# Database Systems, CSCI 4380-01
# Homework # 4 Answers

**Question 1.** We are creating a database to help Greta Thunberg in her efforts to communicate to everyone the urgency of the climate crisis. Note that for this database, the location always refers to a latitude and longitude value.

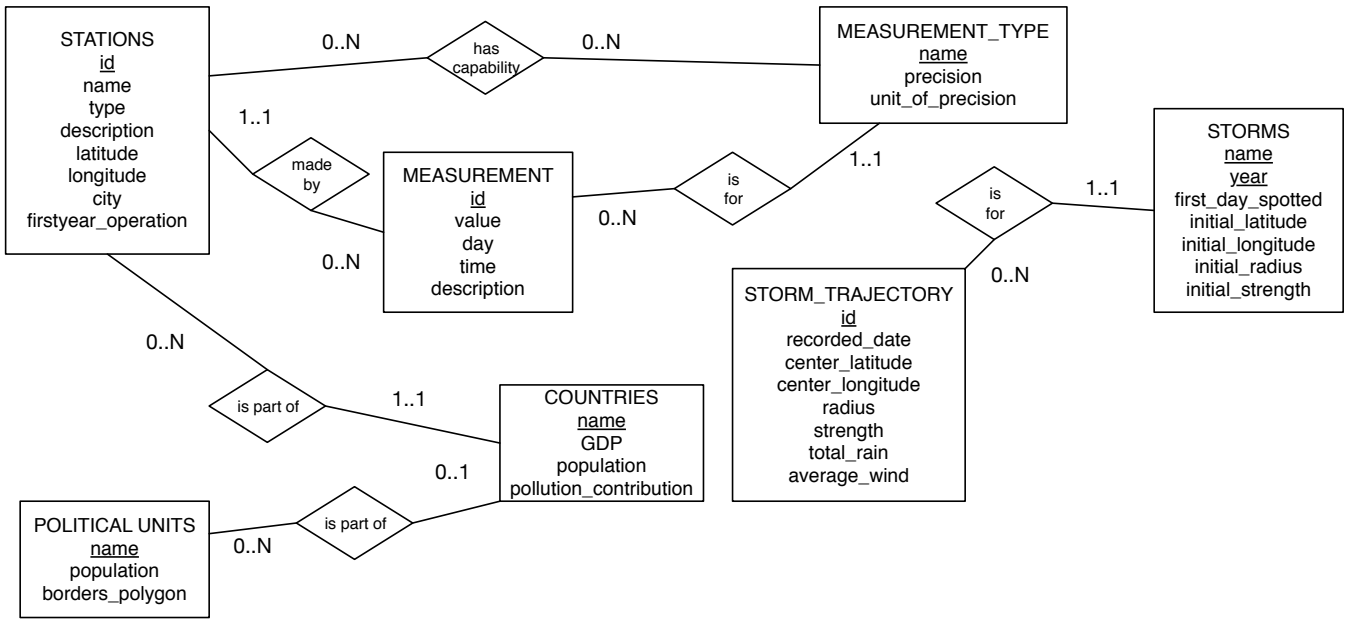
We will store a number of stations. Each station will have an id, name, sensor type, description, location, country, city, first year of operation. Stations have capabilities, types of things they can measure. Each capability type for a station has a name such as temperature, humidity, rain fall, snow fall, air quality, water quality, wind, depth (for underwater stations) and a precision (float) and a unit of precision (string).

We will store actual measurements from these stations. Each measurement has an id, value (float), type of measurement, day and time of measurement and any description. The measurement type is similar to the capability of the station taking values like temperature, humidity, etc.

Next we will store the major storms in the database. For each storm, we will store the name, day it was first spotted, the initial location of the eye of storm (center), its radius and strength (integer). We will then store the location and radius of the storm for each day it was active, its strength, total rain amount and average wind for that day (note that climate crisis is not just causing storms to be stronger, but also may cause them to linger longer and cause more damage).

We will also store political units in the database to understand which communities are most effected from the climate change. For each political unit, we will store name, population and a polygon which is represented as a string for now. A political unit may be a part of a country. For each country, we will store its name, GDP (gross domestic product), population and contribution to pollution (percentage value).

**Answer.** You can choose to store the initial data on a storm in the storm_trajectory entity as well for simplicity. Interestingly, many entities in this database seem unlinked explicitly, but implicitly they are linked by and are comparable with respect to location.




**Question 2.** You are given the figure below representing an E-R diagram. Convert this model to relational data model. List your relations and underline your keys.

**Answer.**

Users(email, name, street, state, city, zip, credit_card)

Messages(id, message_text, message_date, sentby_useremail, receivedby_useremail, prevmessage_id)

Trips(id, reservation_date, is_confirmed?, confirm_date, fromdate, todate, price, is_past, takeruser_email, isforhouse_id)

Houses(id, label, description, street, state, city, zip, owneruser_email)

Pictures(seqid, image, house_id)

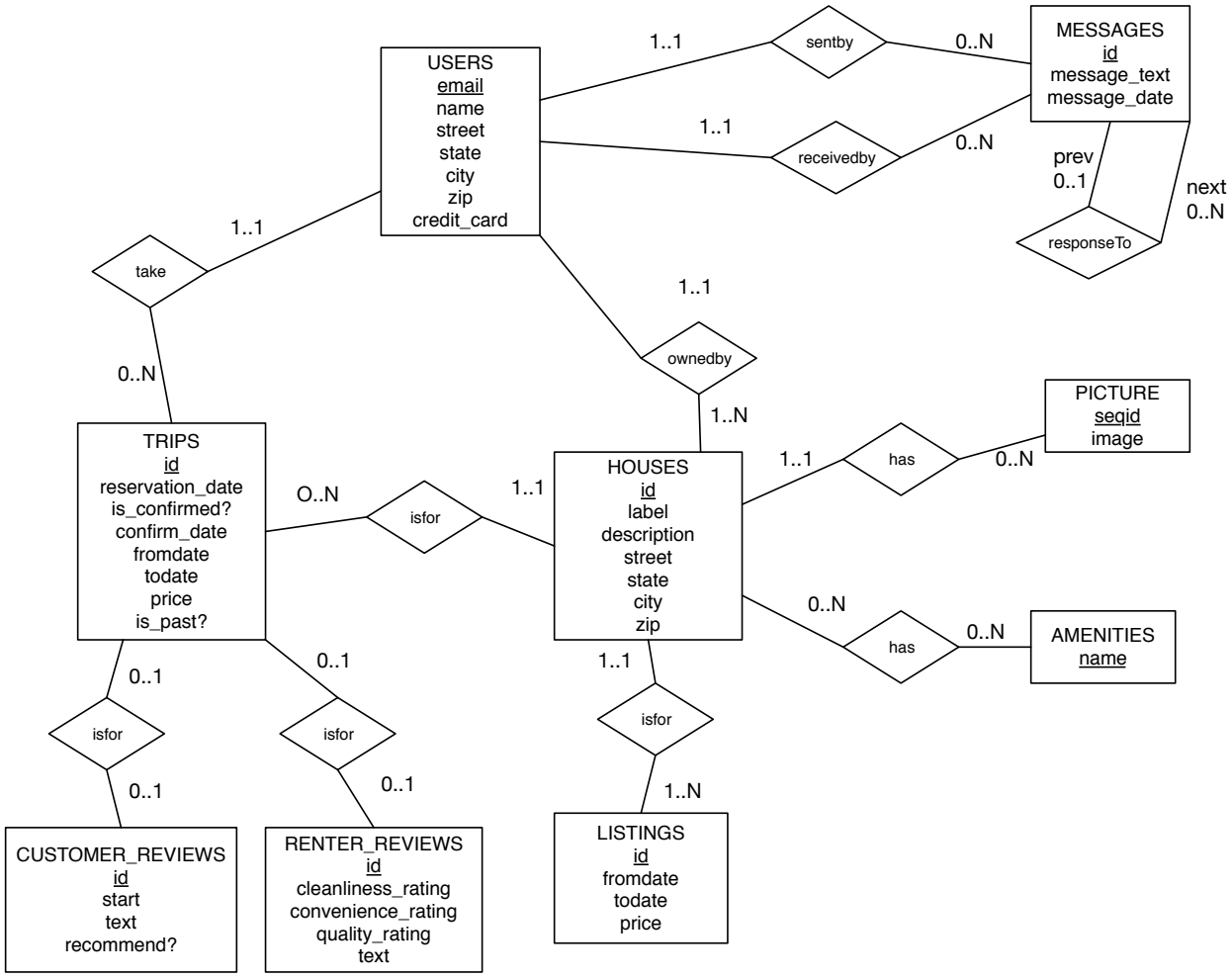
Amenities(name)

HousesAmenities(house_id, amenity_name)

Listings(id, fromdate, todate, price, house_id)

CustomerReviews(id, start, text, recommend?, trip_id)

RenterReviews(id, cleanliness_rating, convenience_rating, quality_rating, text, trip_id)


USERS
email
name
street
state
city
zip
credit_card
sentby
1..1
0..N
MESSAGES
id
message_text
message_date
receivedby
1..1
0..N
prev
0..1
next
0..N
responseTo
take
1..1
0..N
ownedby
1..1
1..N
TRIPS
id
reservation_date
is_confirmed?
confirm_date
fromdate
todate
price
is_past?
O..N
isfor
1..1
HOUSES
id
label
description
street
state
city
zip
has
1..1
0..N
PICTURE
seqid
image
has
0..N
0..N
AMENITIES
name
0..1
isfor
0..1
0..1
isfor
0..1
1..1
isfor
1..N
CUSTOMER_REVIEWS
id
start
text
recommend?
RENTER_REVIEWS
id
cleanliness_rating
convenience_rating
quality_rating
text
LISTINGS
id
fromdate
todate
price