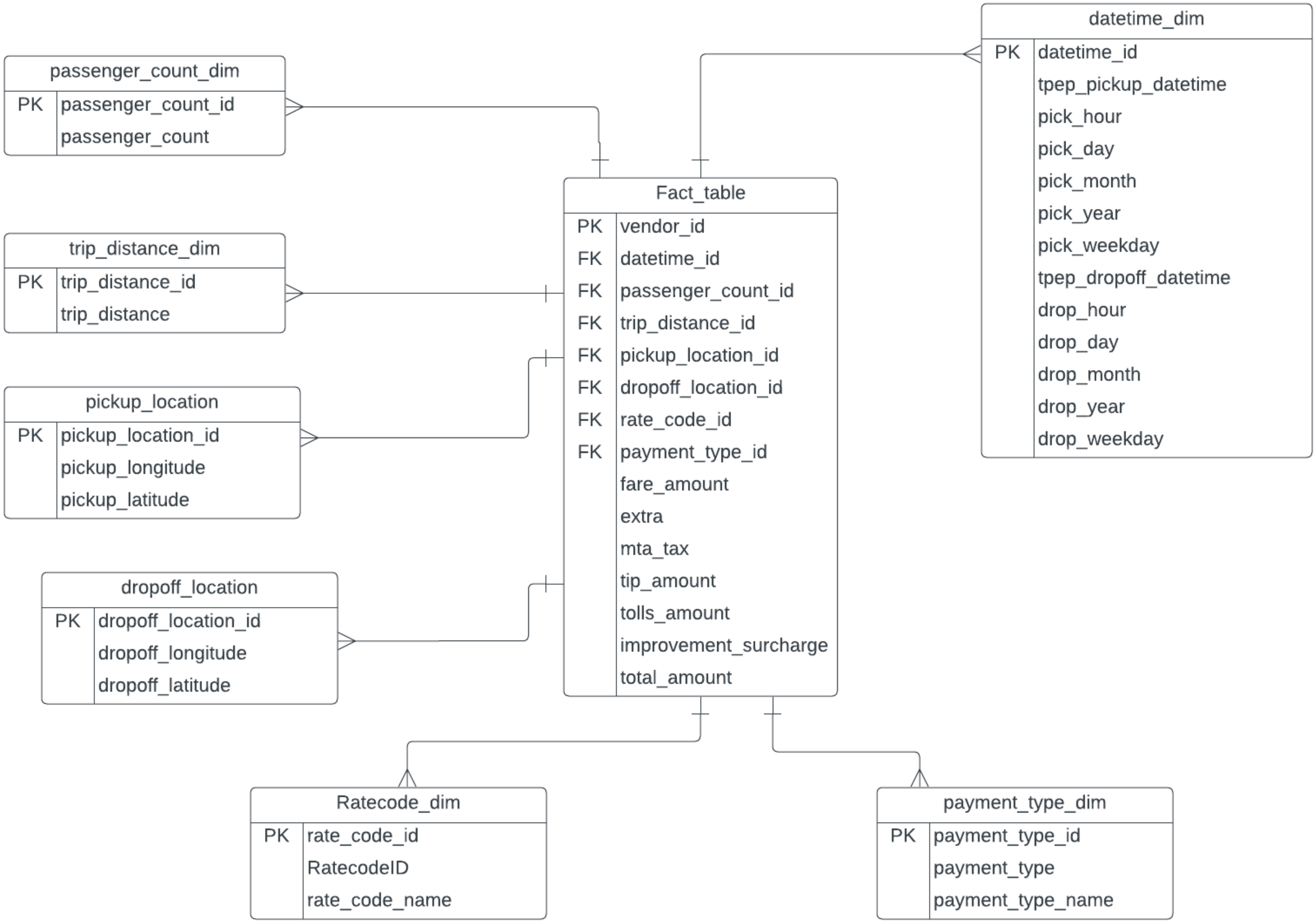


passenger_count_dim	
PK	passenger_count_id
	passenger_count

trip_distance_dim	
PK	trip_distance_id
	trip_distance

pickup_location	
PK	pickup_location_id
	pickup_longitude
	pickup_latitude

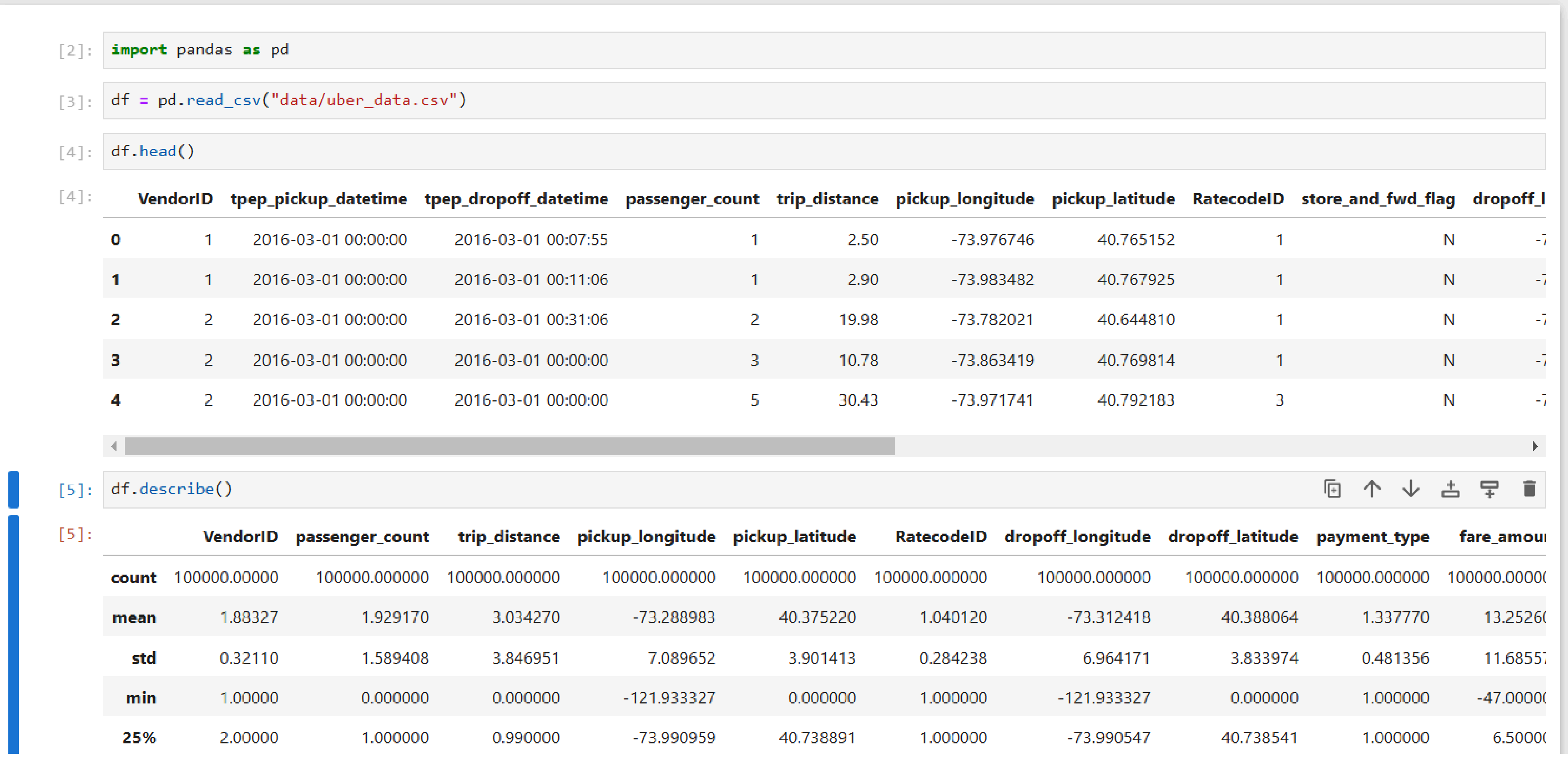
dropoff_location	
PK	dropoff_location_id
	dropoff_longitude
	dropoff_latitude

Fact_table	
PK	vendor_id
FK	datetime_id
FK	passenger_count_id
FK	trip_distance_id
FK	pickup_location_id
FK	dropoff_location_id
FK	rate_code_id
FK	payment_type_id
	fare_amount
	extra
	mta_tax
	tip_amount
	tolls_amount
	improvement_surcharge
	total_amount

datetime_dim	
PK	datetime_id
	tpep_pickup_datetime
	pick_hour
	pick_day
	pick_month
	pick_year
	pick_weekday
	tpep_dropoff_datetime
	drop_hour
	drop_day
	drop_month
	drop_year
	drop_weekday

Ratecode_dim	
PK	rate_code_id
	RatecodeID
	rate_code_name

payment_type_dim	
PK	payment_type_id
	payment_type
	payment_type_name



[2]:
```python
import pandas as pd
```

[3]:
```python
df = pd.read_csv("data/uber_data.csv")
```

[4]:
```python
df.head()
```

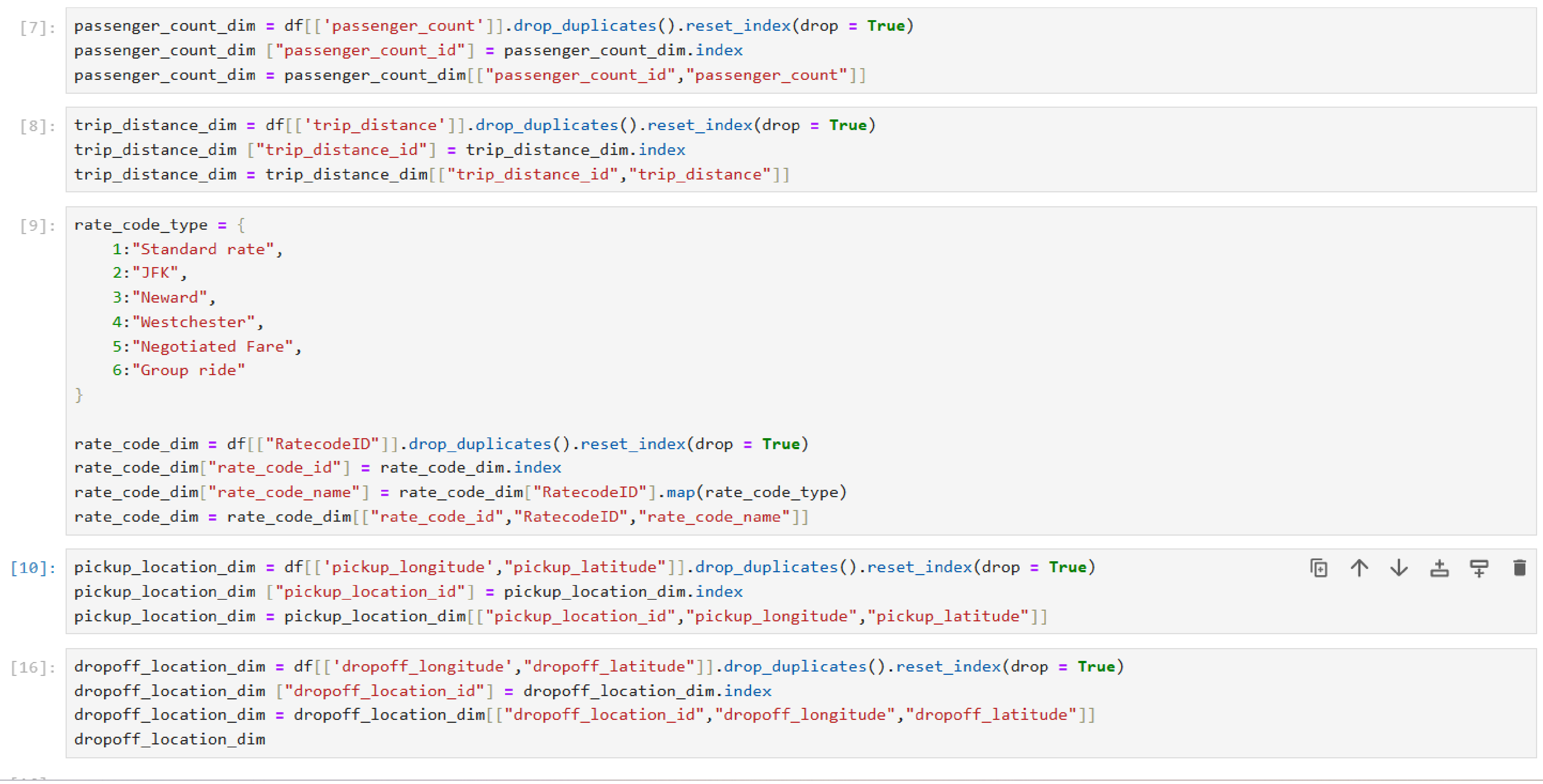
[4]:

	VendorID	tpep_pickup_datetime	tpep_dropoff_datetime	passenger_count	trip_distance	pickup_longitude	pickup_latitude	RatecodeID	store_and_fwd_flag	dropoff_l
0	1	2016-03-01 00:00:00	2016-03-01 00:07:55	1	2.50	-73.976746	40.765152	1	N	-7
1	1	2016-03-01 00:00:00	2016-03-01 00:11:06	1	2.90	-73.983482	40.767925	1	N	-7
2	2	2016-03-01 00:00:00	2016-03-01 00:31:06	2	19.98	-73.782021	40.644810	1	N	-7
3	2	2016-03-01 00:00:00	2016-03-01 00:00:00	3	10.78	-73.863419	40.769814	1	N	-7
4	2	2016-03-01 00:00:00	2016-03-01 00:00:00	5	30.43	-73.971741	40.792183	3	N	-7

[5]:
```python
df.describe()
```

[5]:

	VendorID	passenger_count	trip_distance	pickup_longitude	pickup_latitude	RatecodeID	dropoff_longitude	dropoff_latitude	payment_type	fare_amou
count	100000.00000	100000.000000	100000.000000	100000.000000	100000.000000	100000.000000	100000.000000	100000.000000	100000.000000	100000.00000
mean	1.88327	1.929170	3.034270	-73.288983	40.375220	1.040120	-73.312418	40.388064	1.337770	13.2526
std	0.32110	1.589408	3.846951	7.089652	3.901413	0.284238	6.964171	3.833974	0.481356	11.6855
min	1.00000	0.000000	0.000000	-121.933327	0.000000	1.000000	-121.933327	0.000000	1.000000	-47.0000
25%	2.00000	1.000000	0.990000	-73.990959	40.738891	1.000000	-73.990547	40.738541	1.000000	6.5000



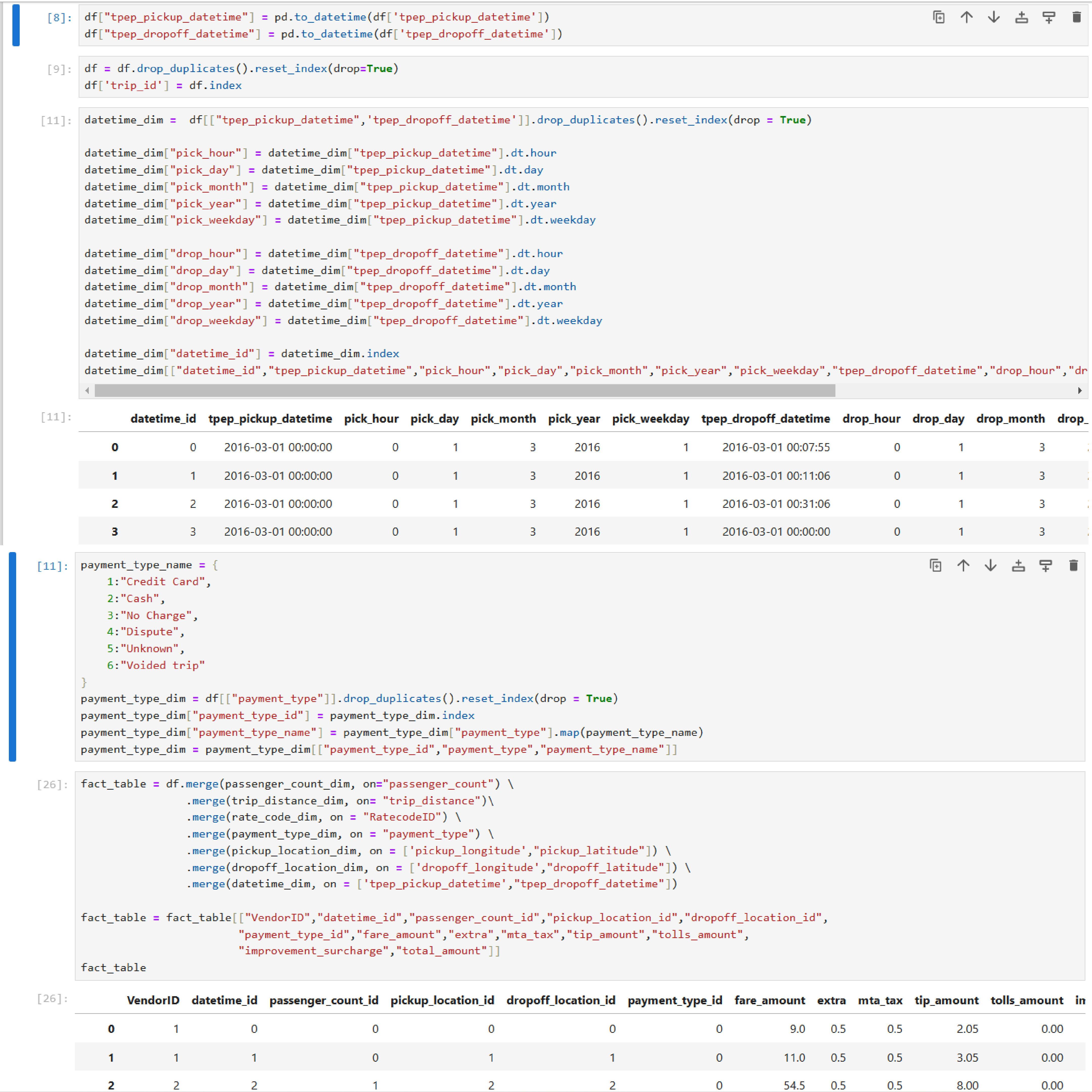
[7]:
```python
passenger_count_dim = df[['passenger_count']].drop_duplicates().reset_index(drop = True)
passenger_count_dim ["passenger_count_id"] = passenger_count_dim.index
passenger_count_dim = passenger_count_dim[["passenger_count_id","passenger_count"]]
```

[8]:
```python
trip_distance_dim = df[['trip_distance']].drop_duplicates().reset_index(drop = True)
trip_distance_dim ["trip_distance_id"] = trip_distance_dim.index
trip_distance_dim = trip_distance_dim[["trip_distance_id","trip_distance"]]
```

[9]:
```python
rate_code_type = {
    1:"Standard rate",
    2:"JFK",
    3:"Neward",
    4:"Westchester",
    5:"Negotiated Fare",
    6:"Group ride"
}

rate_code_dim = df[["RatecodeID"]].drop_duplicates().reset_index(drop = True)
rate_code_dim["rate_code_id"] = rate_code_dim.index
rate_code_dim["rate_code_name"] = rate_code_dim["RatecodeID"].map(rate_code_type)
rate_code_dim = rate_code_dim[["rate_code_id","RatecodeID","rate_code_name"]]
```

[10]:
```python
pickup_location_dim = df[['pickup_longitude',"pickup_latitude"]].drop_duplicates().reset_index(drop = True)
pickup_location_dim ["pickup_location_id"] = pickup_location_dim.index
pickup_location_dim = pickup_location_dim[["pickup_location_id","pickup_longitude","pickup_latitude"]]
```

[16]:
```python
dropoff_location_dim = df[['dropoff_longitude',"dropoff_latitude"]].drop_duplicates().reset_index(drop = True)
dropoff_location_dim ["dropoff_location_id"] = dropoff_location_dim.index
dropoff_location_dim = dropoff_location_dim[["dropoff_location_id","dropoff_longitude","dropoff_latitude"]]
dropoff_location_dim
```



[8]:

```
df["tpep_pickup_datetime"] = pd.to_datetime(df['tpep_pickup_datetime'])
df["tpep_dropoff_datetime"] = pd.to_datetime(df['tpep_dropoff_datetime'])
```

[9]:

```
df = df.drop_duplicates().reset_index(drop=True)
df['trip_id'] = df.index
```

[11]:

```
datetime_dim =  df[["tpep_pickup_datetime",'tpep_dropoff_datetime']].drop_duplicates().reset_index(drop = True)

datetime_dim["pick_hour"] = datetime_dim["tpep_pickup_datetime"].dt.hour
datetime_dim["pick_day"] = datetime_dim["tpep_pickup_datetime"].dt.day
datetime_dim["pick_month"] = datetime_dim["tpep_pickup_datetime"].dt.month
datetime_dim["pick_year"] = datetime_dim["tpep_pickup_datetime"].dt.year
datetime_dim["pick_weekday"] = datetime_dim["tpep_pickup_datetime"].dt.weekday

datetime_dim["drop_hour"] = datetime_dim["tpep_dropoff_datetime"].dt.hour
datetime_dim["drop_day"] = datetime_dim["tpep_dropoff_datetime"].dt.day
datetime_dim["drop_month"] = datetime_dim["tpep_dropoff_datetime"].dt.month
datetime_dim["drop_year"] = datetime_dim["tpep_dropoff_datetime"].dt.year
datetime_dim["drop_weekday"] = datetime_dim["tpep_dropoff_datetime"].dt.weekday

datetime_dim["datetime_id"] = datetime_dim.index
datetime_dim[["datetime_id","tpep_pickup_datetime","pick_hour","pick_day","pick_month","pick_year","pick_weekday","tpep_dropoff_datetime","drop_hour","dr
```

[11]:

	datetime_id	tpep_pickup_datetime	pick_hour	pick_day	pick_month	pick_year	pick_weekday	tpep_dropoff_datetime	drop_hour	drop_day	drop_month	drop_
0	0	2016-03-01 00:00:00	0	1	3	2016	1	2016-03-01 00:07:55	0	1	3	
1	1	2016-03-01 00:00:00	0	1	3	2016	1	2016-03-01 00:11:06	0	1	3	
2	2	2016-03-01 00:00:00	0	1	3	2016	1	2016-03-01 00:31:06	0	1	3	
3	3	2016-03-01 00:00:00	0	1	3	2016	1	2016-03-01 00:00:00	0	1	3	

[11]:

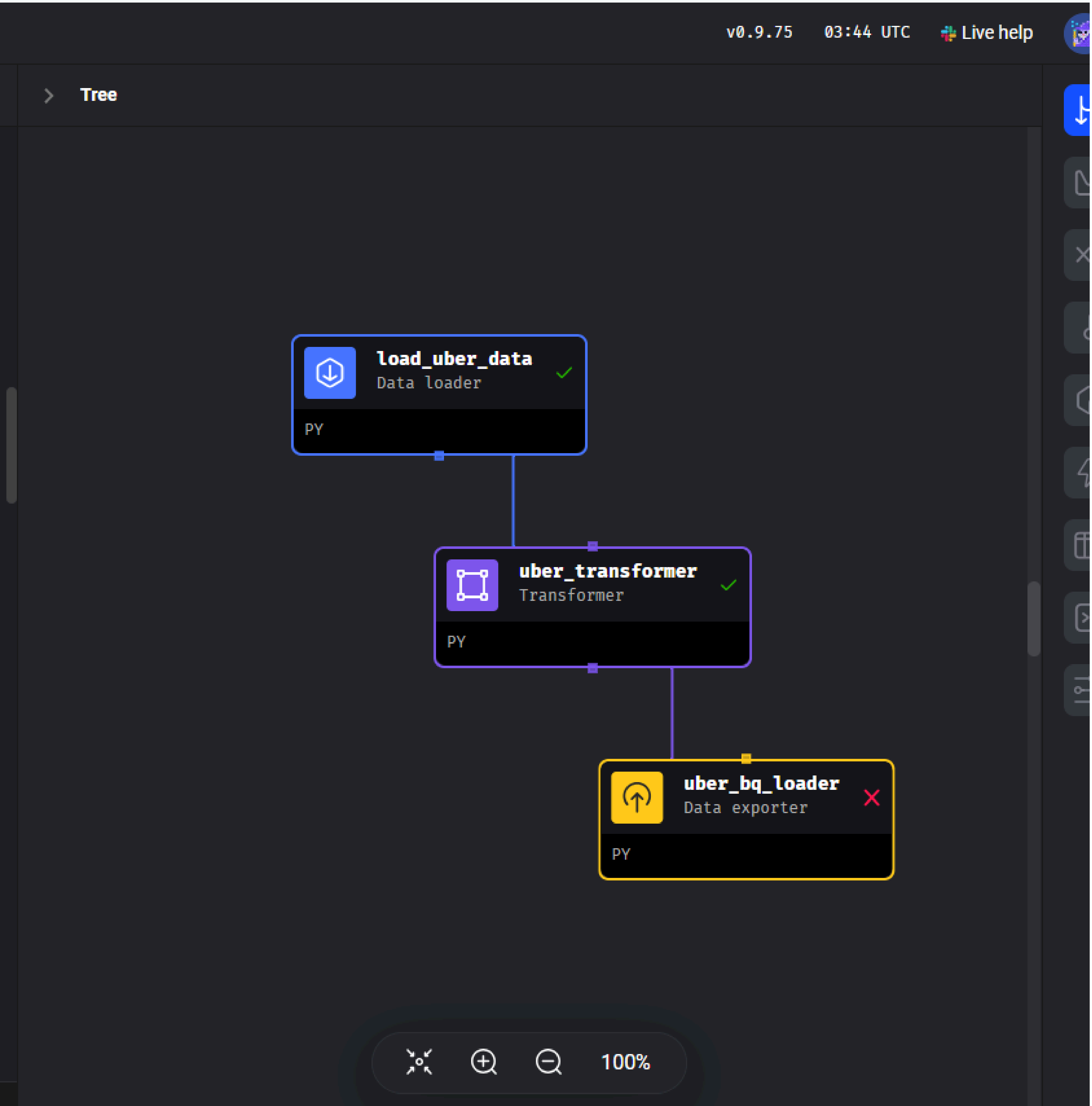
```
payment_type_name = {
    1:"Credit Card",
    2:"Cash",
    3:"No Charge",
    4:"Dispute",
    5:"Unknown",
    6:"Voided trip"
}
payment_type_dim = df[["payment_type"]].drop_duplicates().reset_index(drop = True)
payment_type_dim["payment_type_id"] = payment_type_dim.index
payment_type_dim["payment_type_name"] = payment_type_dim["payment_type"].map(payment_type_name)
payment_type_dim = payment_type_dim[["payment_type_id","payment_type","payment_type_name"]]
```

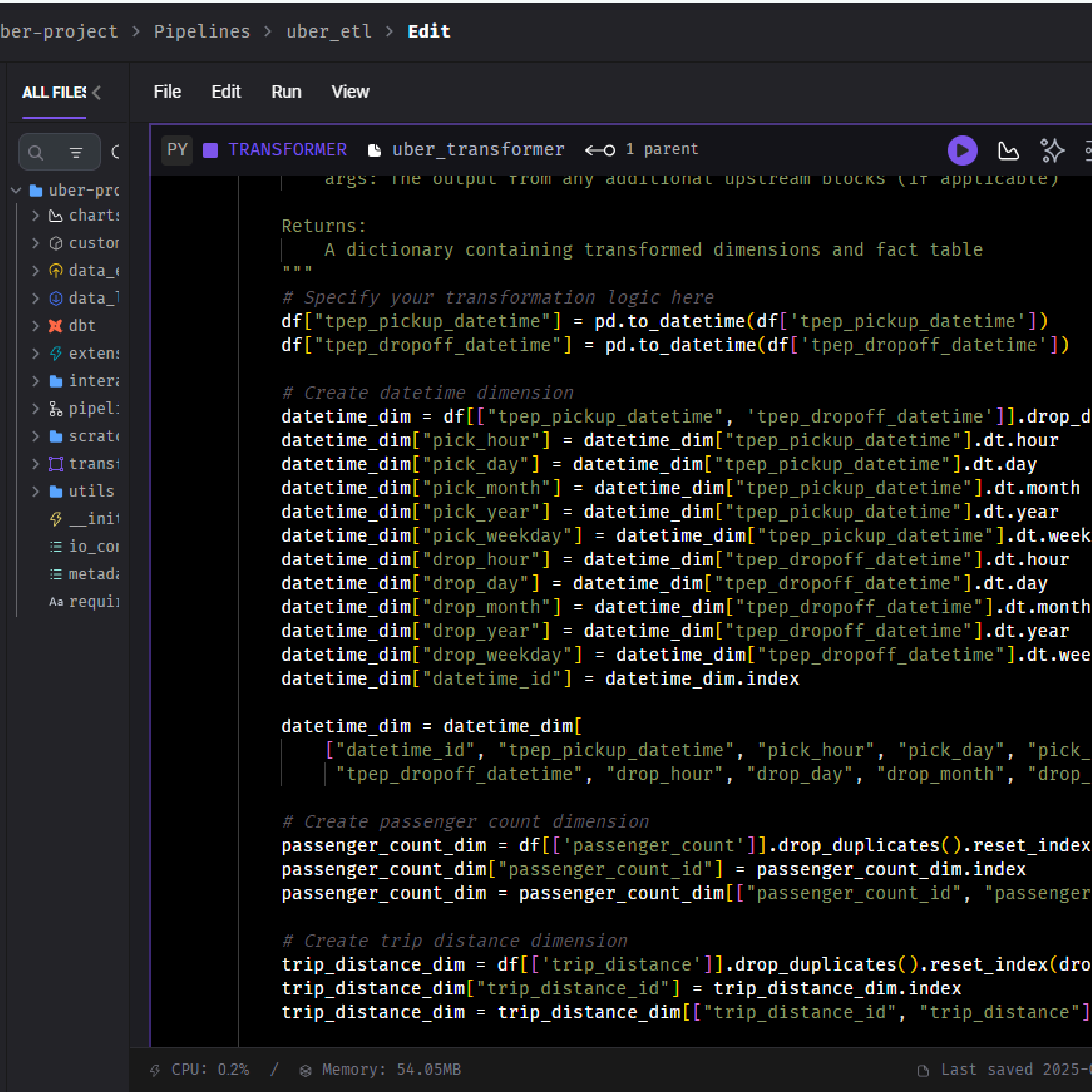
[26]:

```
fact_table = df.merge(passenger_count_dim, on="passenger_count") \
             .merge(trip_distance_dim, on= "trip_distance")\
             .merge(rate_code_dim, on = "RatecodeID") \
             .merge(payment_type_dim, on = "payment_type") \
             .merge(pickup_location_dim, on = ['pickup_longitude',"pickup_latitude"]) \
             .merge(dropoff_location_dim, on = ['dropoff_longitude',"dropoff_latitude"]) \
             .merge(datetime_dim, on = ['tpep_pickup_datetime',"tpep_dropoff_datetime"])

fact_table = fact_table[["VendorID","datetime_id","passenger_count_id","pickup_location_id","dropoff_location_id",
                    "payment_type_id","fare_amount","extra","mta_tax","tip_amount","tolls_amount",
                    "improvement_surcharge","total_amount"]]
fact_table
```

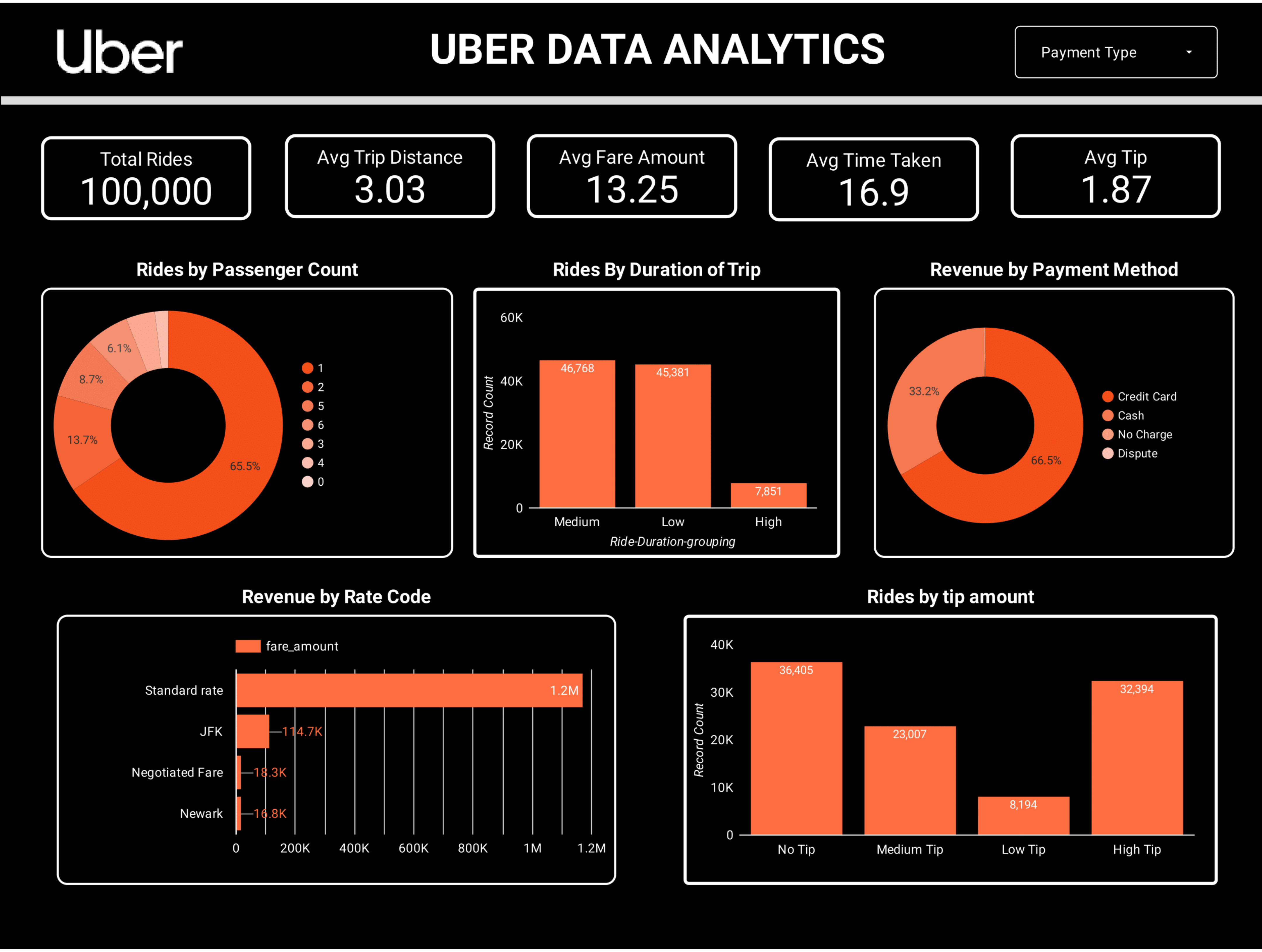
[26]:

	VendorID	datetime_id	passenger_count_id	pickup_location_id	dropoff_location_id	payment_type_id	fare_amount	extra	mta_tax	tip_amount	tolls_amount	in
0	1	0	0	0	0	0	9.0	0.5	0.5	2.05	0.00	
1	1	1	0	1	1	0	11.0	0.5	0.5	3.05	0.00	
2	2	2	1	2	2	0	54.5	0.5	0.5	8.00	0.00	


v0.9.75
03:44 UTC
Live help
Tree
load_uber_data
Data loader
✓
PY
uber_transformer
Transformer
✓
PY
uber_bq_loader
Data exporter
✗
PY
100%



```
        args: The output from any additional upstream blocks (if applicable)

    Returns:
        A dictionary containing transformed dimensions and fact table
    """
    # Specify your transformation logic here
    df["tpep_pickup_datetime"] = pd.to_datetime(df['tpep_pickup_datetime'])
    df["tpep_dropoff_datetime"] = pd.to_datetime(df['tpep_dropoff_datetime'])

    # Create datetime dimension
    datetime_dim = df[["tpep_pickup_datetime", 'tpep_dropoff_datetime']].drop_d
    datetime_dim["pick_hour"] = datetime_dim["tpep_pickup_datetime"].dt.hour
    datetime_dim["pick_day"] = datetime_dim["tpep_pickup_datetime"].dt.day
    datetime_dim["pick_month"] = datetime_dim["tpep_pickup_datetime"].dt.month
    datetime_dim["pick_year"] = datetime_dim["tpep_pickup_datetime"].dt.year
    datetime_dim["pick_weekday"] = datetime_dim["tpep_pickup_datetime"].dt.week
    datetime_dim["drop_hour"] = datetime_dim["tpep_dropoff_datetime"].dt.hour
    datetime_dim["drop_day"] = datetime_dim["tpep_dropoff_datetime"].dt.day
    datetime_dim["drop_month"] = datetime_dim["tpep_dropoff_datetime"].dt.month
    datetime_dim["drop_year"] = datetime_dim["tpep_dropoff_datetime"].dt.year
    datetime_dim["drop_weekday"] = datetime_dim["tpep_dropoff_datetime"].dt.wee
    datetime_dim["datetime_id"] = datetime_dim.index

    datetime_dim = datetime_dim[
        ["datetime_id", "tpep_pickup_datetime", "pick_hour", "pick_day", "pick_
         "tpep_dropoff_datetime", "drop_hour", "drop_day", "drop_month", "drop_

    # Create passenger count dimension
    passenger_count_dim = df[['passenger_count']].drop_duplicates().reset_index
    passenger_count_dim["passenger_count_id"] = passenger_count_dim.index
    passenger_count_dim = passenger_count_dim[["passenger_count_id", "passenger

    # Create trip distance dimension
    trip_distance_dim = df[['trip_distance']].drop_duplicates().reset_index(dro
    trip_distance_dim["trip_distance_id"] = trip_distance_dim.index
    trip_distance_dim = trip_distance_dim[["trip_distance_id", "trip_distance"]
```



Uber

UBER DATA ANALYTICS

Payment Type ▾

Total Rides: 100,000

Avg Trip Distance: 3.03

Avg Fare Amount: 13.25

Avg Time Taken: 16.9

Avg Tip: 1.87

Rides by Passenger Count

Rides By Duration of Trip

Revenue by Payment Method

Revenue by Rate Code

Rides by tip amount