CS 327E Lecture 10

Shirley Cohen

February 29, 2016

Agenda

- Announcements
- Readings for today
- Reading Quiz
- Concept Questions
- Homework for next time

Announcements

- Midterm exams will be returned at the end of class
- Midterm solutions will be available through Canvas
- Come to office hours if you want to review your exam

Homework for Today

- Chapter 6 from the Beginning Database Design book
- Exercises at the end of Chapter 6

Quiz Question 1

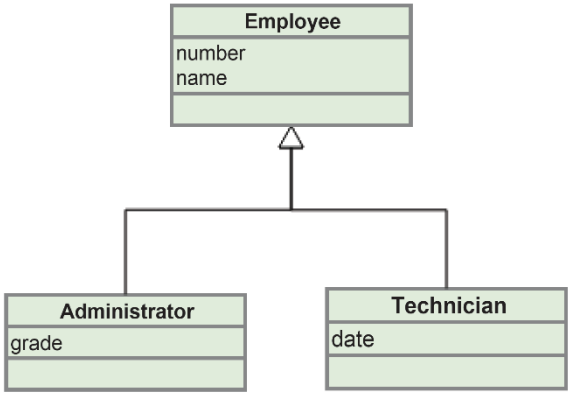
What is the **Open Closed Principle**?

A. Classes should be created for all instances of a parent class.
B. Classes should be associated solely by inheritance.
C. Classes should be open for extension and closed for modification.
D. Classes should be open for modification and closed for extension.

Quiz Question 2

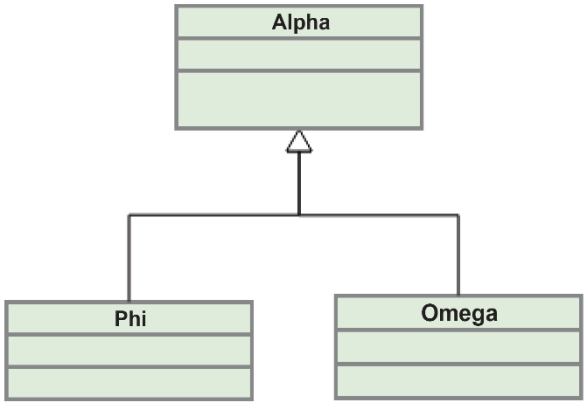
An Administrator is:

A. sometimes a Technician.
B. sometimes an Employee.
C. always an Employee.
D. never an Employee.



Quiz Question 3

John Doe is a Phi. According to the text, which should occur?



A. John Doe should be made an object of Phi.
B. John Doe should be made a subclass of Phi.
C. John Doe should be made into the parent class of Alpha.
D. All of the above

Quiz Question 4

What is a proper use of inheritance?

A. The midterm is an exam and thus is also a grade.
B. A rectangle has four sides and thus is also a square.
C. My phone has a screen and thus is also a laptop.
D. None of the above.

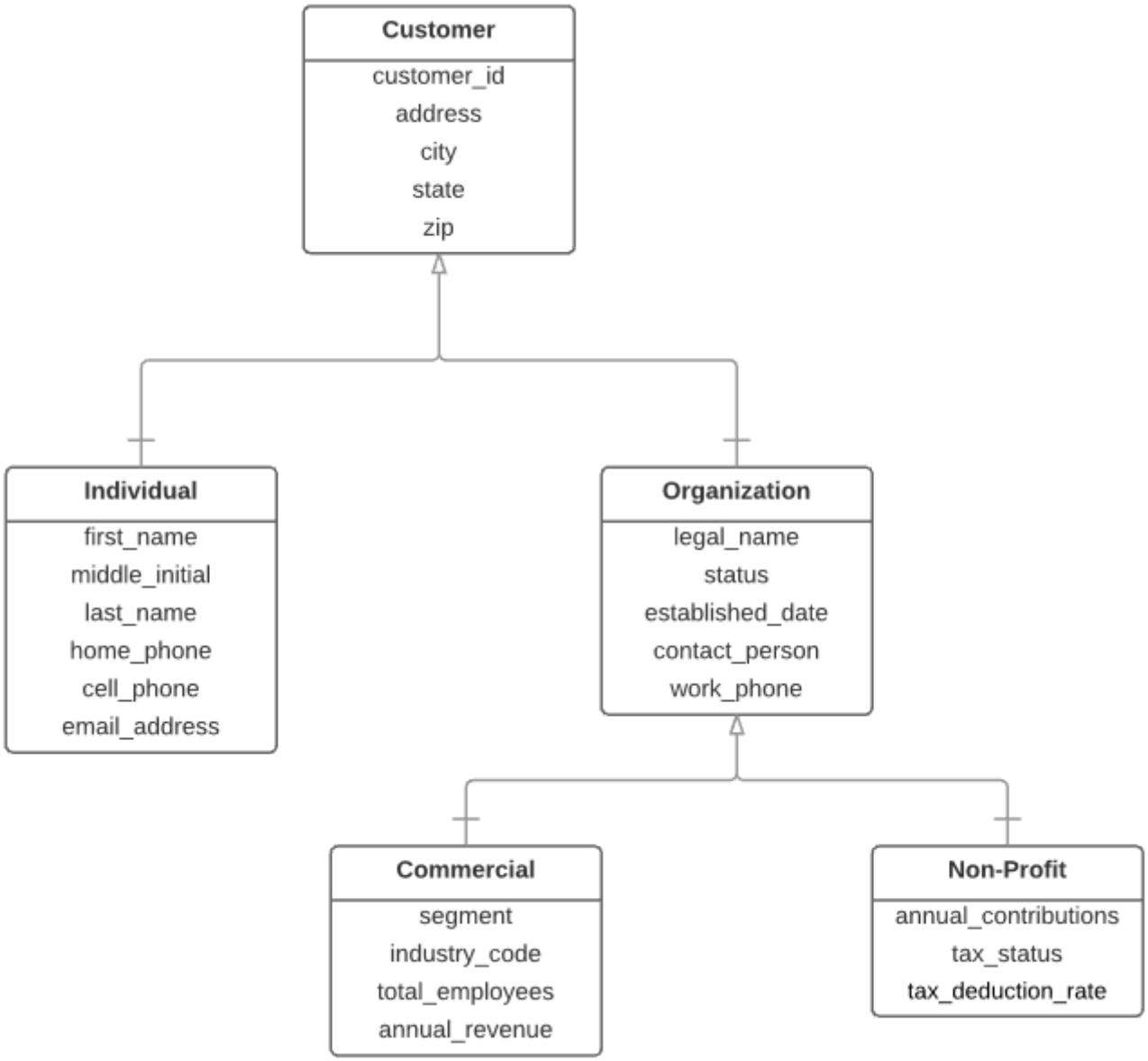
Quiz Question 5

What is a proper use of composition?

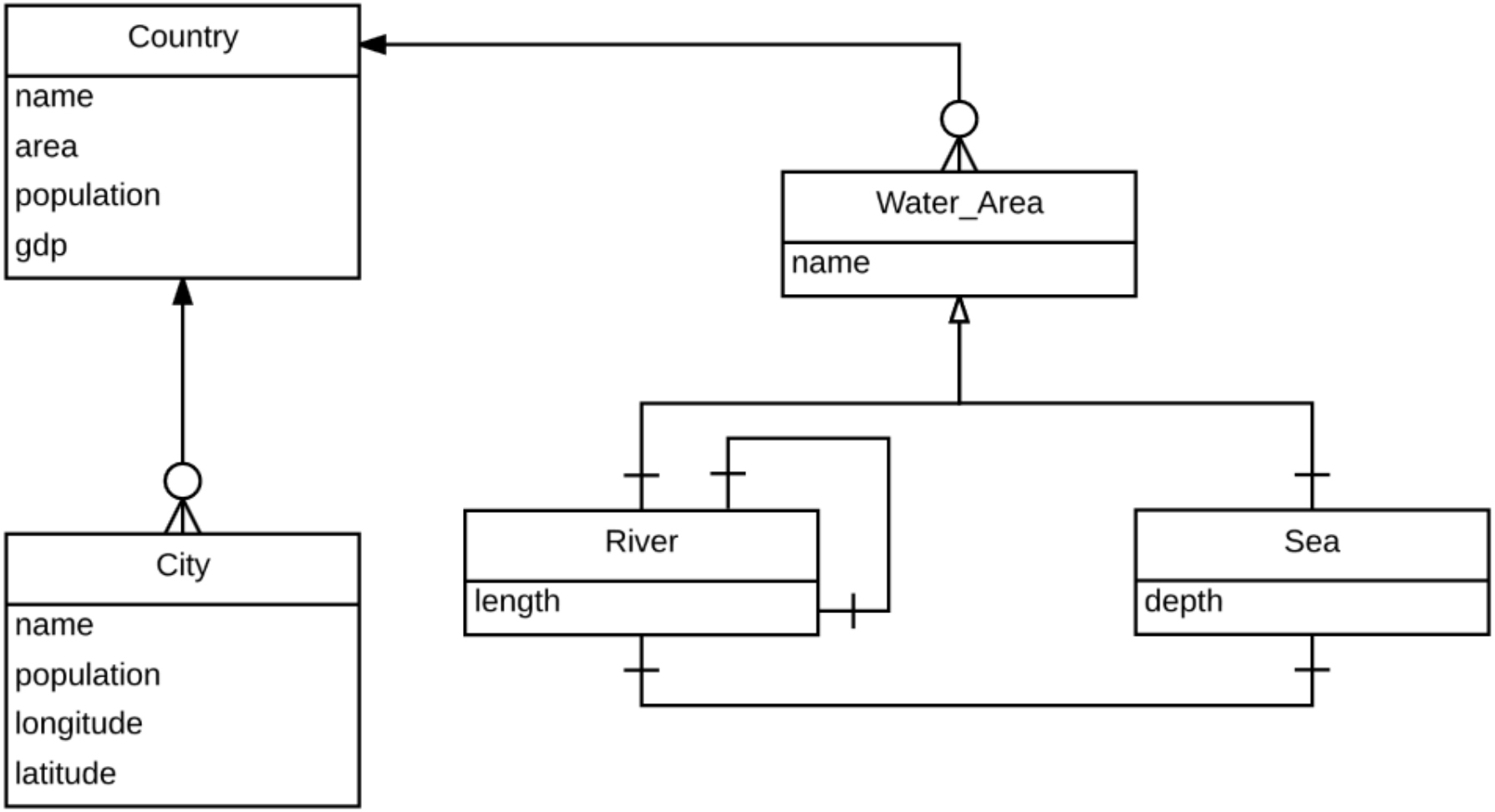
A. A textbook is part of a course.
B. A program is part of a product.
C. A line-item is part of an invoice.
D. All of the above.

Hierarchies in Data Modeling

Customer Diagram



Geography Diagram



Concept Question 1: Car Insurance

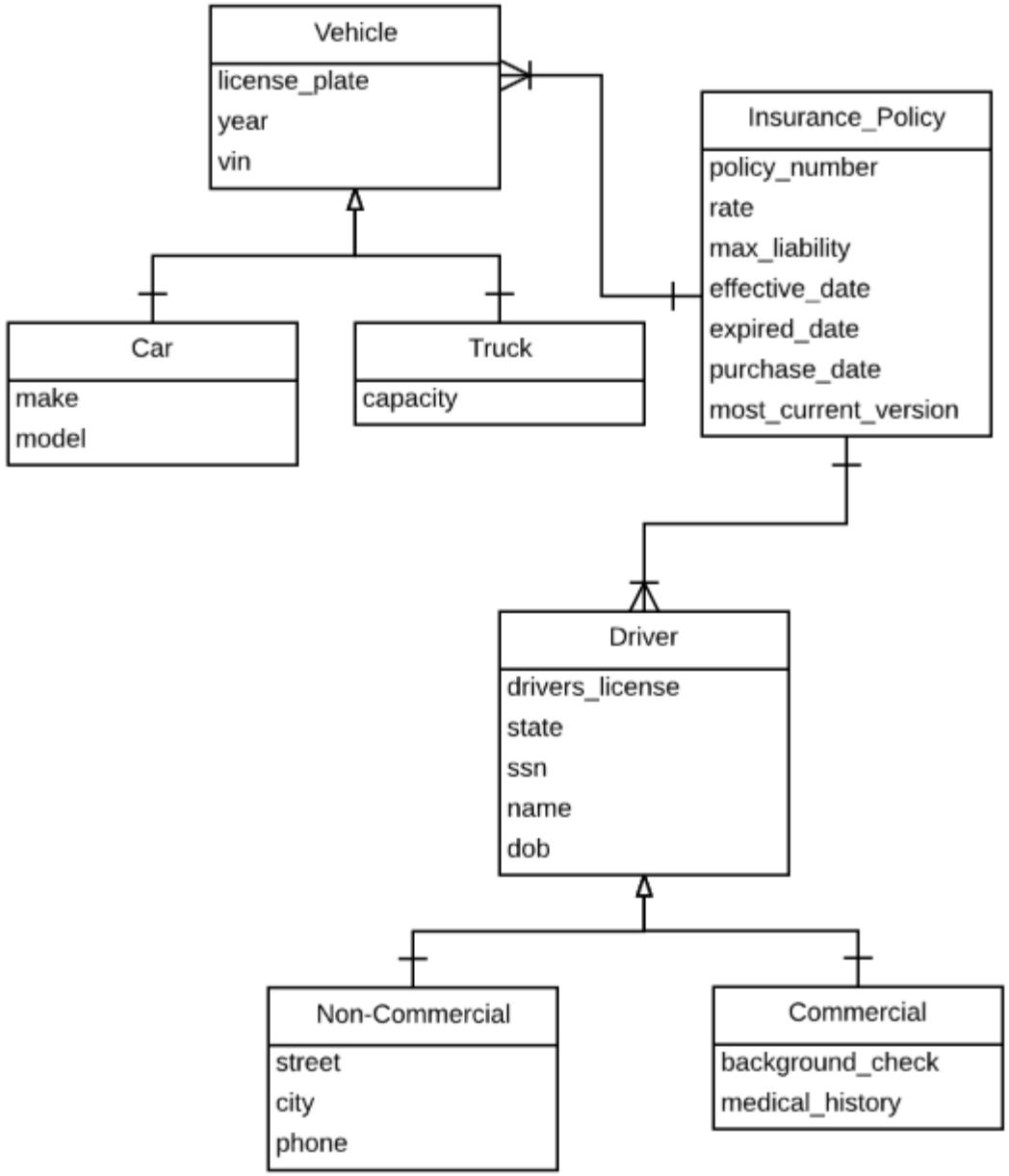
Consider a company that sells insurance coverage for vehicles and their owners. A *vehicle* is either a *car* or *truck*. The owner of a vehicle is a *commercial* driver, regular *non-commercial* driver or both. A truck must be insured to a commercial driver and a car must be insured by a non-commercial driver.

What is the relationship between a *Vehicle* and a *Car* in this example?

A. Inherited relationship

B. Composition relationship

Car Insurance Diagram



Concept Question 2: Discussion Forum

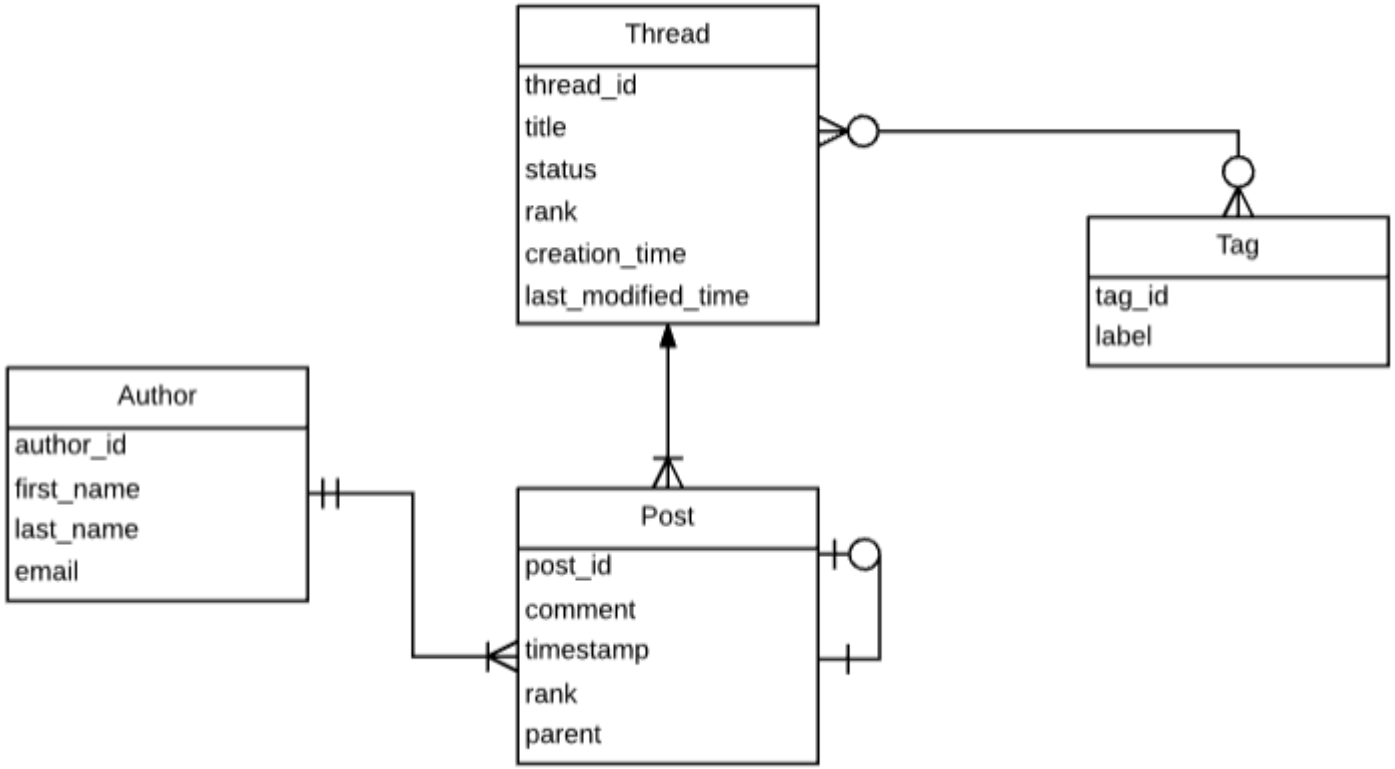
Consider an online discussion forum. It has multiple conversations or *threads*, each of which is composed of at least one *post*. A post must belong to a thread and is deleted if the thread is deleted. A post either has one parent when it is a response to another post or has no parent when it is the first post in a thread. A thread can be tagged with one or more *labels* that describe the subject matter of the conversation (e.g. “movies”, “music”, “jobs”, etc.).

Would you use composition to represent the relationship between a post and its parent post?

A. Yes

B. No

Discussion Forum Diagram



Data Modeling Exercise: Part 2

Recall the high-level objective of a **Personal Time Assistant** which is to help us better manage our precious time. In this exercise we want to introduce class hierarchies and specialized subclasses and incorporate them into the overall schema design. The idea is to model a class hierarchy only when it can help the system decide on how to schedule certain time commitments.

To help you get started, here are a few possible class hierarchies to consider :

1) An *Event* class hierarchy with specialized subclasses such as Meeting, Conference, Job Interview, Appointment, etc.

2) A *Task (or To-Do)* class hierarchy with specialized subclasses such as Homework, Walk Dog, Do Laundry, Answer Emails, etc.

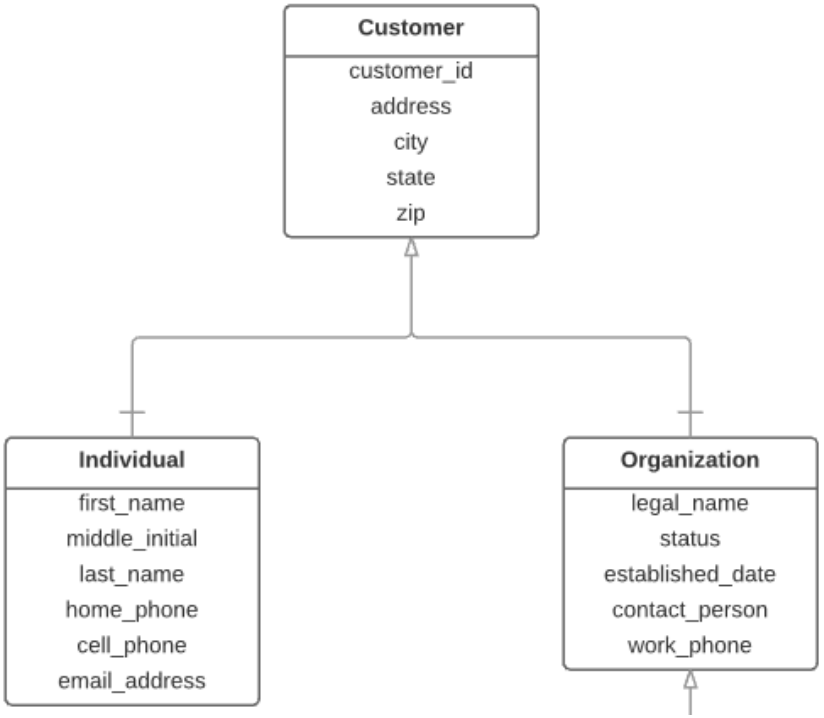
3) A *Habit* class hierarchy with specialized subclasses such as Blogging, Working Out, Drinking Water, etc.

4) A *Project* class hierarchy with specialized subclasses such as Writing Report, Building Database, Applying to Grad School, etc.

Note: These are only suggestions and you don't need to use them. Please come up with class hierarchies that make sense to your group.

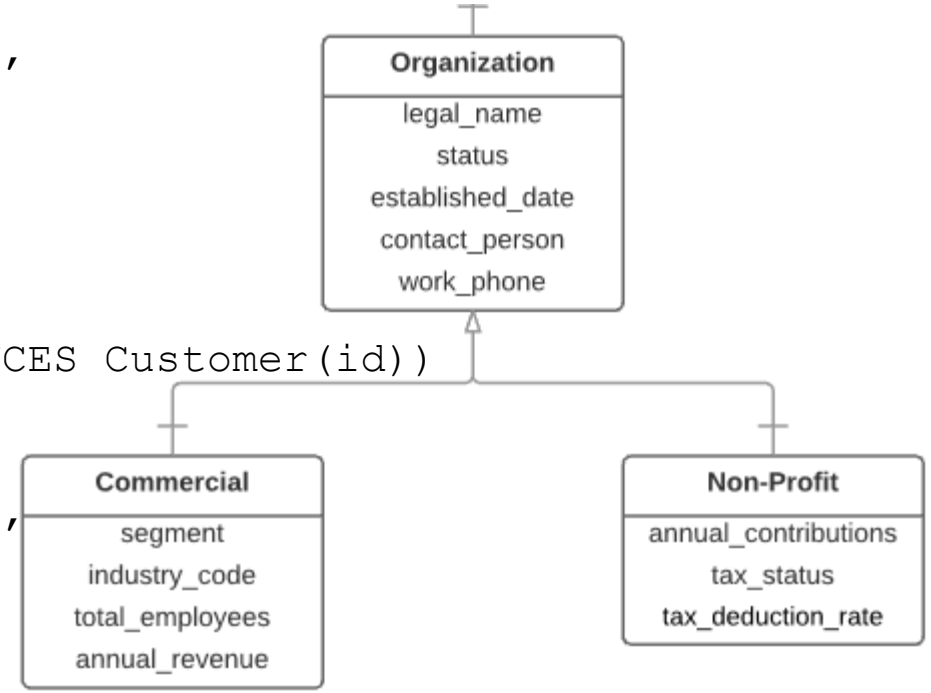
Customer Subclasses to Relations

```
CREATE TABLE Customer (
  customer_id INT PRIMARY KEY,
  address VARCHAR(50) NOT NULL,
  city VARCHAR(30) NOT NULL,
  state CHAR(2) NOT NULL,
  zip CHAR(5) NOT NULL)

CREATE TABLE Individual (
   customer_id NUMBER(8) PRIMARY KEY,
   first_name VARCHAR(50) NOT NULL,
   last_name VARCHAR(50) NOT NULL,
   home_phone VARCHAR(15),
   cell_phone VARCHAR(15),
   email VARCHAR(50),
   FOREIGN KEY (customer_id) REFERENCES Customer(customer_id))

CREATE TABLE Organization (
   customer_id NUMBER(8) PRIMARY KEY,
   legal_name VARCHAR(50) NOT NULL,
   status CHAR(1),
   established_date DATE,
   contact_person VARCHAR(100),
   work_phone VARCHAR(15)),
   FOREIGN KEY (customer_id) REFERENCES Customer(customer_id))
```



Concept Question 3

```
CREATE TABLE Organization (
   customer_id NUMBER(8) PRIMARY KEY,
   legal_name VARCHAR(50) NOT NULL,
   status CHAR(1),
   established_date DATE,
   contact_person VARCHAR(100),
   work_phone VARCHAR(15)),
   FOREIGN KEY (customer_id) REFERENCES Customer(id))

CREATE TABLE Commercial (
   customer_id NUMBER(8) PRIMARY KEY,
   segment CHAR(1),
   industry_code CHAR(5),
   total_employees INT,
   annual_revenue DOUBLE,
   FOREIGN KEY (customer_id) REFERENCES x(customer_id))

CREATE TABLE NonProfit (
   customer_id NUMBER(8) PRIMARY KEY,
   annual_contributions DOUBLE,
   tax_status CHAR(1),
   tax_deduction_rate DOUBLE,
   FOREIGN KEY (customer_id) REFERENCES x(customer_id))
```



What is x?

A. Customer
B. Organization
C. Either one
D. Neither one

Concept Question 4

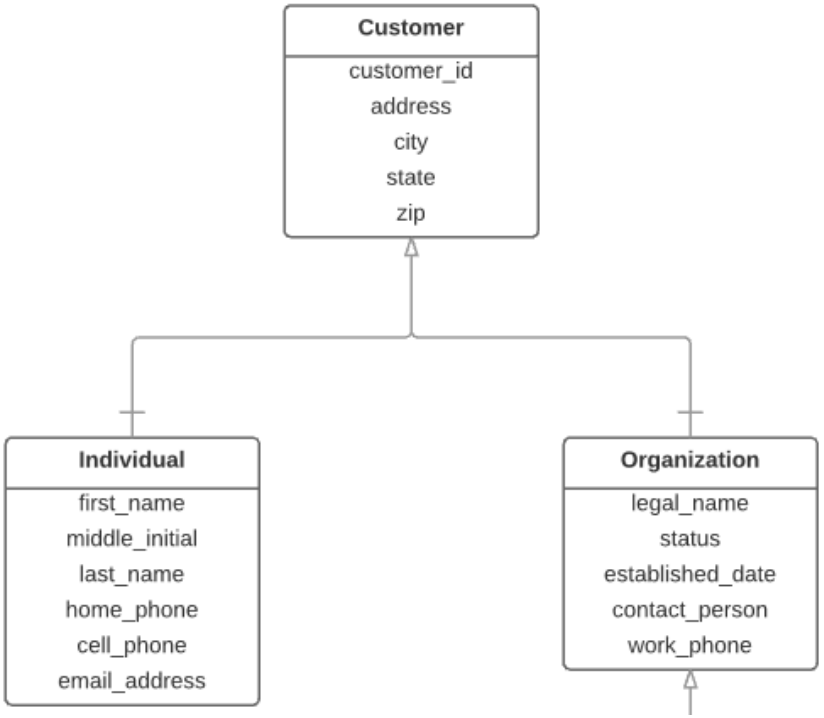
What kind of queries are awkward with this design?

```
CREATE TABLE Customer (
  customer_id INT PRIMARY KEY,
  address VARCHAR(50) NOT NULL,
  city VARCHAR(30) NOT NULL,
  state CHAR(2) NOT NULL,
  zip CHAR(5) NOT NULL)

CREATE TABLE Individual (
   customer_id NUMBER(8) PRIMARY KEY,
   first_name VARCHAR(50) NOT NULL,
   last_name VARCHAR(50) NOT NULL,
   ...
   FOREIGN KEY (customer_id) REFERENCES Customer(customer_id))

CREATE TABLE Organization (
   customer_id NUMBER(8) PRIMARY KEY,
   legal_name VARCHAR(50) NOT NULL,
   ...
   FOREIGN KEY (customer_id) REFERENCES Customer(customer_id))
```

A. Look-up all attributes of a customer by *last_name*
B. Look-up all attributes of a customer by *legal_name*
C. Look-up the attributes of all customers whose city = 'Austin'
D. All of the above

Customer Subclasses to Relations

```
CREATE TABLE Customer (
  customer_id INT PRIMARY KEY,
  address VARCHAR(50) NOT NULL,
  city VARCHAR(30) NOT NULL,
  state CHAR(2) NOT NULL,
  zip CHAR(5) NOT NULL,
  customer_type CHAR(1)
  CHECK customer_type IN ('I', 'O'))
```




```
CREATE TABLE Individual (
   customer_id NUMBER(8) PRIMARY KEY,
   first_name VARCHAR(50) NOT NULL,
   last_name VARCHAR(50) NOT NULL,
   home_phone VARCHAR(15),
   ...
   FOREIGN KEY (customer_id) REFERENCES Customer(customer_id))

CREATE TABLE Organization (
   customer_id NUMBER(8) PRIMARY KEY,
   legal_name VARCHAR(50) NOT NULL,
   status CHAR(1),
   ...
   FOREIGN KEY (customer_id) REFERENCES Customer(customer_id))
```

Homework for Next Time

- Read chapter 7 from the Beginning Database Design book
- Exercises at the end of chapter 7