DEDS: Exercise 4

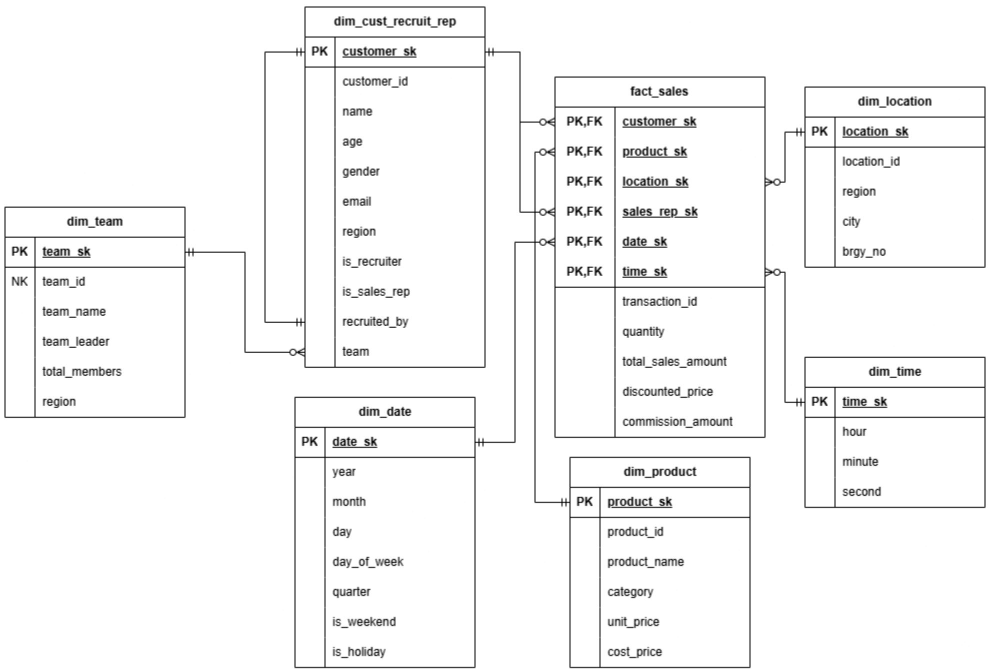
Vincent Itucal

Updated ERD

From the original ERD of LT4, I have instead combined customer, recruiter and sales rep dimension table into one (`dim_duct_recruit_rep`). I just added columns to identify if they are recruiters or sales rep. I have also related the `dim_team` with the `dim_duct_recruit_rep` introducing some snowflake schema elements (dim tables referencing dim tables). I have also created `dim_date` and `dim_time` tables to decouple date and time.

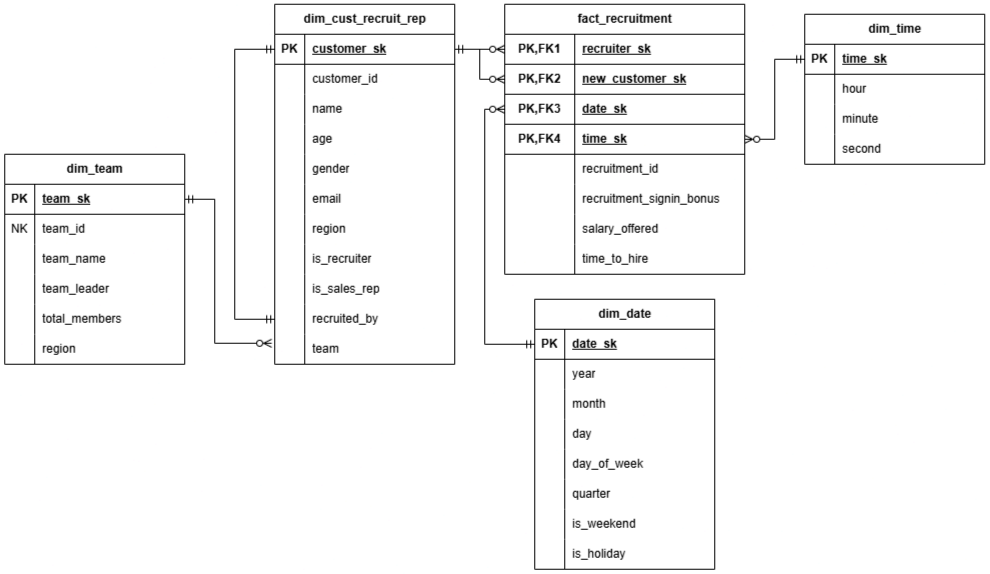
Business Process 1: Sales Performance Tracking

- Grain: A singe sales transaction made by a customer



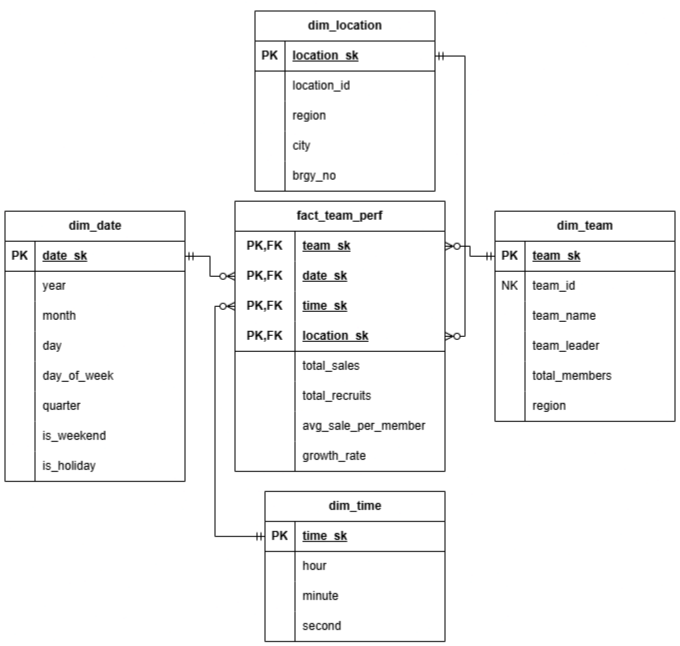
Business Process 2: Recruitment and Network

- Grain: A single recruitment event

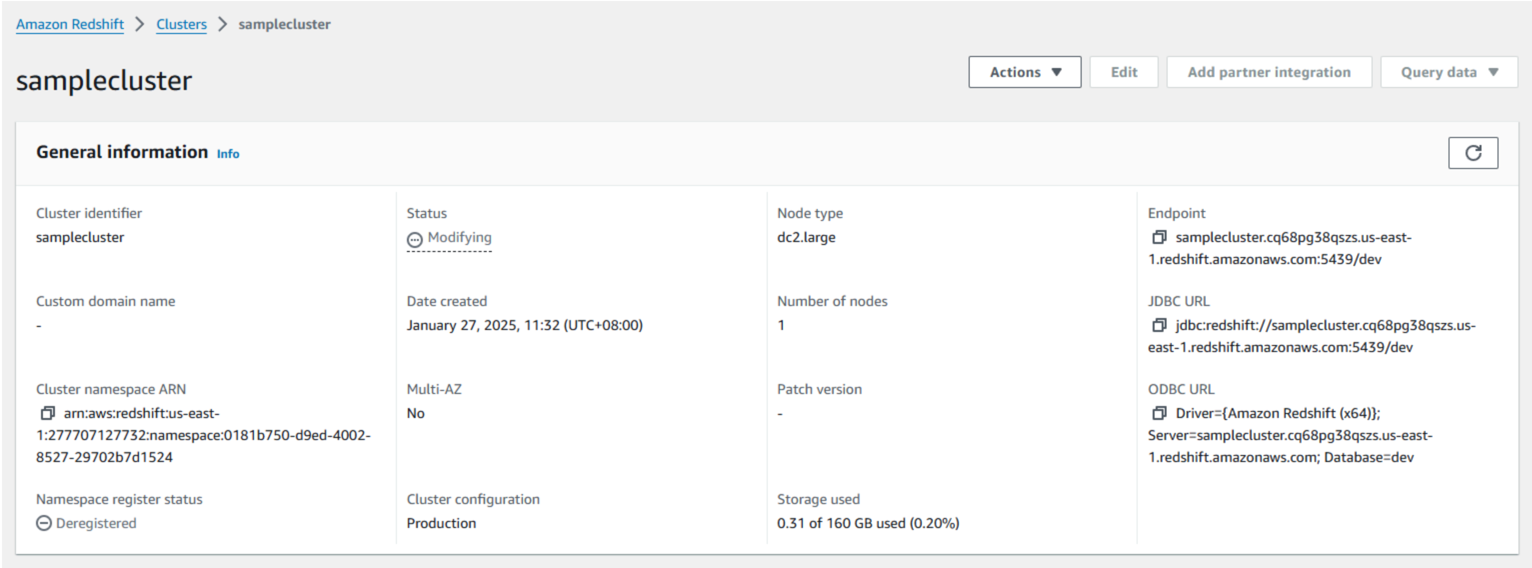


Business Process 3: Team Performance

- Grain: Sales team's performance within a time period



Redshift Console Details



In [1]:
```
from faker import Faker
import random
import uuid
import psycopg2
import redshift_connector
import pandas as pd
from datetime import datetime, timedelta
from sqlalchemy import create_engine
from sqlalchemy_redshift import dialect
from decouple import Config, RepositoryEnv
```

In [2]:
```
import warnings
warnings.filterwarnings('ignore')
```

In [3]:
```
%load_ext sql
```

In [4]:
```
config = Config(RepositoryEnv("../db_pass.txt"))
db_password = config("olap_pass")
olap_connection = config("olap_connection")
```

In [5]:
```
connection_string = f"postgresql+psycopg2://vincent:{db_password}@{olap_connection}:5439/mlm"
```

In [6]:
```
# engine = create_engine(connection_string)
```

In [7]:
```
engine = redshift_connector.connect(
    host=olap_connection,
    port=5439,
    database='mlm',
    user='vincent',
    password=db_password
)
```

In [8]:
```
get_ipython().run_line_magic('sql', connection_string)
```

Connecting to 'postgresql+psycopg2://vincent:***@samplecluster.cq68pg38qszs.us-east-1.redshift.amazonaws.com:5439/mlm'

Create Table Scripts

In [9]:
```
%%sql
CREATE TABLE IF NOT EXISTS dim_team (
        team_sk INT,
        team_id TEXT,
```

```sql
        team_name TEXT,
        team_leader TEXT,
        total_members INT,
        region TEXT
);

CREATE TABLE IF NOT EXISTS dim_date (
        date_sk INT,
        year INT,
        month TEXT,
        day INT,
        day_of_week TEXT,
        quarter INT,
        is_weekend BOOLEAN,
        is_holiday BOOLEAN
);

CREATE TABLE IF NOT EXISTS dim_time (
        time_sk INT,
        hour INT,
        minute INT,
        second INT
);

CREATE TABLE IF NOT EXISTS dim_cust_recruit_rep (
        customer_sk INT,
        customer_id TEXT,
        name TEXT,
        age INT,
        gender TEXT,
        email TEXT,
        region TEXT,
        is_recruiter BOOLEAN,
        is_sales_rep BOOLEAN,
        recruited_by INT,
        team INT
);

CREATE TABLE IF NOT EXISTS dim_product (
        product_sk INT,
        product_id TEXT,
        product_name TEXT,
        category TEXT,
        unit_price NUMERIC(10, 4),
        cost_price NUMERIC(10, 4)
);

CREATE TABLE IF NOT EXISTS dim_location (
        location_sk INT,
        location_id TEXT,
        region TEXT,
        city TEXT,
        brgy_no INT
);

CREATE TABLE IF NOT EXISTS fact_sales (
        customer_sk INT,
        product_sk INT,
        location_sk INT,
        sales_rep_sk INT,
        date_sk INT,
        time_sk INT,
        transaction_id TEXT,
        quantity INT,
        total_sales_amount NUMERIC(10, 4),
        discounted_price NUMERIC(10, 4),
        commission_amount NUMERIC(10, 4)
);

CREATE TABLE IF NOT EXISTS fact_recruitment (
        recruiter_sk INT,
        new_customer_sk INT,
        date_sk INT,
        time_sk INT,
        recruitment_id TEXT,
        recruitment_signin_bonus NUMERIC(10, 4),
        salary_offered NUMERIC(10, 4),
        time_to_hire INT
);

CREATE TABLE IF NOT EXISTS fact_team_perf (
        team_sk INT,
        date_sk INT,
        time_sk INT,
```

```
        location_sk INT,
        total_sales NUMERIC(10, 4),
        total_recruits INT,
        avg_sale_per_member NUMERIC(10, 4),
        growth_rate DOUBLE PRECISION
);
```

Running query in 'postgresql+psycopg2://vincent:***@samplecluster.cq68pg38qszs.us-east-1.redshift.amazonaws.com:5439/mlm'

Out[9]:

List of Databases

In [10]:
```
%%sql
SELECT * FROM pg_database;
```

Running query in 'postgresql+psycopg2://vincent:***@samplecluster.cq68pg38qszs.us-east-1.redshift.amazonaws.com:5439/mlm'

7 rows affected.

Out[10]:

datname	datdba	encoding	datistemplate	datallowconn	datlastsysoid	datvacuumxid	datfrozenxid	dattablespace	datconfig	datacl
awsdatacatalog	1	6	False	False	102558	967	967	0	None	None
template0	1	6	True	False	102558	967	967	1663	None	{rdsdb=CTA/rdsdb}
dev	1	6	False	True	102558	0	0	1663	['enable_query_profiler_instrumentation=true']	None
padb_harvest	1	6	False	True	102558	0	0	1663	None	None
sys:internal	1	6	False	True	102558	967	967	1663	None	None
template1	1	6	True	True	102558	967	967	1663	None	{rdsdb=CTA/rdsdb}
mlm	101	6	False	True	102558	967	967	1663	None	None

List of tables in mlm database

In [11]:
```
%sql \dt
```

Running query in 'postgresql+psycopg2://vincent:***@samplecluster.cq68pg38qszs.us-east-1.redshift.amazonaws.com:5439/mlm'

Out[11]:

schema	name	type	owner
public	dim_cust_recruit_rep	table	vincent
public	dim_date	table	vincent
public	dim_location	table	vincent
public	dim_product	table	vincent
public	dim_team	table	vincent
public	dim_time	table	vincent
public	fact_recruitment	table	vincent
public	fact_sales	table	vincent
public	fact_team_perf	table	vincent

In [12]:
```
table_list = [
    "dim_date", "dim_time", "dim_cust_recruit_rep", "dim_team", "dim_product",
    "dim_location", "fact_sales", "fact_recruitment", "fact_team_perf"
]
```

Schema of tables

In [13]:
```
for table in table_list:
    print(f"{table}")
    schema_query = f"columns --table {table}"
    display(get_ipython().run_line_magic('sqlcmd', schema_query))
```

```
dim_date
```

name	type	nullable	default	autoincrement	comment
date_sk	INTEGER	True	None	False	None
year	INTEGER	True	None	False	None
month	VARCHAR(256)	True	None	False	None
day	INTEGER	True	None	False	None
day_of_week	VARCHAR(256)	True	None	False	None
quarter	INTEGER	True	None	False	None
is_weekend	BOOLEAN	True	None	False	None
is_holiday	BOOLEAN	True	None	False	None

`dim_time`

name	type	nullable	default	autoincrement	comment
time_sk	INTEGER	True	None	False	None
hour	INTEGER	True	None	False	None
minute	INTEGER	True	None	False	None
second	INTEGER	True	None	False	None

`dim_cust_recruit_rep`

name	type	nullable	default	autoincrement	comment
customer_sk	INTEGER	True	None	False	None
customer_id	VARCHAR(256)	True	None	False	None
name	VARCHAR(256)	True	None	False	None
age	INTEGER	True	None	False	None
gender	VARCHAR(256)	True	None	False	None
email	VARCHAR(256)	True	None	False	None
region	VARCHAR(256)	True	None	False	None
is_recruiter	BOOLEAN	True	None	False	None
is_sales_rep	BOOLEAN	True	None	False	None
recruited_by	INTEGER	True	None	False	None
team	INTEGER	True	None	False	None

`dim_team`

name	type	nullable	default	autoincrement	comment
team_sk	INTEGER	True	None	False	None
team_id	VARCHAR(256)	True	None	False	None
team_name	VARCHAR(256)	True	None	False	None
team_leader	VARCHAR(256)	True	None	False	None
total_members	INTEGER	True	None	False	None
region	VARCHAR(256)	True	None	False	None

`dim_product`

name	type	nullable	default	autoincrement	comment
product_sk	INTEGER	True	None	False	None
product_id	VARCHAR(256)	True	None	False	None
product_name	VARCHAR(256)	True	None	False	None
category	VARCHAR(256)	True	None	False	None
unit_price	NUMERIC(10, 4)	True	None	False	None
cost_price	NUMERIC(10, 4)	True	None	False	None

`dim_location`

name	type	nullable	default	autoincrement	comment
location_sk	INTEGER	True	None	False	None
location_id	VARCHAR(256)	True	None	False	None
region	VARCHAR(256)	True	None	False	None
city	VARCHAR(256)	True	None	False	None
brgy_no	INTEGER	True	None	False	None

```
fact_sales
```

name	type	nullable	default	autoincrement	comment
customer_sk	INTEGER	True	None	False	None
product_sk	INTEGER	True	None	False	None
location_sk	INTEGER	True	None	False	None
sales_rep_sk	INTEGER	True	None	False	None
date_sk	INTEGER	True	None	False	None
time_sk	INTEGER	True	None	False	None
transaction_id	VARCHAR(256)	True	None	False	None
quantity	INTEGER	True	None	False	None
total_sales_amount	NUMERIC(10, 4)	True	None	False	None
discounted_price	NUMERIC(10, 4)	True	None	False	None
commission_amount	NUMERIC(10, 4)	True	None	False	None

```
fact_recruitment
```

name	type	nullable	default	autoincrement	comment
recruiter_sk	INTEGER	True	None	False	None
new_customer_sk	INTEGER	True	None	False	None
date_sk	INTEGER	True	None	False	None
time_sk	INTEGER	True	None	False	None
recruitment_id	VARCHAR(256)	True	None	False	None
recruitment_signin_bonus	NUMERIC(10, 4)	True	None	False	None
salary_offered	NUMERIC(10, 4)	True	None	False	None
time_to_hire	INTEGER	True	None	False	None

```
fact_team_perf
```

name	type	nullable	default	autoincrement	comment
team_sk	INTEGER	True	None	False	None
date_sk	INTEGER	True	None	False	None
time_sk	INTEGER	True	None	False	None
location_sk	INTEGER	True	None	False	None
total_sales	NUMERIC(10, 4)	True	None	False	None
total_recruits	INTEGER	True	None	False	None
avg_sale_per_member	NUMERIC(10, 4)	True	None	False	None
growth_rate	DOUBLE_PRECISION	True	None	False	None

Faker to create dummy data

In [14]:
```python
fake = Faker()

def generate_dim_date():
    data = []
    for day in range(1, 30):  # February days
        date_sk = f"{day:02}02{fake.year()[2:]}"
        data.append((date_sk, fake.year(), 'February', day, fake.day_of_week(), fake.random_int(1, 4), bool(fake.random_int(0, 1)), bool(fake.random_int(0, 1))))
    return data

def generate_dim_time():
    data = []
    for hour in range(24):
```

```python
        for minute in range(60):
            for second in range(60):
                time_sk = f"{hour:02}{minute:02}{second:02}"
                data.append((time_sk, hour, minute, second))
    return data

def generate_dim_cust_recruit_rep(n=100):
    data = []
    for _ in range(n):
        customer_sk = fake.random_int(1000, 9999)
        data.append((customer_sk, str(uuid.uuid4()), fake.name(), fake.random_int(18, 65), fake.random_element(['Male', 'Female']),
                     fake.email(), fake.state(), bool(fake.random_int(0, 1)), bool(fake.random_int(0, 1)), fake.random_int(1000, 9999), fake.random_int(1, 50)))
    return data

def generate_dim_team(n=50):
    data = []
    for _ in range(n):
        team_sk = fake.random_int(100, 999)
        data.append((team_sk, str(uuid.uuid4()), fake.company(), fake.name(), fake.random_int(5, 50), fake.state()))
    return data

def generate_dim_product(n=100):
    data = []
    for _ in range(n):
        product_sk = fake.random_int(1000, 9999)
        data.append((product_sk, str(uuid.uuid4()), fake.word(), fake.random_element(['Electronics', 'Clothing', 'Food']), round(random.uniform(10, 500), 4), round(random.uniform(5, 250), 4)))
    return data

def generate_dim_location(n=50):
    data = []
    for _ in range(n):
        location_sk = fake.random_int(100, 999)
        data.append((location_sk, str(uuid.uuid4()), fake.state(), fake.city(), fake.random_int(1, 100)))
    return data

def generate_fact_sales(n=100):
    data = []
    for _ in range(n):
        data.append((fake.random_int(1000, 9999), fake.random_int(1000, 9999), fake.random_int(100, 999), fake.random_int(1000, 9999),
                     fake.random_int(101, 299), fake.random_int(0, 235959), str(uuid.uuid4()),
                     fake.random_int(1, 10), round(random.uniform(10, 1000), 4), round(random.uniform(5, 800), 4), round(random.uniform(1, 200), 4)))
    return data

def generate_fact_recruitment(n=100):
    data = []
    for _ in range(n):
        data.append((fake.random_int(1000, 9999), fake.random_int(1000, 9999), fake.random_int(101, 299),
                     fake.random_int(0, 235959), str(uuid.uuid4()), round(random.uniform(500, 5000), 4), round(random.uniform(30000, 100000), 4), fake.random_int(1, 60)))
    return data

def generate_fact_team_perf(n=50):
    data = []
    for _ in range(n):
        data.append((fake.random_int(100, 999), fake.random_int(101, 299), fake.random_int(0, 235959), fake.random_int(100, 999), round(random.uniform(1000, 50000), 4),
                     fake.random_int(1, 500), round(random.uniform(50, 5000), 4), round(random.uniform(-0.5, 1.5), 4)))
    return data

def save_to_redshift(df, table_name, engine):
    with engine.cursor() as cursor:
        cursor.write_dataframe(df, table_name)
        engine.commit()

df_dim_date = pd.DataFrame(generate_dim_date(),
                           columns=['date_sk', 'year', 'month', 'day', 'day_of_week', 'quarter', 'is_weekend', 'is_holiday'])
df_dim_time = pd.DataFrame(generate_dim_time(), columns=['time_sk', 'hour', 'minute', 'second']).head(100)
df_dim_cust_recruit_rep = pd.DataFrame(generate_dim_cust_recruit_rep(),
                                       columns=['customer_sk', 'customer_id', 'name', 'age', 'gender', 'email', 'region', 'is_recruiter', 'is_sales_rep', 'recruited_by', 'team'])
df_dim_team = pd.DataFrame(generate_dim_team(), columns=['team_sk', 'team_id', 'team_name', 'team_leader', 'total_members', 'region'])
df_dim_product = pd.DataFrame(generate_dim_product(), columns=['product_sk', 'product_id', 'product_name', 'category', 'unit_price', 'cost_price'])
df_dim_location = pd.DataFrame(generate_dim_location(), columns=['location_sk', 'location_id', 'region', 'city', 'brgy_no'])
df_fact_sales = pd.DataFrame(generate_fact_sales(),
                             columns=['customer_sk', 'product_sk', 'location_sk', 'sales_rep_sk', 'date_sk', 'time_sk', 'transaction_id', 'quantity', 'total_sales_amount',
                                      'discounted_price', 'commission_amount'])
df_fact_recruitment = pd.DataFrame(generate_fact_recruitment(), columns=['recruiter_sk', 'new_customer_sk', 'date_sk', 'time_sk', 'recruitment_id',
                                                                         'recruitment_signin_bonus', 'salary_offered', 'time_to_hire'])
df_fact_team_perf = pd.DataFrame(generate_fact_team_perf(), columns=['team_sk', 'date_sk', 'time_sk', 'location_sk', 'total_sales', 'total_recruits', 'avg_sale_per_member', 'growth_rate'])

save_to_redshift(df_dim_date, 'dim_date', engine)
save_to_redshift(df_dim_time.head(100), 'dim_time', engine)
save_to_redshift(df_dim_cust_recruit_rep, 'dim_cust_recruit_rep', engine)
save_to_redshift(df_dim_team, 'dim_team', engine)
save_to_redshift(df_dim_product, 'dim_product', engine)
save_to_redshift(df_dim_location, 'dim_location', engine)
save_to_redshift(df_fact_sales, 'fact_sales', engine)
```

```python
save_to_redshift(df_fact_recruitment, 'fact_recruitment', engine)
save_to_redshift(df_fact_team_perf, 'fact_team_perf', engine)
```

First 100 rows of each table

In [15]:
```python
for table in table_list:
    print(f"{table}")
    display(pd.read_sql(f"select * from {table}", engine))
```

dim_date

	date_sk	year	month	day	day_of_week	quarter	is_weekend	is_holiday
0	10202	1979	February	1	Monday	3	False	False
1	20220	1999	February	2	Thursday	2	True	True
2	30271	1993	February	3	Saturday	4	False	False
3	40221	1995	February	4	Tuesday	1	False	False
4	50298	1981	February	5	Thursday	4	True	False
5	60289	2021	February	6	Wednesday	1	True	True
6	70280	1998	February	7	Wednesday	3	False	True
7	80214	2023	February	8	Monday	4	False	True
8	90203	1990	February	9	Friday	2	True	False
9	100291	2018	February	10	Tuesday	4	True	False
10	110298	2013	February	11	Saturday	1	False	False
11	120213	1984	February	12	Sunday	1	False	False
12	130296	2005	February	13	Monday	4	False	False
13	140293	1980	February	14	Monday	2	True	True
14	150208	1991	February	15	Thursday	1	True	True
15	160209	2014	February	16	Friday	1	False	False
16	170282	2023	February	17	Friday	4	True	False
17	180205	2018	February	18	Tuesday	1	False	True
18	190294	1986	February	19	Monday	4	True	False
19	200275	1993	February	20	Sunday	4	False	True
20	210202	1976	February	21	Sunday	3	False	True
21	220206	1999	February	22	Tuesday	1	True	False
22	230294	1971	February	23	Wednesday	1	True	True
23	240223	1996	February	24	Tuesday	2	True	False
24	250207	1998	February	25	Sunday	4	False	True
25	260207	2015	February	26	Thursday	4	True	False
26	270286	2007	February	27	Sunday	4	True	False
27	280201	1995	February	28	Friday	1	False	False
28	290295	1996	February	29	Saturday	2	False	False

dim_time

	time_sk	hour	minute	second
0	0	0	0	0
1	1	0	0	1
2	2	0	0	2
3	3	0	0	3
4	4	0	0	4
...	...	...	...	...
95	135	0	1	35
96	136	0	1	36
97	137	0	1	37
98	138	0	1	38
99	139	0	1	39

100 rows × 4 columns

`dim_cust_recruit_rep`

	customer_sk	customer_id	name	age	gender	email	region	is_recruiter	is_sales_rep	recruited_by	team
0	5579	7a070299-2807-41aa-81c7-24e087c7fe29	Monica Bridges	57	Male	smithpeter@example.org	Illinois	False	True	9311	10
1	6161	2454dd9b-7c20-44e5-859c-c9364fca5ff5	Linda Miller	61	Male	stephen33@example.org	Connecticut	False	False	1683	22
2	6877	5d9f09ae-e91c-4ded-92c6-b733f9b06794	Bradley Khan	60	Female	wyattholly@example.org	Kentucky	True	True	3809	16
3	5096	2764e040-0786-4bfe-94b4-fc5878d3ea59	Dustin Brewer	20	Female	shieldskristina@example.net	Kansas	True	True	4866	15
4	9232	c08cfd84-e64f-4537-ba5c-684e9c8680bd	Christina Bauer	26	Female	millerjohn@example.com	Colorado	True	True	9638	40
...	...	...	...	...	...	...	...	...	...	...	...
95	4835	8b051b84-cb2b-4bde-b9af-1edcbe188988	Jonathan Wiggins	58	Male	sheila17@example.com	Michigan	False	True	3836	8
96	8977	b2a49f53-3b51-498e-a1cc-ef6a00cd9070	Nicole Smith	32	Female	kathleen02@example.net	Texas	False	False	9949	34
97	8527	f29bbc37-8546-4b2a-b0f7-cb8a1d62d0e4	Brandon Cook	27	Male	melissamercer@example.org	West Virginia	True	True	8907	26
98	5997	b847b59f-1dff-40d6-b5b0-4568f2347110	Erin Wood	21	Male	annettelawson@example.org	New Hampshire	False	True	6530	18
99	5350	a66f5e5d-debd-4c43-94c4-5346ac268cef	Allison Gutierrez	18	Female	meganwheeler@example.org	Virginia	False	False	6913	20

100 rows × 11 columns

`dim_team`

	team_sk	team_id	team_name	team_leader	total_members	region
0	708	8a8126e2-0158-404b-9ec6-de9e975d7bf3	Schneider, Cook and Preston	Preston Bishop	13	New York
1	933	48b3b67b-0383-4420-9ba0-258a39173560	Chapman-Singh	Ronald Anderson	5	Rhode Island
2	367	96a4c1c0-3fc4-4b54-9994-9b8cee14f886	Huang, Torres and Arnold	James Buckley	43	Alabama
3	788	dcb0a7ef-530f-4a03-9eea-6b2cc17efc72	Meza LLC	Amanda Curtis	25	Montana
4	489	0f56eadb-4b3e-441e-941d-37f14494e9b4	Mendez-Murphy	Anthony Collins	43	Oregon
5	666	5cdee403-f4d7-40b5-bca4-a7a5d76cfa4f	Edwards-Perry	Cassandra Randall	30	Iowa
6	716	c8c463a4-3a59-4e74-85ec-e671bd0bd5d8	Campbell LLC	Julie Hicks	17	Indiana
7	692	48021f97-ceae-4e68-99e7-74e7fc27d03b	Le, Hammond and Crawford	Courtney Smith	30	Minnesota
8	400	5e3f74c6-81ff-4a0a-a701-a87f96b94286	Adams-Scott	Paul Jackson	12	South Carolina
9	930	2b58e453-4371-4503-9f65-7a7014251f8b	David-Lopez	Jason Tran	30	South Dakota
10	165	387df63f-c902-423b-8a3f-4a09ac7d20cd	King-Russell	Ryan Stewart	49	New Hampshire
11	162	55fd77a8-3d0c-4469-ba36-f39d1c5b4f1f	Griffith, Blair and Hawkins	Ruth White	26	Nevada
12	546	518bb74b-fb8d-4e81-845a-cc296f2a73f7	Willis, Patel and Wilson	Jason Collins	36	Arkansas
13	776	8b946e60-ced4-49bd-95c9-f5b95ff8b0b4	Taylor, Fowler and Martinez	Andrew Meyers	8	Mississippi
14	452	03140976-52e1-4958-907c-fd867187c159	Herrera, Wallace and Cook	Joseph Hill	13	California
15	286	70616744-2364-4434-b3d2-b1bed89cf60a	Hunter, Bell and Green	Gerald Flores	45	Maine
16	575	f936ca63-9060-4d59-8f02-8c7b6d2f173d	Barrett, Dorsey and West	Brian Adams	25	Wyoming
17	974	665afb88-51ad-42af-9a60-ebb541eb5b97	Miller, Kim and Riggs	John Newman	49	Wyoming
18	806	81fd6eb7-719a-4b2a-aa13-46ee53bfeee5	Chandler Group	Matthew Evans	23	Delaware
19	648	98c6f058-b239-41e2-bb2e-9bc846543dff	Castillo, Schneider and Glenn	Heather Brown	44	North Carolina
20	584	d3f1e262-d1f6-45f3-8347-ed3cddac97eb	Hunter PLC	Kristin Holmes	33	Iowa
21	911	f07e4dd4-cf4c-4c93-b4e1-3edfe1e586ea	Irwin-Moss	Mark Mclaughlin	34	Michigan
22	970	c1209cfc-04b1-47d0-b730-0b03c2df9a38	Harris Inc	Brenda Stanley	36	Virginia
23	142	7c0c71cb-e87e-41c8-af14-e15b04b0b53a	Dunn PLC	Matthew Martin	43	Nevada
24	588	cdcf67e4-e220-443d-9e07-c5bdb439005f	Allen LLC	Thomas Abbott	41	Oklahoma
25	848	7ba9224f-c697-4582-a87d-ad716461d491	Martinez Ltd	Kevin Donovan	14	Utah
26	673	54c9f977-09ce-4991-8ce1-dfcce377298b	Rogers-Barnes	Vanessa Santiago	31	Oklahoma
27	787	27a8c646-3c05-4185-867d-8df9807728fb	Miller LLC	Benjamin Hoffman	29	Connecticut
28	619	fd538b42-3e7a-4c1e-bb06-58d7abb2beb9	Bishop Ltd	Joseph Salas	17	Indiana
29	499	60046978-58c5-4384-a9fe-52fd405b01e6	Hopkins PLC	Eddie Kline	11	New Jersey
30	330	e9d4c184-c9fe-4bfc-9365-9c2a64df151b	Burton-Burke	Brandy Butler	36	Vermont
31	187	3ade0a89-5046-4578-b173-b3c874b1d6d7	Miller, Griffin and Collins	Bobby Perkins	11	Virginia
32	158	1b5f0724-4852-4353-a096-60c1549e5f98	Webster, Sheppard and Jacobson	Danielle Allen	19	Nevada
33	729	c6b7326b-0365-4ba2-8d76-7ccfce446598	Vincent, Gould and Summers	Timothy Smith	16	Idaho
34	565	68d968aa-950f-48d1-9d65-94eaea973b23	Weiss, Gallagher and Conner	Cory Rodriguez	24	West Virginia
35	167	311e96e8-c0f8-482a-839f-63e5b53e17a6	Hood-Johns	Anna Hernandez	36	Michigan
36	431	385da68f-d669-4cee-acf4-acf3f36d1c6a	Turner Group	Patrick Mendez	9	Michigan
37	555	09afc215-ebc4-455a-b66f-2e8738312d01	Montgomery Inc	Dennis May	48	West Virginia
38	402	2a9139a9-4fab-4b81-9de0-a1d504405191	Davis-Turner	Kimberly Peck	20	Wisconsin
39	724	10a6db96-09d5-4672-b05d-45ffe22a9c31	Camacho LLC	Paul Hughes	50	Delaware
40	200	79a31085-f3e0-46a6-9738-354335b760a3	Hoover, Brown and Thompson	Brianna Bush	7	Wyoming
41	450	8b7a47de-f40b-43a2-b5a0-b619109dd503	Lang, Miller and Moreno	Wanda Woods	15	South Dakota
42	468	d73b534f-bcc8-43a0-8c20-18c086df5cf5	Barker-Morris	Daniel Wright	7	Louisiana
43	598	06ac376f-90e5-4682-9266-c2fe3b71fa78	Brown, Wright and Payne	Daniel Watson	18	Georgia
44	461	03779148-dc3e-436a-81eb-e932451d3fde	Bush-Fernandez	Jeffrey Moore	36	Ohio
45	454	b255da5b-9dd0-4dd8-ae56-0c7e3b83125f	Adams and Sons	Rebecca Key	15	South Dakota

	team_sk	team_id	team_name	team_leader	total_members	region
46	401	9098ab6b-d44e-4665-a80b-61a5d5d49f24	Ponce, Saunders and Wilson	Michelle Henry	49	North Dakota
47	226	32ffa78b-7fec-4e70-8d8f-2ebb378470be	Brewer, Blake and Powers	Austin Short	42	Mississippi
48	272	efd23b38-04af-4c7d-9d61-56253debec80	Edwards-Martinez	Jonathan Shannon	23	Texas
49	579	cba29237-e5fd-424d-a227-cc02c4cfc2ce	Mcpherson PLC	Tony Rodriguez	41	Illinois

`dim_product`

	product_sk	product_id	product_name	category	unit_price	cost_price
0	7926	61dd9450-684b-4385-a7f6-36eb898789c1	yeah	Food	304.6788	22.9929
1	9924	e9a215e6-6645-4cdd-853f-9afe9d2c2843	remain	Clothing	231.0320	81.6926
2	2729	9c4dbb9c-f7d3-4b55-ad26-9c71b874da7f	sport	Clothing	411.6383	91.5611
3	2533	855b42a8-eb7f-4b05-929e-59cc089a9ec3	sit	Food	122.7748	55.4194
4	1670	789bdd9c-2133-44d3-8422-cdb2c265d77c	short	Clothing	397.0455	136.0577
...	...	...	...	...	...	...
95	7236	da79613a-af14-45ba-b422-cfabfdf57816	agree	Electronics	230.5208	168.2802
96	8271	0cb2ec47-5fc8-4258-a87e-6218c7d42a5c	picture	Electronics	68.8582	25.4750
97	7554	30b9845a-66d8-4419-8ef7-0b9fa91652d3	yeah	Food	398.1844	212.7884
98	8543	21a34694-eccd-4efe-a8b6-77518c7409e1	student	Food	253.1189	106.4218
99	2528	5704ca0e-348a-4902-b759-4b09d044f827	hospital	Electronics	37.2279	195.8684

100 rows × 6 columns

`dim_location`

	location_sk	location_id	region	city	brgy_no
0	830	4c6452a6-6f8b-48f4-8954-408626ad4df8	Rhode Island	Carlland	2
1	957	e91a6ff9-9bb2-4fd6-afed-26d30232cb35	Virginia	Sallyton	57
2	992	98bb3151-2efe-44d9-bf0d-4cef54727c64	Maine	South Tiffany	35
3	524	40f3ec6d-966a-4909-8cd2-29d2ea49f3ac	Arkansas	Lake Paula	87
4	143	d0062110-e294-4217-98de-6abc98f4fc26	Maine	West Stephanie	73
5	972	1dd8d2e6-e145-4fe2-a860-896b47e72bdf	Oregon	Morristown	2
6	670	d3c016af-93c1-42f2-8207-ca67fc0d0927	California	Port Davidbury	86
7	744	566eeb62-ea80-4b1b-8441-93204d7a9055	Kansas	Longhaven	46
8	105	1d7c8c5c-1efd-4d50-9dea-10c19b1f6fa4	Massachusetts	Jessicashire	29
9	738	81d29de2-c454-49fd-b14c-9265984d63cb	Tennessee	West Jenniferland	72
10	689	dd202e49-2ff3-4784-837b-c9eeacf52681	Wyoming	New Brian	82
11	344	063fb02d-1625-4a16-b23a-43a0bbd47a16	West Virginia	Mariofort	83
12	238	ee6528d5-677b-4d10-b54f-0024037c5c91	Kansas	Brownborough	6
13	191	fd019893-1b4e-4979-b61f-bbd52fe61563	Texas	North Teresastad	68
14	491	600f8f28-55b4-4e74-ba4d-2d2a480b43ba	Ohio	South Amber	10
15	929	ed5926c5-537b-424f-8094-c1e7bc166795	Montana	Michaelmouth	81
16	893	cc9a5be7-6e85-4f56-895f-2641cc185503	North Dakota	Bethport	30
17	735	cd3b969e-578b-4f2e-9301-63c77d48b3c5	Maine	Royborough	99
18	655	4bcebcf4-69e3-431d-868a-1bcb7bdc93c6	Colorado	Davisside	85
19	290	0c5af289-76d6-4fc3-964f-780a03fad503	Kansas	Fowlershire	35
20	127	00ecc5be-e04f-4fa4-83e7-510d7f7d94d3	Texas	Lake Scotthaven	8
21	245	929d3d14-de62-4ffb-bed8-fa37ced25bf5	Maryland	Richardburgh	34
22	584	f7f57b7b-0cb4-4c2b-9a3c-077bfa97a678	Utah	South Annetown	81
23	623	dee04209-eff7-4375-8d1e-b11e3583f803	California	North Ryan	62
24	771	1bfb3ee1-003a-4662-ac55-e3ecfcdcf046	Michigan	Port Elizabethfort	51
25	586	8bbcd6bb-5cc0-4bf8-9839-794db8237ff3	North Carolina	Lake Stevenburgh	59
26	167	a5930f1c-879b-46a4-805e-dd8180e5f45a	Vermont	Lisachester	3
27	279	ddce9fb5-daf1-4894-ad44-8b8c0dfded5a	Illinois	North Tiffanyville	54
28	292	a372e7f5-30f1-4ee7-8603-1bc85529f591	South Carolina	New Samuel	56
29	620	f7d17296-0e43-4782-877d-5d50e40ffcb7	Tennessee	East Kimberly	17
30	161	23732cff-6854-4048-aa2f-522699cd8ec7	Ohio	Nicholasville	92
31	479	95061d6c-dad3-4e32-b8f9-ea654b06a603	Wyoming	Port Jacob	89
32	503	ad94fcd5-2e7d-45e5-827c-2adf499d5555	Washington	North Nathan	4
33	765	7d516336-2c45-4be7-a871-8025283a0c9d	Illinois	Wilsonborough	1
34	209	6551c6eb-1dbc-4f8e-99d2-50a7e2bcfda8	California	Gloverburgh	32
35	374	049b84cd-2e3f-46f6-9105-bbc3251508fa	Rhode Island	Grimesmouth	9
36	975	1f16c08f-29ec-40d5-a8ef-cb419a349d7c	New Jersey	Feliciaton	26
37	623	5f627075-b575-42d9-b046-1af8eed78b5a	Minnesota	North Randyburgh	79
38	383	fed023b4-451f-4206-a385-2450a3f8581e	California	Gregoryfurt	63
39	790	3422b5aa-f98d-4d05-b89a-54bdc5482e37	Louisiana	Port Luis	3
40	646	bf1198ff-99d9-42c3-b0e8-a872288a81e3	Louisiana	Victorstad	18
41	950	2bb80814-3658-40b1-874e-8a8e5b44367a	South Carolina	East Stacey	14
42	447	3e7cac28-a26f-4ab6-8a39-036c4725151b	Maryland	Port Lisaport	72
43	961	a5fb6204-be97-4cde-a4bc-539bd889ae9c	Indiana	Jeremyburgh	62
44	942	b2adab06-0b5b-472a-9348-d45934436655	Alabama	Lawrencehaven	96
45	687	443d8bac-5b99-42e9-b0fd-10645cadf1df	West Virginia	East Dominique	73

	location_sk	location_id	region	city	brgy_no
46	422	5fb336bb-1428-4ab5-bac6-f5a1164d0239	Pennsylvania	Toddborough	67
47	427	a7a19aac-767f-4fef-a58e-f3759bba478a	New York	Meganside	10
48	992	c2668fc1-23b2-41a0-ba95-2c11d194d96e	Louisiana	Maryburgh	66
49	912	d3969585-7688-44df-9754-0e1daa7d8a23	Alaska	Jimtown	59

`fact_sales`

	customer_sk	product_sk	location_sk	sales_rep_sk	date_sk	time_sk	transaction_id	quantity	total_sales_amount	discounted_price	commission_amount
0	3955	4608	943	5124	177	29750	42d22833-a766-4f88-8561-4212a3c7daae	7	199.0121	746.0681	55.2889
1	2286	8274	127	4343	289	110100	b28bd9cc-a6ef-43d1-9f1a-c235e4fdde35	1	693.6929	427.0241	178.6760
2	4188	1401	937	4671	276	199918	bd3b3431-86ff-4431-a10d-5cbf46eb353b	3	157.6499	350.9947	110.9702
3	5412	8916	292	2615	135	192024	2f4bf96b-01d9-4e80-be76-ca4938bab632	7	903.6272	353.5532	71.0142
4	9638	4710	658	6124	157	139597	efeb60c8-a37f-4823-a89e-da477a6f605c	9	344.9422	623.7854	198.5183
...	...	...	...	...	...	...	...	...	...	...	...
95	2213	3251	167	2289	179	54599	659d8ad9-5c08-496b-bce5-3369c2b3e951	4	728.6556	493.6816	6.6093
96	4918	9960	246	4276	126	35462	75ea8835-82fb-4ea8-ac02-63d039e25c80	1	637.6421	535.2821	181.3705
97	8923	5794	202	7793	191	111572	763f206a-d09a-4913-bd9c-7fe5053daef9	7	913.0265	744.1450	45.2883
98	2199	1843	558	2362	179	168015	33895538-2d5b-4c5a-aa5e-2e80946c8cff	4	169.4959	401.3938	161.1486
99	7890	8277	130	3410	241	72846	6b937211-4e7b-47ae-9cb1-99ac69bfc8fa	6	796.3160	187.1548	34.6234

100 rows × 11 columns

`fact_recruitment`

	recruiter_sk	new_customer_sk	date_sk	time_sk	recruitment_id	recruitment_signin_bonus	salary_offered	time_to_hire
0	6843	1884	149	8929	8d14d130-39bf-4660-9523-eef25e47711d	926.8707	98700.0081	56
1	4598	7606	175	173341	d93e9e87-d1d3-42ad-adf6-4c880f3c06f4	2464.5531	49177.1993	35
2	2686	4977	177	160485	446589f8-ddd5-4766-b8b5-0f85cc88a026	3166.1594	50997.6965	41
3	5630	3646	145	181655	1f243301-fb0c-437f-a57e-6a440e0dc5f4	2297.7647	98543.5407	25
4	7692	1647	233	154920	04576846-18de-4f24-a181-c4ff76d508b2	4096.7034	34462.9647	26
...	...	...	...	...	...	...	...	...
95	9618	8165	190	51356	254f2973-4bab-4829-a161-b188c29b9f61	724.6629	95369.3581	16
96	4953	2099	242	212497	daaadf27-2fc0-4e00-b73a-16ad0e98bde4	4142.6088	66264.4258	33
97	1943	8661	220	55622	f3faf6f6-a9e4-4b18-97e5-e770985ebe97	4225.8292	74478.7679	23
98	4644	6373	137	219513	ed0331a7-ff89-4315-baa2-720b1bcf4b21	3586.2737	60002.4670	26
99	3093	1176	145	148859	0fa272ed-9ee2-4949-8cd7-3f9069aca5b2	1640.6516	62139.5771	7

100 rows × 8 columns

`fact_team_perf`

	team_sk	date_sk	time_sk	location_sk	total_sales	total_recruits	avg_sale_per_member	growth_rate
0	977	210	91477	903	40416.5037	68	3832.0172	-0.2062
1	620	193	191294	154	40571.1844	494	2435.7806	-0.1231
2	591	165	159241	870	12929.5333	72	1432.2971	1.1747
3	755	289	187998	513	10131.9441	180	3005.8052	0.0169
4	793	281	110370	918	42622.4431	221	1069.7222	-0.2653
5	245	130	87844	693	2525.1506	275	1524.1580	-0.4932
6	469	198	44565	945	2853.9631	496	2243.1680	0.9151
7	303	142	222429	947	8632.1807	110	1775.1672	1.4668
8	338	126	172860	479	11726.8062	495	3821.1974	1.2098
9	123	150	13345	130	18248.2930	101	1927.4724	-0.3580
10	889	210	156132	266	16696.4556	324	1583.7184	0.6402
11	518	297	25357	650	44574.5428	202	2927.4067	0.9580
12	541	275	170052	314	26625.2130	105	2752.4705	1.4965
13	768	171	165483	676	6101.0900	396	2535.1039	1.1727
14	980	113	43880	644	32718.0129	135	1160.7105	1.1254
15	170	124	215480	500	34466.4504	414	4313.5833	0.0827
16	782	287	164811	148	1021.3683	494	756.3597	-0.3948
17	624	214	226519	373	25223.4726	150	1150.3244	-0.2542
18	724	263	206740	844	23870.8957	199	1906.0446	0.6266
19	997	141	52628	514	1480.2845	223	265.5959	1.3293
20	578	168	142412	814	11667.5178	217	4722.9387	1.4270
21	260	180	198057	934	1234.8992	346	1970.9428	-0.3897
22	289	120	193365	954	39979.4783	388	3371.4247	1.4398
23	125	288	29683	867	32917.0396	492	825.8104	0.6224
24	482	125	142257	282	24732.3875	469	4621.8078	-0.4872
25	373	296	11915	224	27350.1063	346	3635.9514	-0.3130
26	127	115	201332	559	36652.3862	64	629.1723	-0.0331
27	984	256	63230	386	7711.6732	191	2520.1986	1.1805
28	643	224	35161	622	26640.8709	52	4775.4038	1.2256
29	389	153	147207	166	5224.2291	205	1067.9169	0.7236
30	382	121	44887	857	14388.0367	359	438.1724	0.3191
31	259	147	49765	212	40449.4300	482	2001.1924	-0.4039
32	365	162	9012	866	28491.5277	421	4016.3147	-0.4173
33	562	226	168179	445	11422.8552	132	4724.8709	1.1600
34	179	200	98759	416	47291.1839	460	622.5098	-0.0046
35	587	282	80920	935	38662.0722	365	575.2417	0.6073
36	227	212	34069	749	33943.1075	198	2569.0151	1.3649
37	922	148	187693	304	48833.9877	37	2489.0394	-0.4475
38	630	275	110848	157	47995.2591	34	2620.7789	1.3070
39	837	191	73971	704	4384.7574	424	1873.4297	0.1849
40	949	194	50346	809	41940.5185	159	1773.4797	1.4616
41	178	237	105188	813	14823.6198	343	1988.7581	0.4521
42	598	232	4341	218	27635.8563	86	1474.5794	1.1532
43	869	108	108829	323	33594.0734	260	873.7590	0.8954
44	222	265	14424	717	15897.8967	160	3238.1379	0.9657
45	664	119	121710	535	35333.5326	225	1759.3843	1.1696

	team_sk	date_sk	time_sk	location_sk	total_sales	total_recruits	avg_sale_per_member	growth_rate
46	629	198	49321	755	6849.1887	167	4019.9317	-0.4930
47	379	101	75635	154	13621.0082	311	3666.7396	0.1020
48	308	229	194549	726	49486.5922	327	1241.4614	0.5380
49	983	194	211839	847	33705.7744	69	1225.2623	1.3819

Number of rows of each table

In [16]:
```
for table in table_list:
    print(f"{table}")
    count_query = f"select COUNT(*) from {table}"
    display(get_ipython().run_line_magic('sql', count_query))
```

```
dim_date
```
Running query in 'postgresql+psycopg2://vincent:***@samplecluster.cq68pg38qszs.us-east-1.redshift.amazonaws.com:5439/mlm'

1 rows affected.

count
29

```
dim_time
```
Running query in 'postgresql+psycopg2://vincent:***@samplecluster.cq68pg38qszs.us-east-1.redshift.amazonaws.com:5439/mlm'

1 rows affected.

count
100

```
dim_cust_recruit_rep
```
Running query in 'postgresql+psycopg2://vincent:***@samplecluster.cq68pg38qszs.us-east-1.redshift.amazonaws.com:5439/mlm'

1 rows affected.

count
100

```
dim_team
```
Running query in 'postgresql+psycopg2://vincent:***@samplecluster.cq68pg38qszs.us-east-1.redshift.amazonaws.com:5439/mlm'

1 rows affected.

count
50

```
dim_product
```
Running query in 'postgresql+psycopg2://vincent:***@samplecluster.cq68pg38qszs.us-east-1.redshift.amazonaws.com:5439/mlm'

1 rows affected.

count
100

```
dim_location
```
Running query in 'postgresql+psycopg2://vincent:***@samplecluster.cq68pg38qszs.us-east-1.redshift.amazonaws.com:5439/mlm'

1 rows affected.

count
50

```
fact_sales
```
Running query in 'postgresql+psycopg2://vincent:***@samplecluster.cq68pg38qszs.us-east-1.redshift.amazonaws.com:5439/mlm'

1 rows affected.

count
100

```
fact_recruitment
```
Running query in 'postgresql+psycopg2://vincent:***@samplecluster.cq68pg38qszs.us-east-1.redshift.amazonaws.com:5439/mlm'

1 rows affected.

count
100

```
fact_team_perf
```
Running query in 'postgresql+psycopg2://vincent:***@samplecluster.cq68pg38qszs.us-east-1.redshift.amazonaws.com:5439/mlm'

1 rows affected.

count
50

Rollback/Start from scratch queries

In [17]:

```
# %%sql
# DROP TABLE IF EXISTS
#     dim_team, dim_date, dim_time, dim_cust_recruit_rep,
#     dim_product, dim_location, fact_sales, fact_recruitment,
#     fact_team_perf
# CASCADE;
```