2. Create the table structure with appropriate data types before loading with SQL Loader?

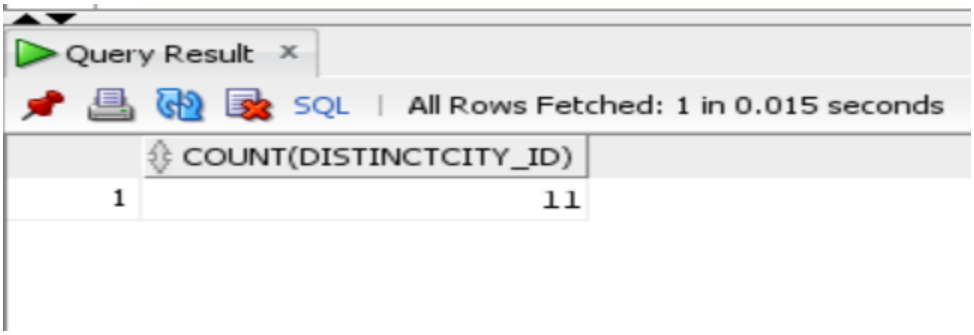
```
create table dim_city (city_id number,city_name varchar2(50),country varchar2(20));

create table fact_trip (trip_uuid varchar2(50),datestr date,product_type_name
varchar2(50),city_id number,driver_uuid varchar2(50),is_completed varchar2(25),eta
number,ata number,ufp_fare float,fare_final float);

ALTER TABLE dim_city ADD CONSTRAINT pk_dim_city PRIMARY KEY(CITY_ID);

ALTER TABLE fact_trip ADD CONSTRAINT pk_fact_trip PRIMARY KEY(TRIP_UUID);
```

3. Answer the following questions

a. How many city_ids does uberPOOL operate in?

```
select count(DISTINCT city_id)
from fact_trip
where product_type_name='uberPOOL';
```

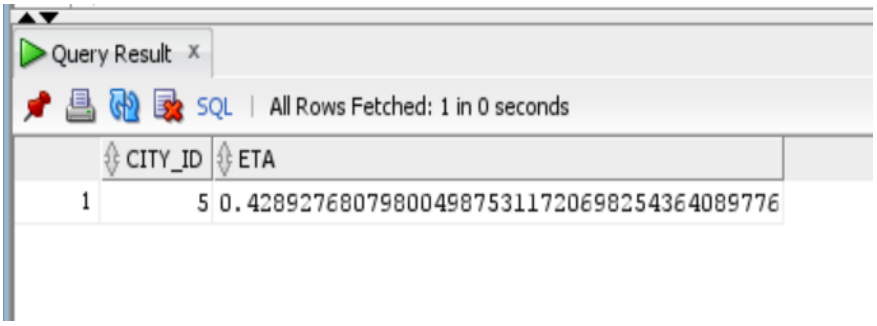
OUTPUT:



b. Which city_id has the highest error in ETA (where error in ETA = {(eta - ata)/ata}) for the given time period?

```
select * from(select city_id,(eta-ata)/ata as ETA
from fact_trip  order by ETA desc)where rownum=1;
```

OUTPUT:-

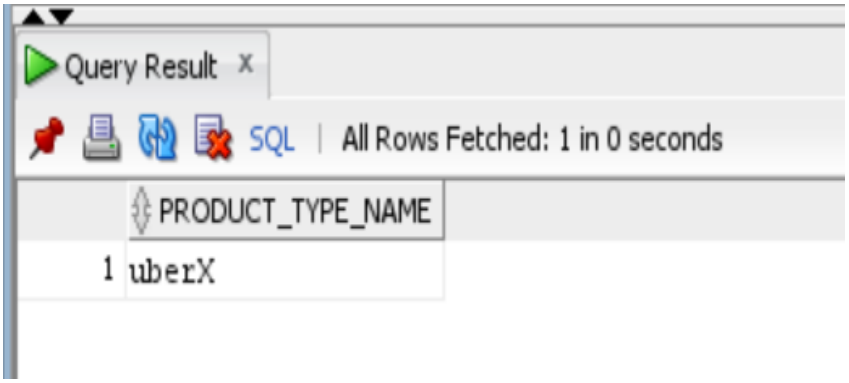


c. Which is the product type with highest total revenue in SanFrancisco?

```
select product_type_name from fact_trip where fare_final = (select
max(fare_final) from
```

```
(select fare_final from fact_trip where city_id=(select city_id from dim_city where
    city_name = 'SanFrancisco')))
```

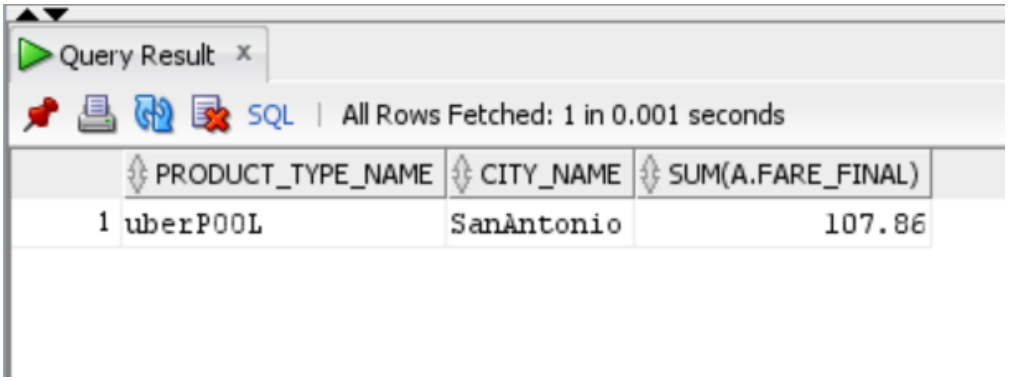
OUTPUT:-



d. Which are the products in each city where total revenue(fare_final) > $1000?

```
select a.product_type_name, b.city_name, sum(a.fare_final) from dim_city b join
    fact_trip a on a.city_id=b.city_id
group by a.product_type_name, b.city_name having sum(a.fare_final)>100;
```

OUTPUT:-



e. Get to 2nd highest country by Uber Revenue (fare_final) for 2nd week of June 2018 across product

```
select * from(select d.country, f.fare_final,
rownum as rank from dim_city d join fact_trip f on d.city_id=f.city_id
where to_char(datestr, 'W')=2 order by 2) where mod(rank, 2)=0;
```

f. Get WOW growth % for US region for June Month. WOW- Week over week .

```
select (((select sum(fare_final) from fact_trip where to_char(datestr, 'W')=1)  -
(select sum(fare_final) from fact_trip where to_char(datestr, 'W')=2))
  / (select sum(fare_final) from fact_trip where to_char(datestr, 'W')=1)) * 100 as
"Growth%"
from fact_trip group by datestr;
```

g. **Growth % = ((Current week fare final - previous week fare final) / previous week fare final) * 100**