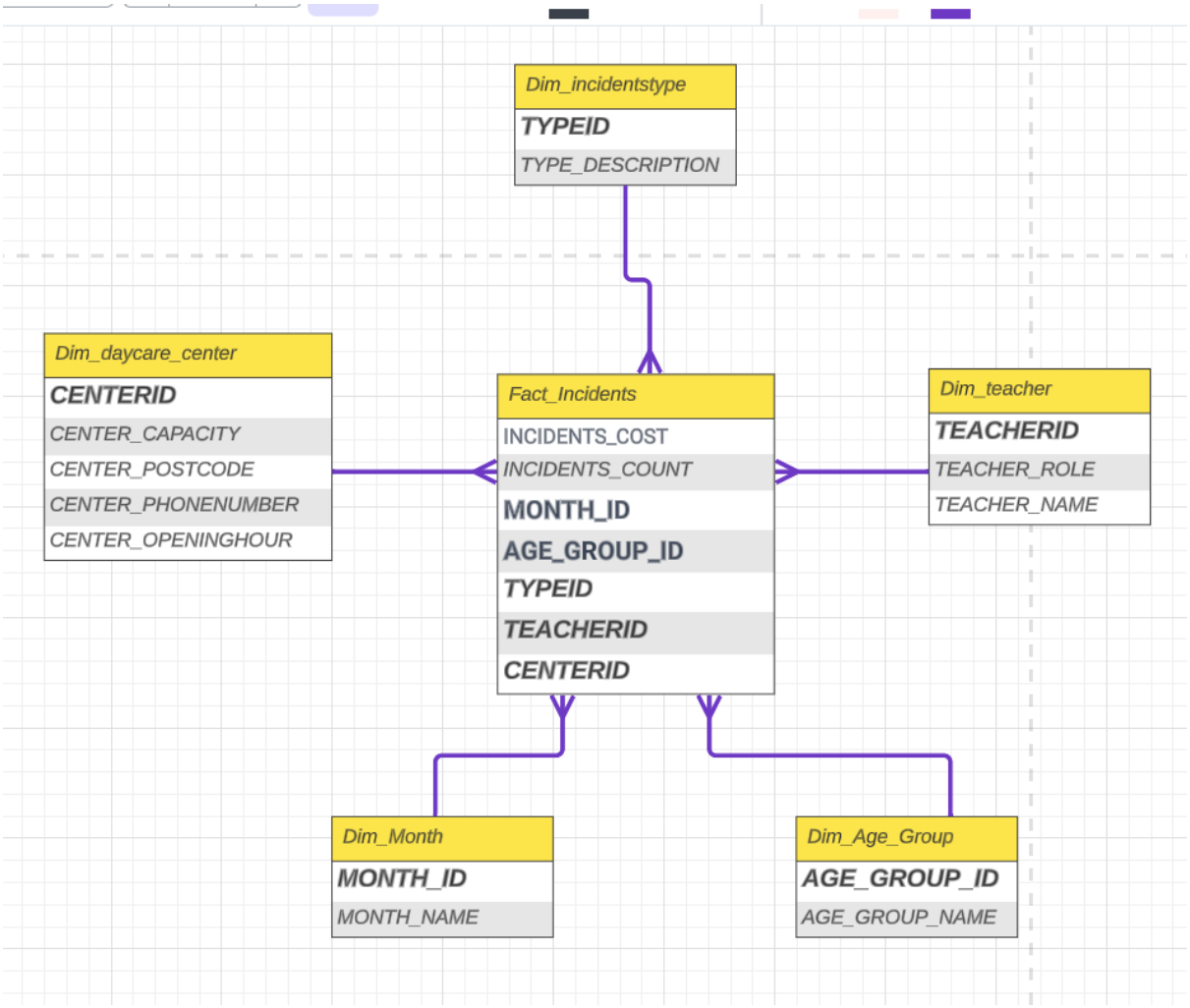
FIT5137 S2 2023 Assignment 1
Take home test

Shixin Huang 31029248 2023-08-12

The star schema diagram – Task 1 -28pts

Star schema Diagram

https://lucid.app/lucidchart/f4bcd10f-9f25-4f62-89f3-0867bc80d665/edit?viewport_loc=-1082%2C-1466%2C2455%2C1252%2Cxs1j6J_YewBt&invitationId=inv_b5358056-2f9a-455b-961a-c507f7692c75



Explanation for your design choices

1. Dimensional Modeling: The chosen design follows the dimensional modeling approach, which is ideal for analytical purposes. Dimension tables (e.g., Dim_Incident_Type, Dim_Month, Dim_Daycare_Center, Dim_Teacher) hold the descriptive attributes that provide context and enable grouping and filtering of data. The fact table (Fact_Incidents) captures the measures (number of incidents, incident cost) along with foreign keys to connect with dimension tables.
2. Granularity: The chosen granularity in the fact table is at the incident level, allowing for a detailed analysis of each incident. This level of granularity enables drill-down capabilities to investigate incidents by various perspectives.
3. Dimension Hierarchies: The Age Group dimension (Dim_Age_Group) includes two hierarchies - Pre-Kinder (1-2 years old) and Kinder (3-5 years old). This design choice enables flexible analysis at different levels of the age group hierarchy, providing a drill-down capability as requested.
4. Foreign Keys: The use of foreign keys in the fact table establishes relationships with dimension tables. This relationship allows for easy integration and analysis across different dimensions, such as linking incident types, months, daycare centers, teachers, and age groups
5. Aggregation and Summarization: The fact table stores the aggregated measures of the number of incidents and incident cost. This design choice allows for efficient analysis by directly querying the fact table, avoiding the need for lengthy calculations during runtime.

The Two-Column Table Methodology illustration – Task 2-8pts

Dim_teacher		
TEACHERID	*TEACHER_ROLE*	*TEACHER_NAME*
1	math	joe
2	pe	taylor

Dim_incidentstype	
TYPEID	*TYPE_DESCRIPTION*
1	Abrasion & Scrape
2	Asthma & respiratory

Dim_daycare_center				
CENTERID	CENTER_CAPACITY	CENTER_POSTCODE	CENTER_PHONENUMBER	CENTER_OPENINGHOUR
CE1	200	3004	1800978429	9AM-5PM
CE2	200	3131	1300168881	9AM-5PM
CE3	200	3068	1800222543	9AM-5PM

Dim_Age_Group	
AGE_GROUP_ID	AGE_GROUP_NAME
1	Pre-Kinder
2	Kinder

DIM_MONTH	
MONTH_ID	MONTH_NAME
1	JAN
2	FEB
3	MAR

Fact_Incidents						
MONTH_ID	TYPEID	AGE_GROUP_ID	TEACHERID	CENTERID	INCIDENTS_COUNT	INCIDENTS_COST
3	T2	1	TE14	CE3	1	187
2	T4	1	TE7	CE2	1	31

Data warehouse implementation - task 3-44pts

I already copy all the tables from the MonChild account. So in all the queries below, I don't need the prefix for any query. Please execute those parts in my SQL script header before creating any of the dim and fact tables.

Create dimension tables

```
-- Create Dim_Incident_Type
CREATE TABLE Dim_Incident_Type AS
SELECT * FROM incidentstype;
select * from Dim_Incident_Type;
```

```sql
-- Create Dim_Incident_Type
CREATE TABLE Dim_Incident_Type AS
SELECT * FROM incidentstype;
select * from Dim_Incident_Type;
```

CREATE

Output | S31029248.DIM_INCIDENT_TYPE

6 rows

	TYPEID	TYPE_DESCRIPTION
1	T1	Abrasion & Scrape
2	T2	Amputaion
3	T3	Asthma & respiratory
4	T4	Broken bone & fracture & dislocation
5	T5	Electric shock
6	T6	High temperature

```sql
-- Create Dim_Month
CREATE TABLE Dim_Month AS
SELECT DISTINCT EXTRACT(MONTH FROM INCIDENT_DATE) AS MONTH_ID,
                TO_CHAR(INCIDENT_DATE, 'MON') AS MONTH_NAME
FROM children_incidents;
select * from Dim_Month;
```

```
-- Create Dim_Month
CREATE TABLE Dim_Month AS
SELECT DISTINCT EXTRACT(MONTH FROM INCIDENT_DATE) AS MONTH_ID,
                TO_CHAR(INCIDENT_DATE, 'MON') AS MONTH_NAME
FROM children_incidents;
select * from Dim_Month;

-- Create Dim_Daycare_Center
CREATE TABLE Dim_Daycare_Center AS
```

Output | S31029248.DIM_MONTH

8 rows

	MONTH_ID	MONTH_NAME
1	8	AUG
2	4	APR
3	2	FEB
4	1	JAN
5	7	JUL
6	5	MAY
7	3	MAR
8	6	JUN

```
-- Create Dim_Daycare_Center
CREATE TABLE Dim_Daycare_Center AS
SELECT * FROM daycare_center;
select * from Dim_Daycare_Center;
```

```sql
-- Create Dim_Daycare_Center
CREATE TABLE Dim_Daycare_Center AS
SELECT * FROM daycare_center;
select * from Dim_Daycare_Center;

-- Create Dim_Teacher
CREATE TABLE Dim_Teacher AS
SELECT * FROM teacher;
```

Output | S31029248.DIM_DAYCARE_CENTER

3 rows | Tx: Manual

	CENTERID	CENTER_CAPACITY	CENTER_POSTCODE	CENTER
1	CE1	200	3004	
2	CE2	200	3131	
3	CE3	200	3068	

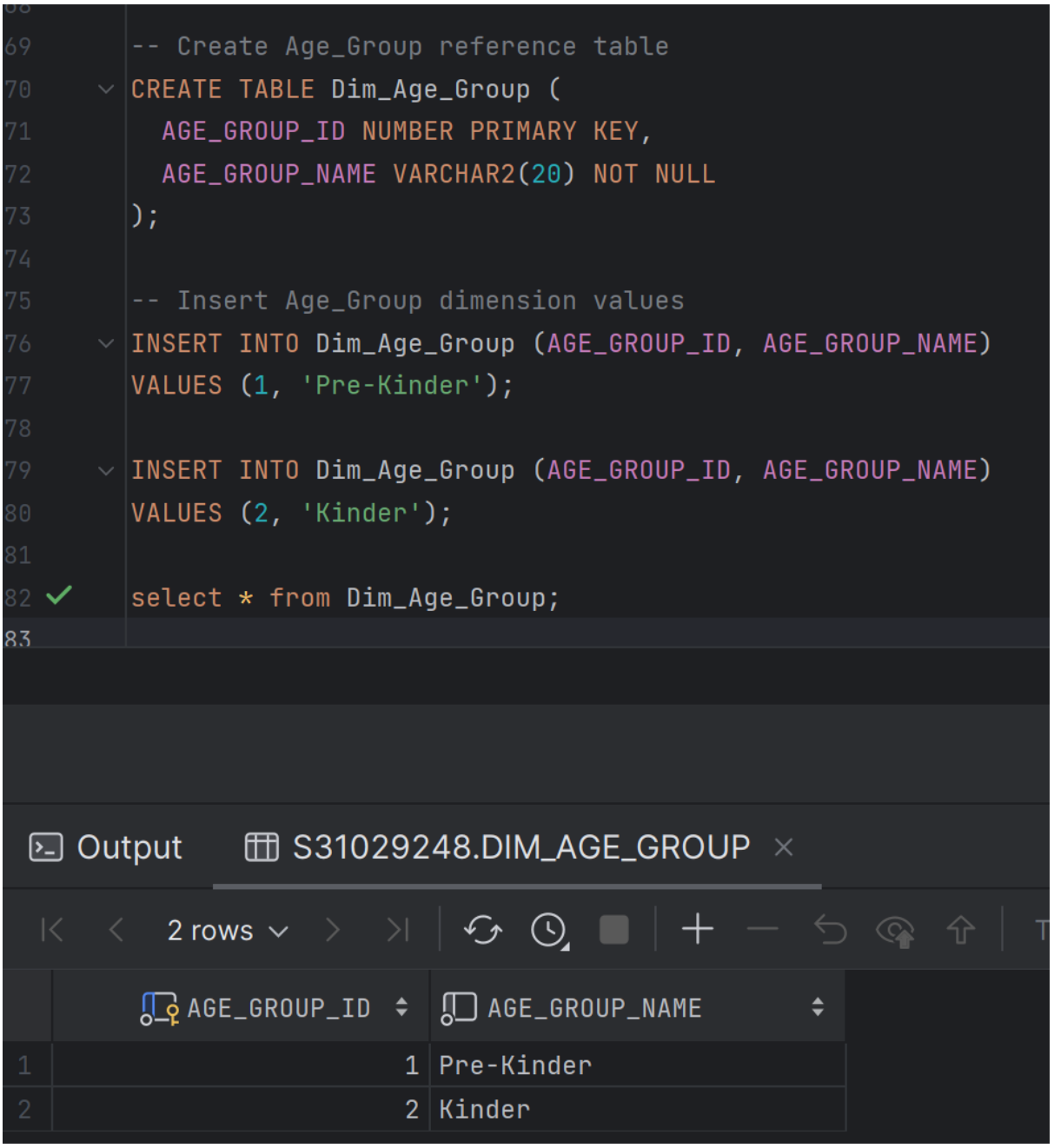
```sql
-- Create Dim_Teacher
CREATE TABLE Dim_Teacher AS
SELECT * FROM teacher;
select * from Dim_Teacher;
```

Output | S31029248.DIM_TEACHER

16 rows | Tx: Manual

	TEACHERID	TEACHER_ROLE	TEACHER_NAME
1	TE1	Early childhood teacher	Arthur Lyu
2	TE2	Assistant educator	Kyler Hardin
3	TE3	Assistant educator	Simeon Vaughn
4	TE4	Assistant educator	Gabriela Sims
5	TE5	Early childhood teacher	Madelynn Obrien
6	TE6	Assistant educator	Kaydence House
7	TE7	Early childhood teacher	Juliette Flores
8	TE8	Early childhood teacher	Markus Hanna
9	TE9	Assistant educator	Nicole Powell

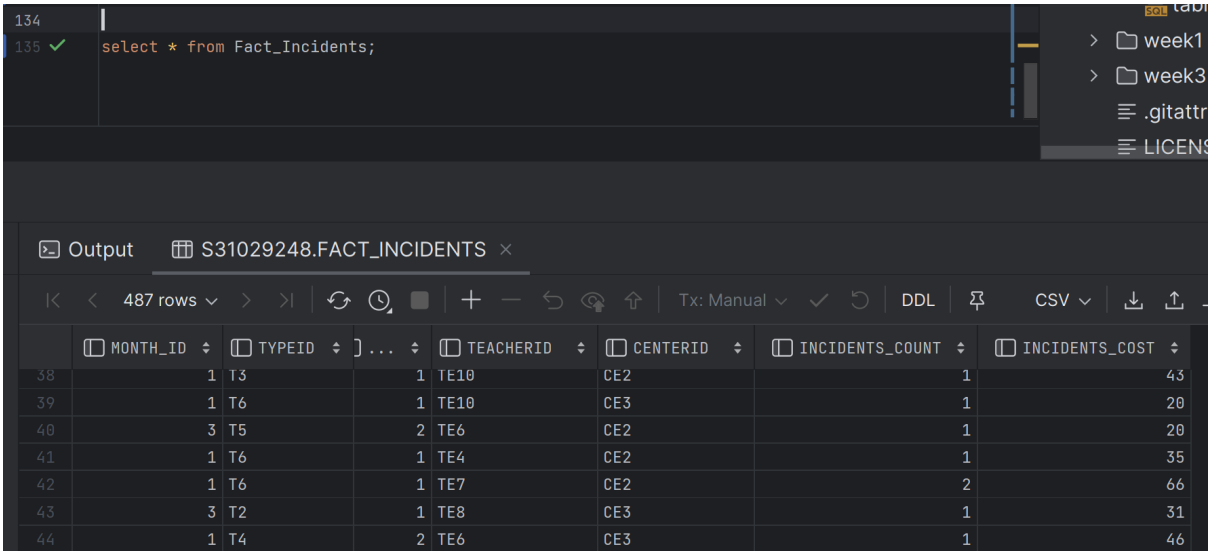
```sql
-- Create Age_Group reference table
CREATE TABLE Dim_Age_Group (
 AGE_GROUP_ID NUMBER PRIMARY KEY,
 AGE_GROUP_NAME VARCHAR2(20) NOT NULL
);

-- Insert Age_Group dimension values
INSERT INTO Dim_Age_Group (AGE_GROUP_ID, AGE_GROUP_NAME)
VALUES (1, 'Pre-Kinder');

INSERT INTO Dim_Age_Group (AGE_GROUP_ID, AGE_GROUP_NAME)
VALUES (2, 'Kinder');

select * from Dim_Age_Group;
```



Create Fact table

```
-- create fact test table
drop table fact_test;

CREATE TABLE fact_test AS
select ci.INCIDENTID,
      ci.INCIDENTS_COST,
      ci.TYPEID,
      ci.TEACHERID,
      dc.CENTERID,
      c.CHILD_AGE,
      EXTRACT(MONTH FROM ci.INCIDENT_DATE) AS MONTH_ID
from children_incidents ci
        join children c on ci.CHILDRENID = c.CHILDRENID
        join daycare_center dc on c.CENTERID = dc.CENTERID;

select * from fact_test;

-- Alter Fact_Test table to add AGE_GROUP_ID
ALTER TABLE fact_test
ADD AGE_GROUP_ID NUMBER;

-- Update AGE_GROUP_ID based on CHILD_AGE
UPDATE fact_test
SET AGE_GROUP_ID = CASE
                       WHEN CHILD_AGE BETWEEN 1 AND 2 THEN 1
                       WHEN CHILD_AGE BETWEEN 3 AND 5 THEN 2
                   END;

select * from fact_test;

-- create Fact_Incidents table from fact_test
CREATE TABLE Fact_Incidents AS
select
   month_id,
   TYPEID,
   AGE_GROUP_ID,
   TEACHERID,
   CENTERID,
   count(INCIDENTID) AS INCIDENTS_COUNT,
   SUM(INCIDENTS_COST) AS INCIDENTS_COST

from fact_test

group by
   month_id,
   TYPEID,
   AGE_GROUP_ID,
   TEACHERID,
   CENTERID;
```

```
select * from Fact_Incidents;
```



SQL query to answer the question - task4-15pts

A. Show the total number of incidents and total incident costs by age group.

```
select
   AGE_GROUP_NAME,
   SUM(INCIDENTS_COUNT) AS TOTAL_INCIDENTS_COUNT,
   SUM(INCIDENTS_COST) AS TOTAL_INCIDENTS_COST
from Fact_Incidents fi
        join Dim_Age_Group dag on fi.AGE_GROUP_ID = dag.AGE_GROUP_ID
group by AGE_GROUP_NAME;
```

```sql
-- A. Show the total number of incidents and total incident costs by age group
select
    AGE_GROUP_NAME,
    SUM(INCIDENTS_COUNT) AS TOTAL_INCIDENTS_COUNT,
    SUM(INCIDENTS_COST) AS TOTAL_INCIDENTS_COST
from Fact_Incidents fi
        join Dim_Age_Group dag on fi.AGE_GROUP_ID = dag.AGE_GROUP_ID
group by AGE_GROUP_NAME;
```

Output | Result 48

2 rows

	AGE_GROUP_NAME	TOTAL_INCIDENTS_COUNT	TOTAL_INCIDENTS_COST
1	Pre-Kinder	222	14758
2	Kinder	314	21723

B. Show the total number of incidents and total incident costs for the teachers whose roles are 'Early childhood teacher'.

```sql
--Show the total number of incidents and total incident costs
-- for the teachers whose roles are Early childhood teacher.
select
   t.TEACHER_NAME,
   SUM(INCIDENTS_COUNT) AS TOTAL_INCIDENTS_COUNT,
   SUM(INCIDENTS_COST) AS TOTAL_INCIDENTS_COST
from Fact_Incidents fi
        join Dim_Teacher t on fi.TEACHERID = t.TEACHERID
where t.TEACHER_ROLE = 'Early childhood teacher'
group by t.TEACHER_NAME;
```

```sql
--Show the total number of incidents and total incident costs
-- for the teachers whose roles are Early childhood teacher.
select
    t.TEACHER_NAME,
    SUM(INCIDENTS_COUNT) AS TOTAL_INCIDENTS_COUNT,
    SUM(INCIDENTS_COST) AS TOTAL_INCIDENTS_COST
from Fact_Incidents fi
        join Dim_Teacher t on fi.TEACHERID = t.TEACHERID
where t.TEACHER_ROLE = 'Early childhood teacher'
group by t.TEACHER_NAME;
```

Output | Result 51

6 rows

	TEACHER_NAME	TOTAL_INCIDENTS_COUNT	TOTAL_INCIDENTS_COST
1	Cade Stark	49	3281
2	Juliette Flores	44	2812
3	Markus Hanna	39	2465
4	Donovan Hill	37	2860
5	Arthur Lyu	10	438
6	Madelynn Obrien	50	2908

C. Show the total number of incidents and total incident costs by incident type in May.

```sql
--Show the total number of incidents and total incident costs by incident
type in May.
SELECT it.TYPEID,
      SUM(INCIDENTS_COUNT) AS TOTAL_INCIDENTS_COUNT,
      SUM(INCIDENTS_COST) AS TOTAL_INCIDENTS_COST
FROM Fact_Incidents fi
JOIN Dim_Incident_Type it ON fi.TYPEID = it.TYPEID
JOIN Dim_Month dm ON fi.MONTH_ID = dm.MONTH_ID
where dm.MONTH_NAME = 'MAY'
GROUP BY it.TYPEID
order by TYPEID asc;
```

```sql
--Show the total number of incidents and total incident costs by incident type in May.
SELECT it.TYPEID,
       SUM(INCIDENTS_COUNT) AS TOTAL_INCIDENTS_COUNT,
       SUM(INCIDENTS_COST) AS TOTAL_INCIDENTS_COST
FROM Fact_Incidents fi
JOIN Dim_Incident_Type it ON fi.TYPEID = it.TYPEID
JOIN Dim_Month dm ON fi.MONTH_ID = dm.MONTH_ID
where dm.MONTH_NAME = 'MAY'
GROUP BY it.TYPEID
order by TYPEID asc;
```

Output | Result 76

6 rows

	TYPEID	TOTAL_INCIDENTS_COUNT	TOTAL_INCIDENTS_COST
1	T1	1	58
2	T2	5	568
3	T3	7	846
4	T4	6	613
5	T5	8	678

D. Show the total number of incidents and total incident costs by daycare center.

```sql
--Show the total number of incidents and total incident costs by daycare
center.
select
   dc.CENTERID,
   SUM(INCIDENTS_COUNT) AS TOTAL_INCIDENTS_COUNT,
   SUM(INCIDENTS_COST) AS TOTAL_INCIDENTS_COST
from Fact_Incidents fi
        join Dim_Daycare_Center dc on fi.CENTERID = dc.CENTERID
group by dc.CENTERID
order by CENTERID ASC;
```

```sql
--Show the total number of incidents and total incident costs by daycare center.
select
    dc.CENTERID,
    SUM(INCIDENTS_COUNT) AS TOTAL_INCIDENTS_COUNT,
    SUM(INCIDENTS_COST) AS TOTAL_INCIDENTS_COST
from Fact_Incidents fi
        join Dim_Daycare_Center dc on fi.CENTERID = dc.CENTERID
group by dc.CENTERID
order by CENTERID ASC;
```

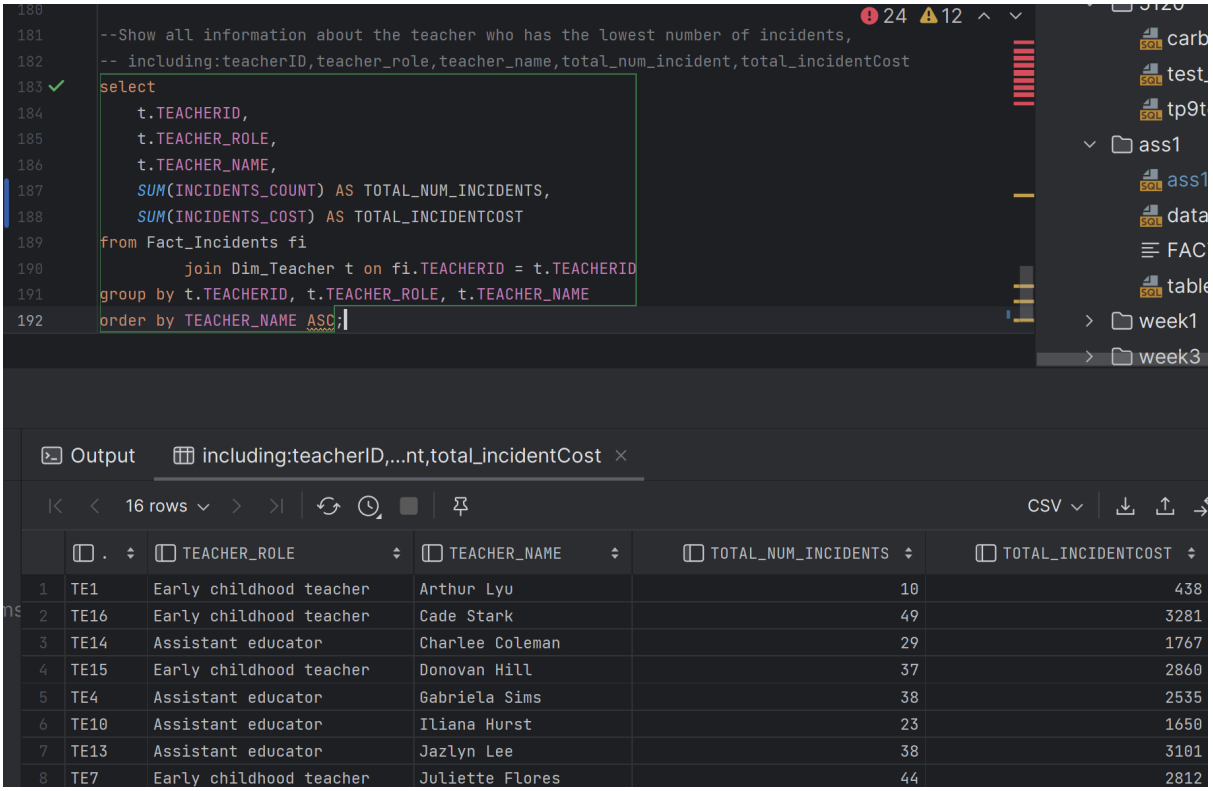
Output | Result 57

3 rows

	CENTERID	TOTAL_INCIDENTS_COUNT	TOTAL_INCIDENTS_COST
1	CE1	218	16147
2	CE2	169	11078
3	CE3	149	9256

E. Show all information about the teacher who has the lowest number of incidents, including:teacherID,teacher_role,teacher_name,total_num_incident,total_incidentCost

```sql
--Show all information about the teacher who has the lowest number of
incidents,
--
including:teacherID,teacher_role,teacher_name,total_num_incident,total_incide
ntCost
select
   t.TEACHERID,
   t.TEACHER_ROLE,
   t.TEACHER_NAME,
   SUM(INCIDENTS_COUNT) AS TOTAL_NUM_INCIDENTS,
   SUM(INCIDENTS_COST) AS TOTAL_INCIDENTCOST
from Fact_Incidents fi
        join Dim_Teacher t on fi.TEACHERID = t.TEACHERID
group by t.TEACHERID, t.TEACHER_ROLE, t.TEACHER_NAME
order by TEACHER_NAME ASC;
```



```
--Show all information about the teacher who has the lowest number of incidents,
-- including:teacherID,teacher_role,teacher_name,total_num_incident,total_incidentCost
select
    t.TEACHERID,
    t.TEACHER_ROLE,
    t.TEACHER_NAME,
    SUM(INCIDENTS_COUNT) AS TOTAL_NUM_INCIDENTS,
    SUM(INCIDENTS_COST) AS TOTAL_INCIDENTCOST
from Fact_Incidents fi
        join Dim_Teacher t on fi.TEACHERID = t.TEACHERID
group by t.TEACHERID, t.TEACHER_ROLE, t.TEACHER_NAME
order by TEACHER_NAME ASC;
```

Output | including:teacherID,...nt,total_incidentCost

16 rows | CSV

.	TEACHER_ROLE	TEACHER_NAME	TOTAL_NUM_INCIDENTS	TOTAL_INCIDENTCOST
1	TE1	Early childhood teacher	Arthur Lyu	10	438
2	TE16	Early childhood teacher	Cade Stark	49	3281
3	TE14	Assistant educator	Charlee Coleman	29	1767
4	TE15	Early childhood teacher	Donovan Hill	37	2860
5	TE4	Assistant educator	Gabriela Sims	38	2535
6	TE10	Assistant educator	Iliana Hurst	23	1650
7	TE13	Assistant educator	Jazlyn Lee	38	3101
8	TE7	Early childhood teacher	Juliette Flores	44	2812