

Figure 1: Q1

## 1 Question 1

### 1.1 1.1 Schema Design

Given the following ER Diagram, please write out the corresponding relational schemas. Be sure to: - remove redundant attributes - underline or bold primary keys - dashed-underline or italicize foreign keys

### 1.2 1.2 Queries

Next, use your schemas to answer the following queries in relational algebra.

1. What are the names of the publishers who have published at least one book written by “JK Rowling”?
2. What are the emails of customers who have strictly more than one ( $>1$ ) of some book in a shopping basket?

3. What are the pairs of different books that are in the same shopping basket?
4. What are the books that are in more than one (>1) shopping basket?
5. What are the names of customers with a book from every publisher in their shopping basket?

## 1.3 Solution

### 1.3.1 1.1 Schema Design

Note that naming conventions may differ student by student; so long as a relation is sensibly named and structured, it should pass. Critically, it should not have less expressiveness than our relations.

```
author (**name**, address, URL)
publisher (**name**, address, phone, URL)
book (**isbn**, title, year, price, author_name references author.name, publisher_name references
    publisher.name)
written_by(**name** references author.name, **ISBN** references book.ISBN)
contains (number, **book_isbn** references book.isbn, **basket_id** references customer.basket_id)
customer (**email**, name, address, phone, basket_id)
warehouse (**code**, address, phone)
stock (number, **book_isbn** references book.isbn, **warehouse_code** references warehouse.code)
```

### 1.3.2 1.2 Queries

1. What are the names of the publishers who have published at least one book written by “JK Rowling”?

$$books\_by\_jkr \leftarrow (book) \bowtie (\sigma_{name="JKRowling"}(author))$$

$$\pi_{name}(\rho_p(publisher) \bowtie_{p.name=ba.publisher\_name} \rho_{ba}(books\_by\_jkr))$$

2. What are the emails of customers who have strictly more than one (>1) of some book in a shopping basket?

$$full\_baskets \leftarrow \pi_{basket\_id}(\sigma_{number>1}(contains))$$

$$\pi_{email}(\rho_c(customer) \bowtie_{c.basket\_id=fb.basket\_id} \rho_{fb}(full\_baskets))$$

3. What are the pairs of books that are in the same shopping basket?

$$disjoint\_pairs \leftarrow \sigma_{b1.isbn \neq b2.isbn}(\rho_{b1}(contains) \times \rho_{b2}(contains))$$

$$\pi_{b1.isbn, b2.isbn}(\sigma_{b1.basket\_id=b2.basket\_id}(disjoint\_pairs))$$

4. What are the books that are in more than one (>1) shopping basket?

$$pairs \leftarrow \rho_{c1}(contains) \times \rho_{c2}(contains)$$

$$\pi_{c1.book\_isbn}(\sigma_{c1.book\_isbn=c2.book\_isbn}(\sigma_{c1.basket\_id \neq c2.basket\_id}(pairs)))$$

5. What are the names of customers with a book from every publisher in their shopping basket?

$$customer\_books \leftarrow \pi_{customer.name, contains.book\_isbn}((customer) \bowtie_{basket\_id} (contains))$$

$$customer\_publishers \leftarrow \pi_{cb.name, book.publisher\_name AS name}(\rho_{cb}(customer\_books) \bowtie_{isbn} (book))$$

$$\rho_{cp}(customer\_publishers) \div \pi_{name}(publisher)$$

## 2 Question 2

### 2.1 2.1 Initial Design

Let's say you are designing a new website for Barry's Beehive Bonanza (BBB) on Thursday, and Barry needs to store customer data. You decide to design a relational database for Barry. Given the following business requirements, create an ER Diagram: - The website should have users, each of which have an email, a name, a password, and any number of phone numbers. - The website should also have beehives, each of which has a name, a location, a size, and a single owner, which must be a user. - While users can own beehives (as described above), they can also subscribe to beehives. A beehive can have many subscribers and a user may subscribe to many beehives. - The website should also have honey types, each of which has a name and a locale. Each beehive should have at least one honey type that they produce.

### 2.2 2.2 Amendments

Oh no! Barry realized that there are actually three kinds of beehives: the Langstroths, the Warres and the Top Bars. Each one needs different information about them; use generalization to accomodate the following new information in your ER Diagram: - All Langstroths, Warres, and Top Bars are beehives, and each beehive is exactly one of a Langstroth, Warre, or Top Bar. - Langstroths have a number of racks attribute. - Warres have a wood type attribute. - Top Bars have a num\_levels attribute.

### 2.3 Solution

[Link](#)

## 3 Question 3

### 3.1 3.1 ER Diagrams

Given the following schema, please draw out the corresponding ER Diagram. Be sure to: - remove redundant attributes - use the ER Diagram elements as laid out in the reference section below

```
person(id, name, age, address, phone_num)
gardener(person_id, years_exp)
tree(tree_id, max_height, years_to_maturity)
trees_cultivated(gardener_id, tree_id)
```

Note that: - All gardeners are people, but not all people are gardeners - A gardener can cultivate many different kinds of trees, and a specific tree can be cultivated by many different gardeners.

### 3.2 3.2 Amendment 1

Now, draw the same ER diagram with the following added details: - Consider now that there are landscapers; landscapers are people, but they are not gardeners. There are no landscapers that are not people. Landscapers have an `hourly_rate`. - There are now also pilots; pilots are people, but are neither gardeners nor landscapers. Pilots fly planes, each of which has a name, weight, and `maximum_flying_height`. Pilots can fly many planes, and planes can be flown by many pilots. Pilots, especially low-flying ones, also have trees that they dislike. Pilots can have many disliked trees, and a tree can be disliked by many pilots.

### 3.3 3.3 Amendment 2

Now, all of these people can be members of a union. A union should have a name, a location of operation, and a founding year. There is each a gardener union, a landscaper union, and a pilot union. Consider now that there is no overlap between these unions. Not all of each profession are in a union, but all unions have at least one member. People can only be in one union. People without jobs cannot be in a union.

### 3.4 3.4 Bringing it back

Lastly, using your new ER Diagram, write a new relational schema for this scenario. If you need to change existing schemas, that is perfectly fine.

## 3.5 Solution

### 3.5.1 3.1-3 ER Diagrams

[Link](#)

### 3.5.2 3.4 Bringing it back

```
person(id, name, age, address, phone_num)
gardener(person_id references person.id, years_exp, union_id references gardener_union.union_id)
tree(tree_id, max_height, years_to_maturity)
```

```

trees_cultivated(gardener_id references gardener.person_id, tree_id references tree.tree_id)

landscaper(person_id references person.id, hourly_rate, union_id references landscaper_union.
            union_id)

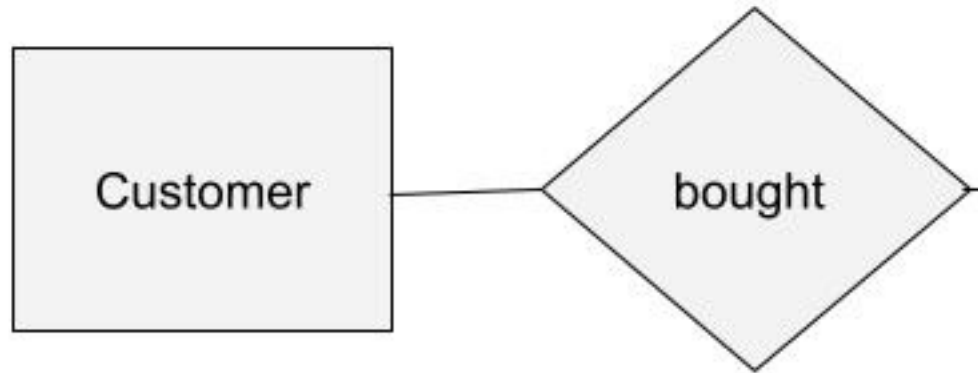
pilot(person_id references person.id, union_id references pilot_union.union_id)
plane(plane_id, name, weight, maximum_flying_height)
flies(pilot_id references pilot.person_id, plane_id references plane.plane_id)
dislikes_tree(pilot_id references pilot.person_id, tree_id references tree.tree_id)

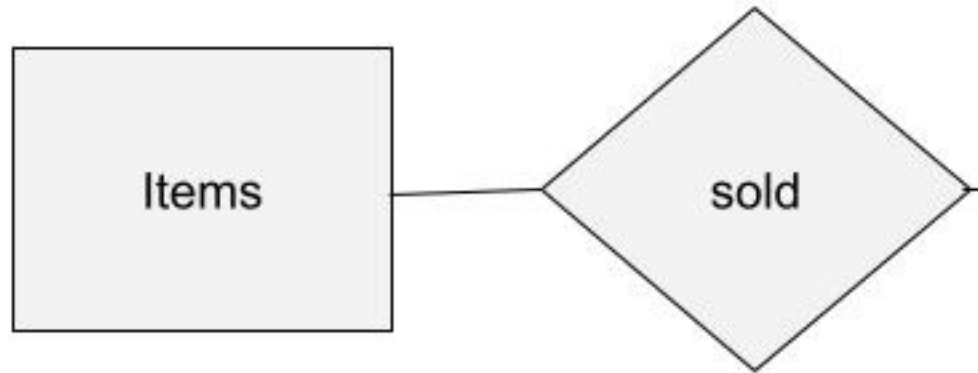
union(union_id, name, location, founding_year)
gardener_union(union_id references union.union_ud)
landscaper_union(union_id references union.union_ud)
pilot_union(union_id references union.union_ud)

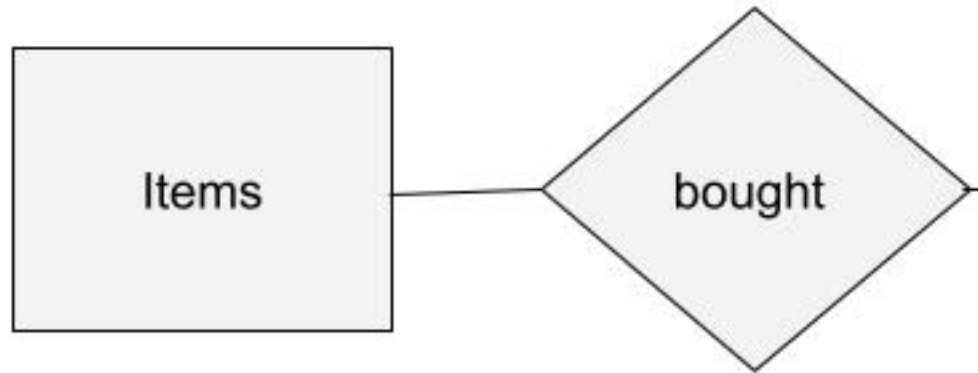
```

## 4 Question 4

Consider the following three ER Diagrams:









Items have an item ID, a name, and a price. Customers have a customer ID and a name. Salepersons have an employee ID and a name. Only one item can be sold at a time. Items are fancy, individually-made works of art that are each unique. Per the artist's request, a customer may only ever purchase one item. The artist is very insistent on this!

1. Which one(s) should we **NOT** use if our only query will be retrieving salespersons who helped customers? Why?
2. Which one(s) should we **NOT** use if our only query will be retrieving the items sold by salespersons? Why?
3. Suppose we are equally interested in gaining information between which customers are being helped by which salespersons and which salespersons are selling which items. Indicate which of the above we should choose and justify your choice.

## 4.1 Solution

### 4.1.1 4.1

We should NOT use the first method. It would require two joins instead of one.

### 4.1.2 4.2

We should NOT use the third method. It would require two joins instead of one.

### 4.1.3 4.3

We should use the second method. Since every relationship's cardinality is 1-1, we mark the foreign key references like in the schema below. The schema would look like:

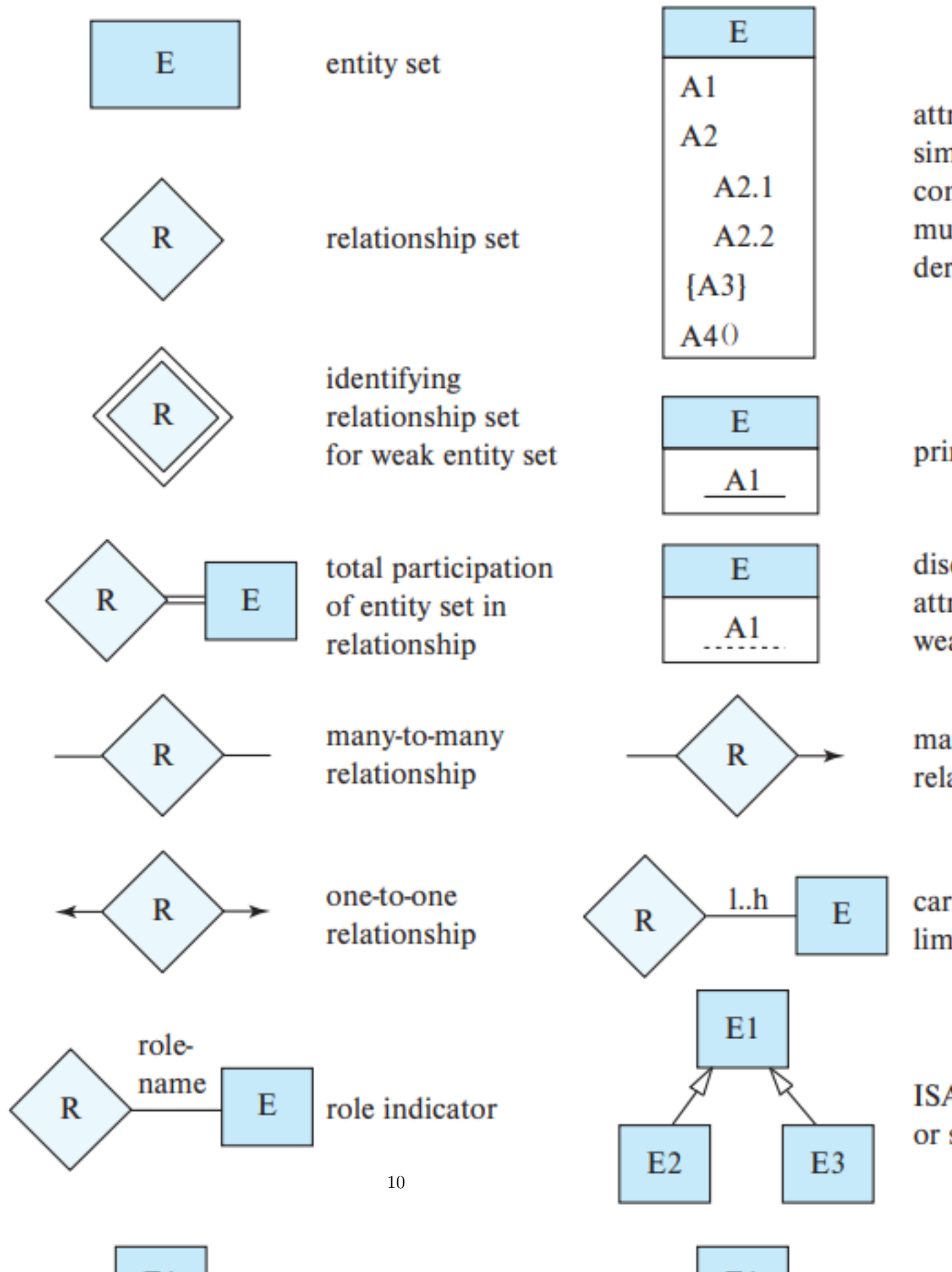
```
Customer(**cust_id**, name, helped_by)
Salesperson(**emp_id**, name)
Item(**item_id**, name, price, sold_by)
```

## 5 FAQ

- There are some discrepancies between the lecture slides and the textbook; when in doubt, follow the textbook as the source of truth.
- Feel free to use any software to draw your ER Diagram. We recommend Google Drawings or Draw.io.

## 6 Reference

---



## Feedback

As this is a new course, we appreciate any feedback you have for us! If you enjoyed this assignment, hated this assignment, or have other thoughts to share, please do so [here](#)!