Homework | module 2 > week 8 > day 19

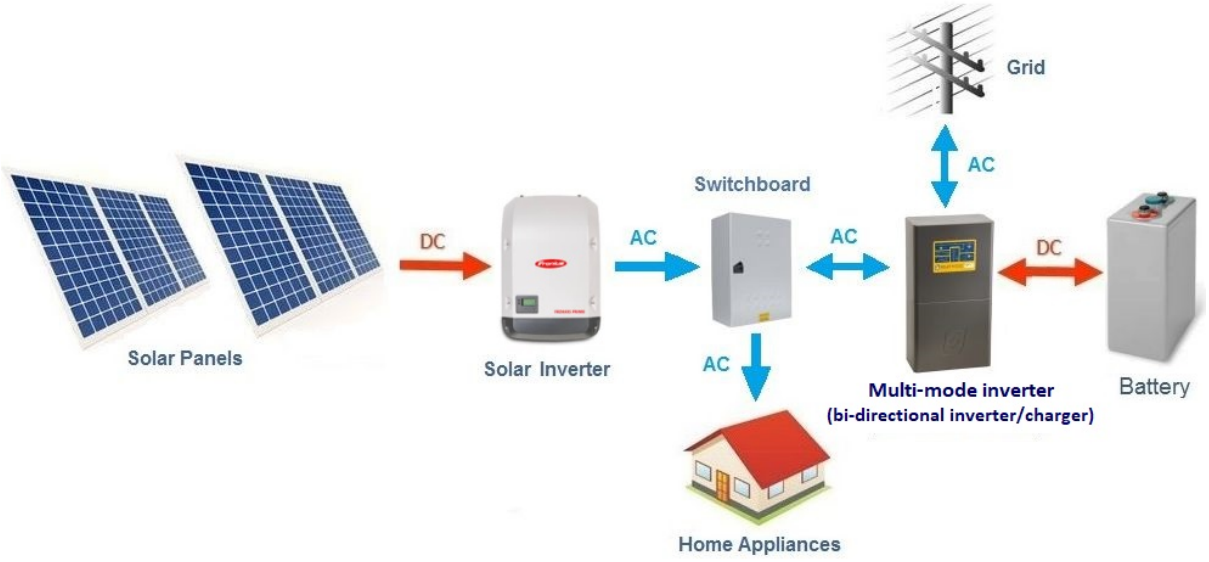
Topics covered: *SUBQUERIES*

Standard Exercise:

If you completed the last homework session, you should have a SolarPower dataset containing two tables:

- SolarPower.Generation_Data
- SolarPower.Weather_Sensor_Data

Roughly speaking, when the sunlight hits a solar panel, a flow of charge is created, resulting in the generation of DC Power (Direct Current). The DC Power is then transferred to an inverter, which converts it into AC Power (Alternating Current), which eventually reaches our homes.



Given this data, answer the following questions:

1. Write a query that shows the average AC and DC Power generated, grouped by each power plant.

```
SELECT PLANT_ID,
 (Select avg(AC_POWER)
 FROM `adept-bond-365418.SolarPower.Generation_Data`),
  (Select avg(DC_POWER)
 FROM `adept-bond-365418.SolarPower.Generation_Data`),
 FROM `adept-bond-365418.SolarPower.Generation_Data`
 group by PLANT_ID
```

2. *In the process of DC-to-AC conversion, no inverter can achieve 100% efficiency. This means that the output (AC) energy is not as high as the input (DC) energy. The efficiency of the inverter, calculated as AC Power / DC Power, generally ranges from 95% to 98%.*
   Starting from the last query, what is the overall average inverter efficiency for each Plant?

```
SELECT *,
       FROM (SELECT PLANT_ID,
        avg(AC_POWER)as avg_ac,
        avg(DC_POWER) as avg_dc,
       round( avg(AC_POWER) / avg(DC_POWER)*100, 2) as overall_avg_for_plant
       FROM `adept-bond-365418.SolarPower.Generation_Data`
       group by PLANT_ID)
```

3. According to this data, which Plant is the most efficient?
   Do you notice anything strange? If so, what could the reason be?

4. Let's now focus on plant_id = 4136001. Write a query that shows the average DA and AC Power as well as the average inverter efficiency for each hour of the day. *Hint: careful about the "division by zero" error*

```
    SELECT
        EXTRACT( hour from DATE_TIME) as hour,
        avg(AC_POWER)as avg_ac,
        avg(DC_POWER) as avg_dc,
        round(avg(AC_POWER)/ avg(DC_POWER)*100 ,2 ) as inverter_efficency
       FROM `adept-bond-365418.SolarPower.Generation_Data`
       Where PLANT_ID = 4136001 and AC_POWER <> 0 and DC_POWER <> 0
       group by hour
       order by hour
```

5. What can you say about the hourly distribution of power generated? Why are there zeros in the resulting table?
6. How many inverters (source_key) are there in the Generation_Data table? And in the Weather_Sensor_Data table?

```
SELECT *,
       FROM (SELECT distinct count (SOURCE_KEY) as num_of_inverter
             FROM `adept-bond-365418.SolarPower.Generation_Data`
```

```
        )


SELECT *,
       FROM (SELECT distinct count (SOURCE_KEY) as num_of_inverter
        FROM `adept-bond-365418.SolarPower.Weather_Data`
       )
```

7. Are there any source keys in the Weather_Sensor_Data table that are also present in the Generation_Data table? Can you think of a way of using a SUBQUERY (in the WHERE clause of a query) to check for this?

SELECT distinct source_key
FROM `SolarPower.Generation_Data`
WHERE source_key in (select distinct source_key FROM `SolarPower.Weather_Sensor_Data`);

8. Let's say that the anomaly in the data you observed at question 3 is due to a measurement error. Write a query that will modify only the DC Power data relative to plant_id 4135001 by moving the decimal point one step to the left (eg: divide by 10). *Note: you won't be able to execute the update statement due to BigQuery's Sandbox limitations, just write down the code that would make the appropriate changes.*

```
-- UPDATE `SolarPower.Generation_Data`
-- SET DC_POWER = DC_POWER/10
-- WHERE PLANT_ID = 4135001
```

Advanced Exercise (optional):

1. Since we can't update the existing table, let's re-create the Generation_Data table via a UNION statement and call it Generation_Data_Clean. Make sure you:
    a. Fix the DC Power problem in the Plant1 table
    b. Add a new string column in the final table called Plant_nr where you manually specify whether that data is relative to "plant_1" or "plant_2"

```
CREATE TABLE `SolarPower.Generation_Data_Clean` AS
SELECT "plant_1" AS PLANT_NR, DATE_TIME, PLANT_ID, SOURCE_KEY,
DC_POWER/10 as DC_POWER, AC_POWER, DAILY_YIELD, TOTAL_YIELD
FROM `SolarPower.Plant_1_Generation_Data`
UNION ALL
SELECT "plant_2" AS PLANT_NR, DATE_TIME, PLANT_ID, SOURCE_KEY, DC_POWER,
AC_POWER, DAILY_YIELD, TOTAL_YIELD
FROM `SolarPower.Plant_2_Generation_Data`;
```

2. The Weather_Sensor_Data table stores records of the average *ambient* (outdoor temp) and *module* (photovoltaic panel temp) temperatures as well as *irradiation* levels (the amount of the sun's power detected by a sensor). What are the average ambient temperature, module temperature and irradiation by hour of day?

```
SELECT extract(hour from DATE_TIME) as hour_of_day,
 avg(AMBIENT_TEMPERATURE) as avg_amb_temp,
 avg(MODULE_TEMPERATURE) as avg_mod_temp,
 avg(MODULE_TEMPERATURE) - avg(AMBIENT_TEMPERATURE) as avg_temp_diff,
 avg(IRRADIATION) as avg_irradiation
FROM `SolarPower.Weather_Sensor_Data`
GROUP BY hour_of_day
ORDER BY hour_of_day;
```

3. What can you say about the data you just generated?

4. Using a JOIN and SUBQUERIES, merge the output table from question 4 (using the new "Clean" table and without the filter on the power plant) to the

table produced in question 8. Add the PLANT_ID to both outputs so that you can use that in the JOIN as well. In the end you should have an output table that looks like this (I also used plant_nr instead of plant_id):

Row	PLANT_ID	hour_of_day	avg_dc_power	avg_ac_power	inverter_efficiency	PLANT_ID_1	hour_of_day_1	avg_amb_temp	avg_mod_temp	avg_irradiation
1	4135001	0	0.0	0.0	*null*	4135001	0	22.760292696655771	20.951019411050524	0.0
2	4135001	1	0.0	0.0	*null*	4135001	1	22.566702119186061	20.840318763894736	0.0
3	4135001	2	0.0	0.0	*null*	4135001	2	22.460107824962794	20.766689789546131	0.0
4	4135001	3	0.0	0.0	*null*	4135001	3	22.319239462239569	20.624820254538694	0.0
5	4135001	4	0.0	0.0	*null*	4135001	4	22.171524788467256	20.469068666592264	0.0
6	4135001	5	0.0	0.0	*null*	4135001	5	22.074731526946326	20.369940101549506	0.000133319349
7	4135001	6	578.11362227459017	56.135778039931168	9.71	4135001	6	22.206515972694877	21.289874470527131	0.041506145944
8	4135001	7	2551.3867204522885	250.23916308646281	9.81	4135001	7	23.34496072683984	26.914710028343574	0.193634098682
9	4135001	8	5088.6282181113093	498.91099976955246	9.8	4135001	8	24.917593987710088	35.499309478186248	0.372674695432

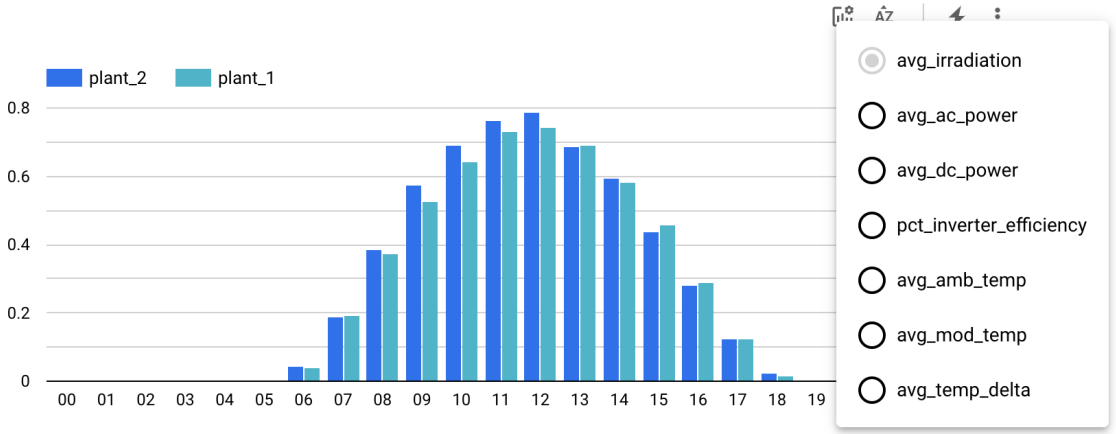
```
SELECT *
FROM
 (SELECT PLANT_ID, extract(hour from DATE_TIME) as hour_of_day,
   avg(DC_POWER) as avg_dc_power,
   avg(AC_POWER) as avg_ac_power,
   round(avg(case when AC_POWER <> 0 then AC_POWER end)/avg(case when
DC_POWER <> 0 then DC_POWER end)*100, 2) as inverter_efficiency
 FROM `SolarPower.Generation_Data`
 GROUP BY PLANT_ID,hour_of_day) a
LEFT JOIN
 (SELECT PLANT_ID, extract(hour from DATE_TIME) as hour_of_day,
   avg(AMBIENT_TEMPERATURE) as avg_amb_temp,
   avg(MODULE_TEMPERATURE) as avg_mod_temp,
   avg(IRRADIATION) as avg_irradiation
 FROM `SolarPower.Weather_Sensor_Data`
 GROUP BY PLANT_ID, hour_of_day) b
 on a.PLANT_ID = b.PLANT_ID
 and a.hour_of_day = b.hour_of_day
ORDER BY a.PLANT_ID, a.hour_of_day;
```

5. Using the previous query as the base, create a new table and call it Hourly_Generation_Weather_Plant.

```
CREATE TABLE SolarPower.Hourly_Generation_Weather_Plant AS
SELECT a.plant_nr, a.hour_of_day, a.avg_dc_power, a.avg_ac_power,
a.pct_inverter_efficiency, b.avg_amb_temp, b.avg_mod_temp, b.avg_irradiation
FROM
 (SELECT plant_nr, PLANT_ID, extract(hour from DATE_TIME) as hour_of_day,
   avg(DC_POWER) as avg_dc_power,
   avg(AC_POWER) as avg_ac_power,
   round(avg(case when AC_POWER <> 0 then AC_POWER end)/avg(case when
DC_POWER <> 0 then DC_POWER end)*100, 2) as pct_inverter_efficiency
 FROM `SolarPower.Generation_Data_Clean`
 GROUP BY plant_nr, PLANT_ID,hour_of_day) a
LEFT JOIN
```

```
 (SELECT PLANT_ID, extract(hour from DATE_TIME) as hour_of_day,
   avg(AMBIENT_TEMPERATURE) as avg_amb_temp,
   avg(MODULE_TEMPERATURE) as avg_mod_temp,
   avg(IRRADIATION) as avg_irradiation
 FROM `SolarPower.Weather_Sensor_Data`
 GROUP BY PLANT_ID, hour_of_day) b
 on a.PLANT_ID = b.PLANT_ID
 and a.hour_of_day = b.hour_of_day
ORDER BY a.PLANT_ID, a.hour_of_day
```

6. Go to Google Data Studio, connect to the last table you created (Hourly_Generation_Weather_Plant) and explore the data in it.
7. Create a chart, like the one below, that shows the relationship between the hour of the day and a variety of metrics (of your choice) and highlights (with colour) the differences between plant_1 and plant2:



8. Create a chart, like the one below, that shows the relationship between irradiation and the inverter's efficiency among the two power plants. What is

the chart telling us? What can you say about this relationship?