Objective

The objective of this team project is for you to apply what you have learned in class to design and develop a database application using PostgreSQL. The project is to be done in teams of four students. The project consists of the following four tasks:

(P01) Design an ER data model for the application. Your ER model should capture as many of the application's requirements as possible.

(P02) Translate your ER data model into a relational database schema. Your relational schema should capture as many of the application's constraints as possible.

(P03) Implement an SQL or PL/pgSQL functions/procedures for each of the functionalities listed in Application Functionalities.

(P03) Implement triggers as required by the specification.

Current Status

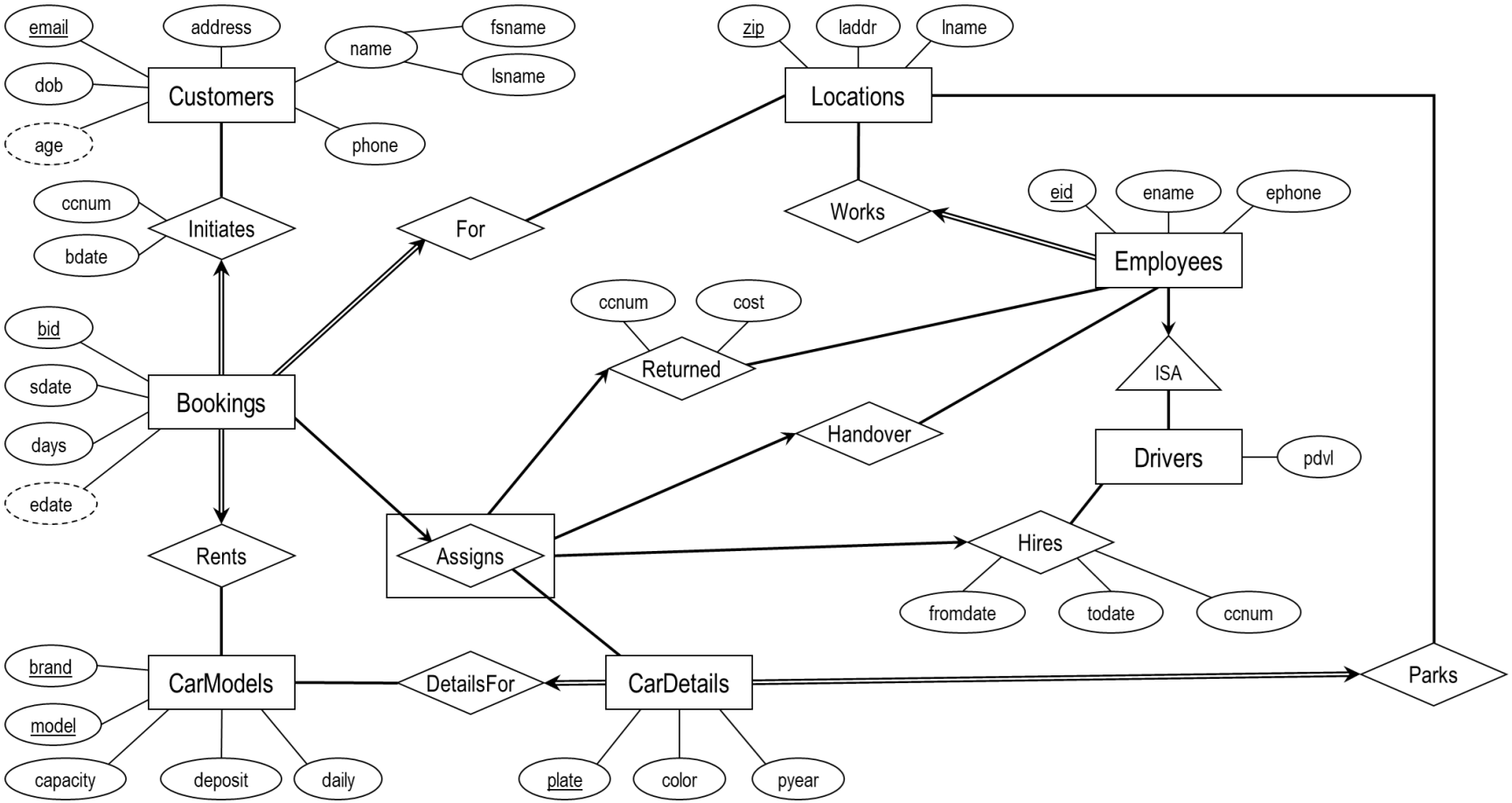
You are given a potential ER diagram on the next page. It is slightly simplified and should capture most of the important constraints necessary for the daily working of Barg Car.

Additionally, you are also given the list of constraints that the company wish to enforce. Note that potentially, not all of these can be enforced, but you should try as best as you can. This list is available in the template file P02.docx.

Your task is to look at the list of constraints in P02.docx closely and try to enforce as many of those as possible when you translate the ER diagram into schema *without using triggers*. For each constraint that is enforced, indicate with a ✓ and for each constraint that is not enforced, indicate with a ✗.

Please also make sure that the key attributes of each entity sets can uniquely identify the other attributes in the entity sets. Additionally, the key attributes of an entity set E_1 should not uniquely identify the key attributes of another entity set E_2. Of course there is an exception to this depending on the constraints.

If there is a cardinality constraint on E_1 with respect to a relationship set R and the entity set E_2 also participates in the relationship set R, then the key attributes of E_1 may uniquely identify the attributes in E_2. That is really just a fancy way of saying what we have already discussed in the lecture (https://thisisadi.yoga/cs2102/slides/L03.html#49).


email
address
name
fsname
lsname
dob
Customers
age
phone
ccnum
Initiates
bdate
For
zip
laddr
lname
Locations
Works
eid
ename
ephone
Employees
ccnum
cost
bid
Returned
ISA
sdate
Bookings
Handover
Drivers
pdvl
days
edate
Rents
Assigns
Hires
fromdate
todate
ccnum
brand
CarModels
DetailsFor
CarDetails
Parks
model
capacity
deposit
daily
plate
color
pyear

Deliverables

Each team is to upload a **zip** file named "`teamNNN.zip`" where "`NNN`" is the three digit team number according to your project group number. You should add leading zeroes to your team number (*e.g.*, `team005.zip`). Submit your **zip** file on Canvas assignment named "Project 02" (https://canvas.nus.edu.sg/courses/52825/assignments/93147). Only the **latest** submission that is submitted **before the deadline** will be marked.

You zip file should contain the following files:

- P02.pdf: containing the constraints, properly marked if they are enforced, as well as individual contribution to the project.
- P02.sql: containing SQL DDL (*e.g.*, `CREATE TABLE`).

As a general guideline, your schema should:

- not capture constraints not specified in the application's constraints.
  - Any constraints that can be captured but did not may be penalized.
- capture as many of the application's constraints as possible as per the specification.
  - Any constraint specified in the specification but is not captured or incorrectly added may be penalized.
- use the most reasonable data type without worrying about space requirement.
- add reasonable real-world constraints on real-world data.
  - For simplicity, you do not have to check for the validity of the email address.

Deadlines

The deadline for the submission is **12:00** of **Saturday** of Week 08 (*i.e.*, **16 March 2024**). Only submissions through Canvas will be accepted. Submissions through other means (*e.g.*, emails) will not be accepted.

Late Submissions

2 marks (*out of 6*) will be deducted for submissions up to **1 day late**. Submissions late for more than 1 day will receive **0 mark** and will not be graded.

Templates

Report

You are given the file P02.docx containing a template for your report with the list of constraints that have been identified by the company. The constraints may be slightly different from the initial requirements, but that is a reality of working in a project that you have to accept. Customers will change their requirements.

You should clearly indicate which constraints are enforced and which are not enforced by the schema.

Grading

The following grading scheme will be used. The details of the grading scheme is hidden.

Validity of Schema (0 Mark)

- Schema should run on PostgreSQL 16.
- Any schema that does not run for any reason will receive an immediate 0 mark for the entire submission.
- Make sure your schema can run on PostgreSQL 16 without any additional libraries.

Data Requirements (1 Marks)

- The schema should not have missing data.
- The schema should use reasonable data type.

Constraint Requirements (3 Marks)

- The schema should capture as many of the application's requirements as possible.
- The schema should not add constraints not specified in the application's requirements whenever possible.
- The schema should capture reasonable real-world constraints (*e.g.*, using `CHECK` or otherwise).

Analysis (2 Marks)

- The report should contain a thorough assessment of the constraints requirements.
- The report should correctly mention the constraints not captured.