickup_type_dim":pickup_type_dim.to_dict(orient='dict'),
94             "payment_method_dim":payment_method_dim.to_dict(orient='dict'),
95             "fact_table":fact_table.to_dict(orient='dict')}

96
97
98 @test
99 def test_output(output, *args) → None:
100     """
101     Template code for testing the output of the block.
102     """
103     assert output is not None, 'The output is undefined'
104
```

ETL data pipeline

Exporter

```

PY DATA EXPORTER mootod_bq_load 1 parent block ←○ ▶ ⌂ ^

Positional arguments for decorated function:

@data_exporter
def export_data(data):
    data → mootod_transformation

        from mage_ai.data_preparation.repo_manager import get_repo_path
        from mage_ai.io.bigquery import BigQuery
        from mage_ai.io.config import ConfigFileLoader
        from pandas import DataFrame
        from os import path

        if 'data_exporter' not in globals():
            from mage_ai.data_preparation.decorators import data_exporter

@data_exporter
def export_data_to_big_query(data, **kwargs) → None:
    """
        Template for exporting data to a BigQuery warehouse.
        Specify your configuration settings in 'io_config.yaml'.

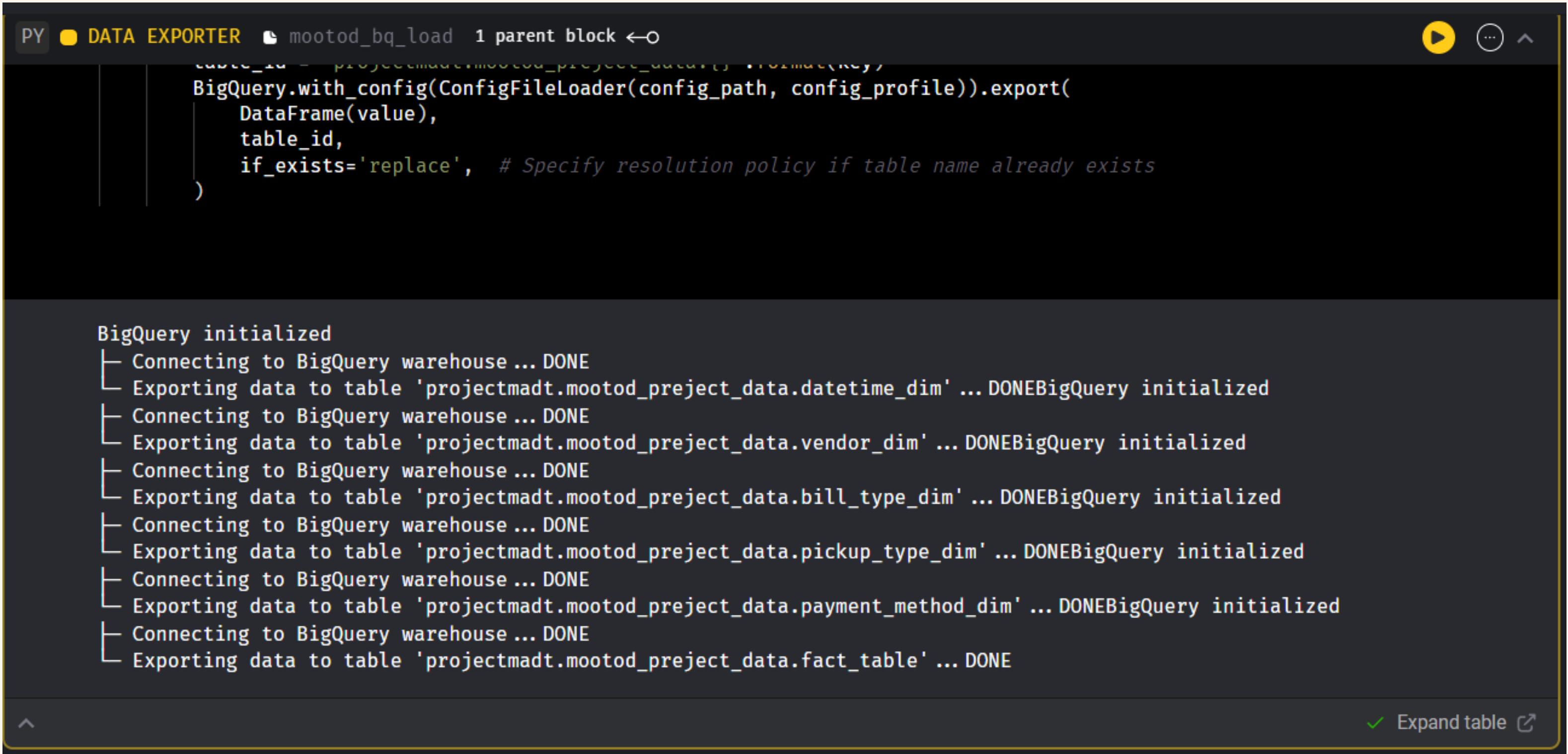
        Docs: https://docs.mage.ai/design/data-loading#bigquery
    """
    config_path = path.join(get_repo_path(), 'io_config.yaml')
    config_profile = 'default'

    for key,value in data.items():
        table_id = 'projectmadt.mootod_project_data.{}'.format(key)
        BigQuery.with_config(ConfigFileLoader(config_path, config_profile)).export(
            DataFrame(value),
            table_id,
            if_exists='replace', # Specify resolution policy if table name already exists
        )

```

ETL data pipeline

Exporter



The screenshot shows a Jupyter Notebook cell with the following content:

```
PY DATA EXPORTER mootod_bq_load 1 parent block ←
    table_id = projectmadt.mootod_project_data.datetime_dim
    BigQuery.with_config(ConfigFileLoader(config_path, config_profile)).export(
        DataFrame(value),
        table_id,
        if_exists='replace', # Specify resolution policy if table name already exists
    )
```

Below the code, the execution output is displayed:

```
BigQuery initialized
└─ Connecting to BigQuery warehouse ... DONE
└─ Exporting data to table 'projectmadt.mootod_project_data.datetime_dim' ... DONE
BigQuery initialized
└─ Connecting to BigQuery warehouse ... DONE
└─ Exporting data to table 'projectmadt.mootod_project_data.vendor_dim' ... DONE
BigQuery initialized
└─ Connecting to BigQuery warehouse ... DONE
└─ Exporting data to table 'projectmadt.mootod_project_data.bill_type_dim' ... DONE
BigQuery initialized
└─ Connecting to BigQuery warehouse ... DONE
└─ Exporting data to table 'projectmadt.mootod_project_data.pickup_type_dim' ... DONE
BigQuery initialized
└─ Connecting to BigQuery warehouse ... DONE
└─ Exporting data to table 'projectmadt.mootod_project_data.payment_method_dim' ... DONE
BigQuery initialized
└─ Connecting to BigQuery warehouse ... DONE
└─ Exporting data to table 'projectmadt.mootod_project_data.fact_table' ... DONE
```

At the bottom right of the output area, there is a button labeled "Expand table" with a checkmark icon.



Data Source Mootod tidfun



Cleaning (Python google colab)



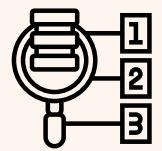
Data lake google cloud storage



ETL data pipeline Mage data tool



Bigquery data warehouse



Query



Dashboard Looker



Machine Learning

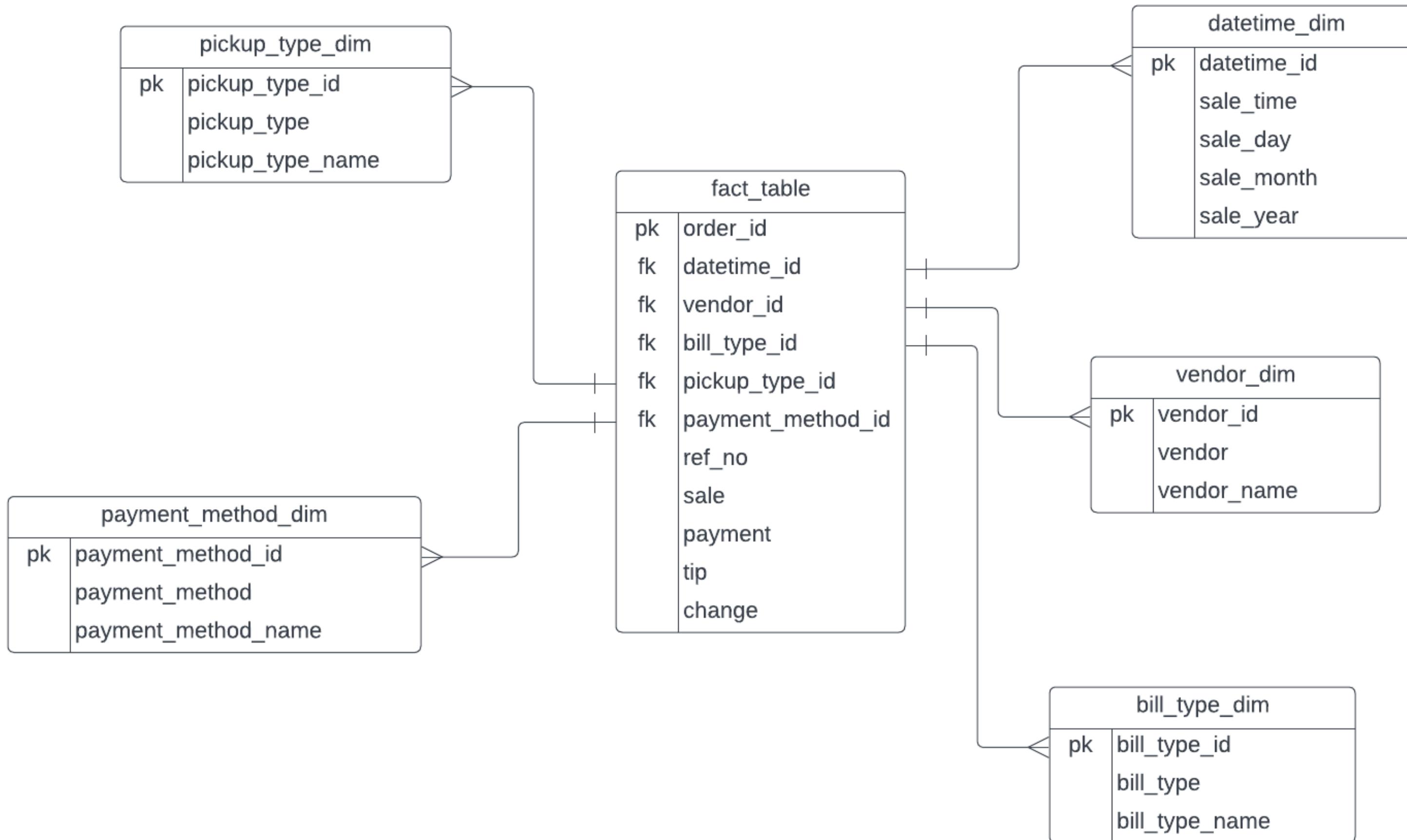
Data Warehouse

The screenshot shows the Google Cloud BigQuery Explorer interface. The top navigation bar includes the Google Cloud logo and the project name "ProjectMADT". The left sidebar has a search bar and various icons for external connections, saved queries, and datasets. The main area displays a list of resources under the "projectmadt" dataset. The list includes:

- External connections
- Saved queries (2)
- mootod_project_data
 - Pickup_name_rider
 - Pickup_rider
 - analytics_table
 - bill_type_dim
 - datetime_dim
 - fact_table
 - group_count_delivery
 - group_time
 - group_timeM5
 - hoursale_table
 - payment_method_dim
 - pickup_type_dim
 - vendor_dim

Four specific tables—bill_type_dim, datetime_dim, fact_table, and payment_method_dim—are highlighted with red rounded rectangles.

ER-DIAGRAM





Data Source Mootod tidfun



Cleaning (Python google colab)



Data lake google cloud storage



ETL data pipeline Mage data tool



Bigquery data warehouse



Query



Dashboard Looker



Machine Learning



Dashboard

[HTTPS://LOOKERSTUDIO.GOOGLE.COM/U/0/REPORTING/4A1DCF6E-52BB-454A-BA3A-B0978E1A42AF/PAGE/P_45X32PX26C](https://lookerstudio.google.com/u/0/reporting/4A1DCF6E-52BB-454A-BA3A-B0978E1A42AF/page/p_45X32PX26C)



Data Source Mootod tidfun



Cleaning (Python google colab)



Data lake google cloud storage



ETL data pipeline Mage data tool



Bigquery data warehouse



Query



Dashboard Looker



Machine Learning

Machine Learning

[HTTPS://COLAB.RESEARCH.GOOGLE.COM/DRIVE/12IKI3I1K3F6FUNPAS8TVFRCN9HEHWUJ](https://colab.research.google.com/drive/12IKI3I1K3F6FUNPAS8TVFRCN9HEHWUJ)

Thank You