

Name

UIN

Midterm Exam

March 13th, 2025

5:00pm - 6:15pm

CS480 - Database Systems

Results

Please leave this empty!

1.1

1.2

1.3

1.4

Sum

Instructions

- Try to answer all the questions using what you have learned in class. Keep hard questions until the end.
- **When writing a query, write the query in a way that it would work over all possible database instances and not just for the given example instance!**
- The exam is closed book and closed notes!
- **For relational algebra questions assume set semantics!**

Consider the following database schema and example instance for a restaurant database:

company

name	country	region
Turboshipping	USA	North America
Huron shipping	Canada	North America
Southwest	USA	North America
SOS Tao	China	Asia
EsBoatsche	Netherlands	Europe

ship

sname	capacity	lengthfeed	owner
Boaty McBoatface	4000	670	Turboshipping
SOS New York	3500	700	Turboshipping
The Haag	500	830	EsBoatsche
Lady Ann	2500	500	Southwest
Leviathan	6000	1010	SOS Tao

route

source	destination	sname	start_date	end_date
New York	Amsterdam	Boaty McBoatface	2025-03-10	2025-03-20
Amsterdam	Shanghai	The Haag	2025-03-03	2025-03-18
Seattle	Tokyo	Lady Ann	2025-04-01	2025-04-12
San Diego	Seattle	Leviathan	2025-03-28	2025-04-02

cargo

source	destination	sname	start_date	item	quantity
New York	Amsterdam	Boaty McBoatface	2025-03-10	coal	300
New York	Amsterdam	Boaty McBoatface	2025-03-10	oil	500
New York	Amsterdam	Boaty McBoatface	2025-03-10	concrete	120
Amsterdam	Shanghai	The Haag	2025-03-03	tuna	200
Amsterdam	Shanghai	The Haag	2025-03-03	veal	100
Seattle	Tokyo	Lady Ann	2025-04-01	freezer	250
Seattle	Tokyo	Lady Ann	2025-04-01	stove	2250
San Diego	Seattle	Leviathan	2025-03-28	coal	5000
San Diego	Seattle	Leviathan	2025-03-28	oil	1000
San Diego	Seattle	Leviathan	2025-03-28	gas	1100

Hints:

- Attributes with black background form the primary key of a relation (.e.g, name for relation restaurant)
- The attribute owner of relation ship is a foreign key to relation company

- The attribute `sname` of relation `route` is a foreign key to relation `ship`
- The attributes `source`, `destination`, `sname`, `start_date` of relation `cargo` form a foreign key to relation `route`
- All foreign keys have been created with the **CASCADE** option.

Part 1.1 Relational Algebra (Total: 28 Points)

Question 1.1.1 (8 Points)

Write a **relational algebra** expression that returns the **destination** and **item** of cargo that is shipped from New York or Amsterdam where the cargo is oil or coal and the quantity is larger than 200.

Solution

$$\pi_{price, calories}(\sigma_{(source=NewYork \vee source=Amsterdam) \wedge (item=oil \vee item=coal) \wedge quantity > 200}(\text{cargo}))$$

Question 1.1.2 (10 Points)

Write a **relational algebra** expression that returns the **sname** of ships and the **region** of their **owner** that are traveling at 2025-03-12 (they are on a route that starts before or on 2025-03-12 and ends on or after 2025-03-12).

Solution

$$\pi_{sname,region}(\text{company} \bowtie_{name=owner} \text{ship} \bowtie \sigma_{start_date \leq 2025-03-12 \wedge end_date \geq 2025-03-12}(\text{route}))$$

Question 1.1.3 (10 Points)

Write a **relational algebra** expression that returns the **source** and **destination** of the longest (in terms of time) shipping route.

Solution

$$\begin{aligned} Q_{len} &\leftarrow \rho_{source,destination,len}(\pi_{source,destination,end_date-start_date}(route)) \\ Q_{max} &\leftarrow \rho_{maxlen}(\gamma_{max(len)}(Q_{len})) \\ Q &\leftarrow \pi_{source,destination}(Q_{len} \bowtie_{len=maxlen} Q_{max}) \end{aligned}$$

Part 1.2 SQL - DDL (Total: 12 Points)

Question 1.2.1 (12 Points)

Write an **SQL DDL statement** that creates a two new relations **price** that records prices of items at certain locations and **item**. Relation **item** stores for each item a **name** (that is unique) and a **category** (e.g., **fuel**). We need to ensure that every entry in relation **cargo** corresponds to an item in relation **item**. Each entry of relation **price** records a **price** for a specific **item** at a certain location (**country** and **region**) for a specific **date**. The combination of **item**, **country** and **date** is unique. Each entry in relation **price** needs to correspond to an existing entry in relation **item**. Attribute **Price** should be a positive number with at most two digits after the dot.

Solution

```
CREATE TABLE item (  
    name TEXT PRIMARY KEY,  
    category TEXT  
);  
  
CREATE TABLE price (  
    price NUMERIC(8,2) CHECK (price > 0),  
    item TEXT REFERENCES item,  
    country TEXT,  
    region TEXT,  
    date DATE,  
    PRIMARY KEY (item, country, region, date)  
);  
  
ALTER TABLE cargo ADD CONSTRAINT cargo_item_fk FOREIGN KEY (item) REFERENCES item;
```

Part 1.3 SQL - Queries (Total: 40 Points)

Question 1.3.1 (11 Points)

Write an **SQL** query that returns the **sname**, **capacity**, and **lengthfeet** for ships that are heading to **Amsterdam** and are shipping **oil** on the route to **Amsterdam**.

Solution

```
SELECT s.sname, s.capacity, s.lengthfeet
FROM ship s,
     route r,
     cargo c
WHERE s.sname = r.sname
     AND r.source = c.source
     AND r.destination = c.destination
     AND r.sname = c.sname
     AND r.start_date = c.start_date
     AND c.item = 'oil'
     AND r.destination = 'Amsterdam';
```

Question 1.3.2 (14 Points)

Write an **SQL query** that returns for each **destination** the **sname** of the ship with the largest **capacity** that is shipping to this destination. The result schema should have attributes **sname** and **destination**.

Solution

```
WITH maxcap AS (  
    SELECT max(capacity) AS maxc,  
           destination  
    FROM ship s, route r  
    WHERE s.sname = r.sname  
    GROUP BY destination  
)  
  
SELECT r.destination, s.sname  
FROM ship s, route r  
WHERE s.sname = r.sname  
      AND capacity = (SELECT maxc  
                      FROM maxcap m  
                      WHERE m.destination = r.destination);
```

Question 1.3.3 (15 Points)

Write an **SQL query** that returns for each company the number of ships that are not on any shipping route. The result schema should have as attributes the **company** and the number of ships.

Solution

```
SELECT owner AS company, count(*) AS numships
FROM ship s
WHERE NOT EXISTS (SELECT *
                  FROM route r
                  WHERE r.sname = s.sname)
GROUP BY owner;
```

Part 1.4 SQL - Updates (Total: 20 Points)

Question 1.4.1 (7 Points)

Write an **SQL statement** that inserts a new ship with name `The unsinkable`, capacity 5000, a length of 900 feet, and owner `Turboshipping`.

Solution

```
INSERT INTO ship  
VALUES ( 'The unsinkable' , 5000, 900, 'Turboshipping' );
```

Question 1.4.2 (13 Points)

Write an **SQL statement** that deletes routes where the ship on this route is overloaded. A ship is overloaded if the sum of quantities of all the items it is loading is above its capacity.

Solution

```
DELETE FROM route r
WHERE (SELECT capacity FROM ship s WHERE s.sname = r.sname)
      < (SELECT sum(quantity)
        FROM cargo c
        WHERE r.sname = c.sname
              AND r.source = c.source
              AND r.destination = c.destination
              AND r.start_date = c.start_date);
```


