SQL-CODE

--Mean temp

```
DROP VIEW IF EXISTS HOBO_mean cascade;
CREATE VIEW HOBO_mean AS
SELECT
avg(data.value) as avg,
metadata.device_id as HOBO_id
FROM data
JOIN metadata ON data.meta_id= metadata.id
WHERE metadata.term_id= 11
group by metadata.device_id
order by metadata.device_id ASC;
SELECT * FROM HOBO_mean LIMIT 15;
```

--Mean daytime temp

```
DROP VIEW IF EXISTS HOBO_day;
CREATE VIEW HOBO_day AS
SELECT
avg(data.value) as avg_t,
metadata.device_id as HOBO_id
FROM data
JOIN metadata ON data.meta_id= metadata.id
WHERE metadata.term_id= 11
AND EXTRACT(HOUR FROM data.tstamp) >= 6
AND EXTRACT(HOUR FROM data.tstamp) < 18
group by metadata.device_id
order by metadata.device_id ASC;
SELECT * FROM HOBO_day LIMIT 15;
```

--Mean nighttime temp

```
DROP VIEW IF EXISTS HOBO_night;
CREATE VIEW HOBO_night AS
SELECT
avg(data.value) as avg_n,
metadata.device_id as HOBO_id
FROM data
JOIN metadata ON data.meta_id= metadata.id
WHERE metadata.term_id= 11
AND (EXTRACT(HOUR FROM data.tstamp) < 6
OR EXTRACT(HOUR FROM data.tstamp) >= 18)
group by metadata.device_id
order by metadata.device_id ASC;
SELECT * FROM HOBO_night LIMIT 15;
```

--Temperature indices in one table

```
DROP TABLE IF EXISTS HOBO_indices;
CREATE TABLE HOBO_indices AS
SELECT d.hobo_id,
m.avg as T_avg,
d.avg_t as T_d,
n.avg_n as T_n,
(d.avg_t-n.avg_n) as T_nd
FROM HOBO_day d
JOIN HOBO_night n ON d.hobo_id= n.hobo_id
JOIN HOBO_mean m ON d.hobo_id= m.hobo_id;
SELECT * FROM HOBO_indices LIMIT 15;
```

--Selecting the closest neighbor of 19/20 and 18/19

```
DROP VIEW IF EXISTS meta21 Cascade;
CREATE VIEW meta21 AS
WITH meta21 AS (
    SELECT *,
    (SELECT id FROM metadata ly WHERE term_id=9 ORDER BY st_distance(m.locatio
n, ly.location) ASC LIMIT 1) as close_meta20_id,
    (SELECT id FROM metadata ly WHERE term_id=7 ORDER BY st_distance(m.locatio
n, ly.location) ASC LIMIT 1) as close_meta19_id
    FROM metadata m
    WHERE term_id=11 AND sensor_id=1
    )
SELECT *
FROM meta21;
SELECT * FROM meta21;
```

--Table with normalized data and measurement index to make it possible to compare the different years.

```
DROP VIEW IF EXISTS data_norm;
CREATE VIEW data_norm AS
SELECT
    row_number() OVER (PARTITION BY meta_id, variable_id ORDER BY tstamp ASC)
as measurement_index,
    *,
    value - avg(value) OVER (PARTITION BY meta_id, variable_id) AS norm,
    avg(value) OVER (PARTITION BY meta_id, variable_id) AS group_avg
FROM data;
SELECT * FROM data_norm;
```

--Create the correlation value by joining the different years by the measurement index. Twice for both years.

```
DROP VIEW IF EXISTS indices;
CREATE VIEW indices AS
    SELECT
        meta21.id,
        avg(d.value) AS "mean",
        corr(d.norm, d20.norm) AS "Tcorr1Y19"
    FROM data_norm d
    JOIN meta21 on meta21.id = d.meta_id
    JOIN metadata m20 on meta21.close_meta19_id=m20.id
    JOIN data_norm d20 on m20.id=d20.meta_id AND d.measurement_index=d20.measu
rement_index
    GROUP BY meta21.id;
SELECT * FROM indices;

DROP VIEW IF EXISTS indices20;
CREATE VIEW indices20 AS
    SELECT
        meta21.id,
        avg(d.value) AS "mean",
        corr(d.norm, d20.norm) AS "Tcorr1Y20"
    FROM data_norm d
    JOIN meta21 on meta21.id = d.meta_id
    JOIN metadata m20 on meta21.close_meta20_id=m20.id
    JOIN data_norm d20 on m20.id=d20.meta_id AND d.measurement_index=d20.measu
rement_index
    GROUP BY meta21.id;
 SELECT * FROM indices20;
```

--Table with both correlation columns

```
DROP VIEW IF EXISTS indices_kira;
CREATE VIEW indices_kira AS
SELECT i.mean, i."Tcorr1Y19", i20."Tcorr1Y20", m.device_id, m.location  FROM i
ndices i
JOIN metadata m on i.id=m.id
JOIN indices20 i20 on i.id=i20.id;
SELECT * FROM indices_kira;
```

--Table with correlation columns and all indices

```
DROP VIEW IF EXISTS HOBO_corr;
CREATE VIEW HOBO_corr AS
SELECT * FROM HOBO_indices
JOIN indices_kira k on k.device_id=HOBO_indices.hobo_id;
SELECT * FROM HOBO_corr
```