0.1 Question 1: Unboxing the Data

0.1.1 Question 1a

As mentioned above, we are working with just one month of data. In the full database (which we don't have access to), tables like the `data` table have billions of rows. What do you notice about the design of the database schema above that helps support the large amount of data and minimize redundancy? Keep your response to at most three sentences.

Hint: There is no need to examine any data here. What is a technique learned in lecture 16? Define that technique.

The design of the database schema is optimized for handling large volumes of data through "normalization," a technique discussed in Lecture 16. Normalization involves organizing the columns (attributes) and tables (relations) of a database to minimize data redundancy and avoid undesirable characteristics like Insertion, Update, and Deletion Anomalies. This makes the database more efficient and scalable, particularly when dealing with tables containing billions of rows.

0.1.2

0.1.3 Question 1d

Do you see any issues with the schema given? In particular, please address the two questions below: - Can you uniquely determine the building given the sensor data? Why? (**Hint:** given a row in the `data` table, can you determine a **uniquely** associated row in `real_estate_metadata` table? Your answer should draw insights from 1b.) - Could `buildings_site_mapping.building` be a valid foreign key pointing to `real_estate_metadata.building_name`? (**Hint:** think about the definition / constraints of a foreign key.)

Please keep your response to **at most three sentences.**

No, you cannot uniquely determine the building given the sensor data because the many-to-many relationship identified in question 1b implies that multiple entries in the real_estate_metadata table can correspond to a single building in the buildings_site_mapping table, preventing a unique association.

No, buildings_site_mapping.building cannot be a valid foreign key pointing to real_estate_metadata.building_name because, as seen from the exercises, one building_name can correspond to multiple buildings, violating the foreign key constraint that requires each value in the referenced column (here, building_name) to be unique.

0.2 Question 3: Entity Resolution

0.2.1 Question 3a

There is a lot of mess in this dataset related to entity names. As a start, have a look at all of the distinct values in the `units` field of the `metadata` table. What do you notice about these values? Are there any duplicates? **Limit your response to one sentence.**

The dataset contains numerous duplicates in the units field due to inconsistent capitalization and abbreviation, such as “Lbs” versus “lbs,” “KWH” versus “kWh,” and varied representations like “V” and “Volts,” alongside ambiguous terms like “subinterval” and “test.”

0.2.2 Question 3d

Moving on, have a look at the `real_estate_metadata` table—starting with the distinct values in the `location` field! What do you notice about these values? Keep your response to at most two sentences.

Typographical errors in the location field result in discrepancies and duplicates, preventing identical locations from matching correctly.