CS275 – Intro to Databases

SQL - Chap.4

- Retrieve the name and address of all employees who work for the 'Research' dept.
  \[ \pi_{\text{name, address}} (\sigma_{\text{Department} = \text{'Research'}} \text{Employee}) \]

- Retrieve the names of employees who have no dependents.
  \[ \rho(D(1 \rightarrow \text{ssn}), \pi_{\text{Dependent}}) \]
  \[ \pi_{\text{name, address}} (\pi_{\text{Employee}} - D) \bowtie \text{Employee} \]

SQL

- Structured Query Language (SQL)
  – Widely used
  – SQL:2011 is the latest standard
SQL

• Structured Query Language (SQL)
  – DDL: Data Definition Language
  – DML: Data Manipulation Language
  – Triggers
  – Embedded and Dynamic SQL
  – Client-server execution and remote access
  – Transaction management
  – Security

SQL

• Structured Query Language (SQL)
  – DDL: Data Definition Language
  – DML: Data Manipulation Language
  – Triggers
  – Embedded and Dynamic SQL
  – Client-server execution and remote access
  – Transaction management
  – Security

SQL

• Select
• Insert
• Update
• Delete
Select

- Basic form
  - SELECT [DISTINCT] select-list
  - FROM from-list
  - WHERE qualification

SELECT

- Sailors(sid:integer, sname:string, rating:integer, age:real)
- Boats(bid:integer, bname: string, color: string)
- Reserves(sid:integer, bid: integer, day: date)

- Find the names and ages of all sailors

SELECT

- Sailors(sid:integer, sname:string, rating:integer, age:real)
- Boats(bid:integer, bname: string, color: string)
- Reserves(sid:integer, bid: integer, day: date)

- Find the names and ages of all sailors
  - SELECT DISTINCT S.sname, S.age
  - FROM Sailors S
SELECT

- Sailors(sid:integer, sname:string, rating:integer, age:real)
- Boats(bid:integer, bname: string, color: string)
- Reserves(sid:integer, bid: integer, day: date)

- Find the names and ages of all sailors
  - SELECT S.sname, S.age
  - FROM Sailors S

SELECT

- Sailors(sid:integer, sname:string, rating:integer, age:real)
- Boats(bid:integer, bname: string, color: string)
- Reserves(sid:integer, bid: integer, day: date)

- Find all sailors with a rating over 7
  - SELECT S.sid, S.sname, S.rating, S.age
  - FROM Sailors AS S
  - WHERE S.rating > 7
SELECT

• Sailors(sid: integer, sname: string, rating: integer, age: real)
• Boats(bid: integer, bname: string, color: string)
• Reserves(sid: integer, bid: integer, day: date)

• Find all sailors with a rating over 7
  – SELECT *
  – FROM Sailors AS S
  – WHERE S.rating > 7

SELECT

• Sailors(sid: integer, sname: string, rating: integer, age: real)
• Boats(bid: integer, bname: string, color: string)
• Reserves(sid: integer, bid: integer, day: date)

• Find all sailors with a rating over 7
  – SELECT *
  – FROM Sailors
  – WHERE rating > 7

SELECT

• What really happens?
  1. Compute the cross-product of the tables in the from-list
  2. Delete rows that fail the qualification conditions
  3. Delete all columns that do not appear in the select-list
  4. If DISTINCT is specified, eliminate duplicate rows
SELECT
• Sailors(sid:integer, sname:string, rating:integer, age:real)
• Boats(bid:integer, bname: string, color: string)
• Reserves(sid:integer, bid: integer, day: date)

• Find the names of the sailors who have reserved boat 103

SELECT
• Sailors(sid:integer, sname:string, rating:integer, age:real)
• Boats(bid:integer, bname: string, color: string)
• Reserves(sid:integer, bid: integer, day: date)

• Find the names of the sailors who have reserved boat 103
  — SELECT S.sname
  — FROM Sailors S, Reserves R
  — WHERE S.sid=R.sid AND R.bid = 103

SELECT
• Sailors(sid:integer, sname:string, rating:integer, age:real)
• Boats(bid:integer, bname: string, color: string)
• Reserves(sid:integer, bid: integer, day: date)

• Find the names of the sailors who have reserved boat 103
  — SELECT S.sname
  — FROM Sailors S, Reserves R
  — WHERE R.bid = 103

X
SELECT

- Sailors(sid:integer, sname:string, rating:integer, age:real)
- Boats(bid:integer, bname: string, color: string)
- Reserves(sid:integer, bid: integer, day: date)

- Find the names of the sailors who have reserved boat 103
  - SELECT sname
  - FROM Sailors, Reserves
  - WHERE Sailors.sid=Reserves.sid AND bid = 103

SELECT

- Sailors(sid:integer, sname:string, rating:integer, age:real)
- Boats(bid:integer, bname: string, color: string)
- Reserves(sid:integer, bid: integer, day: date)

- Find the names of the sailors who have reserved boat 103
  - SELECT sname
  - FROM Sailors NATURAL JOIN Reserves
  - WHERE bid = 103

SELECT

- Sailors(sid:integer, sname:string, rating:integer, age:real)
- Boats(bid:integer, bname: string, color: string)
- Reserves(sid:integer, bid: integer, day: date)

- Find the ID’s of the sailors who have reserved a red boat
SELECT

- Sailors(sid:integer, sname:string, rating:integer, age:real)
- Boats(bid:integer, bname: string, color: string)
- Reserves(sid:integer, bid: integer, day: date)

- Find the ID's of the sailors who have reserved a red boat
  - SELECT R.sid
  - FROM Boats B, Reserves R
  - WHERE Boats.bid=Reserves.bid AND B.color='red'

SELECT

- Sailors(sid:integer, sname:string, rating:integer, age:real)
- Boats(bid:integer, bname: string, color: string)
- Reserves(sid:integer, bid: integer, day: date)

- Find the names of the sailors who have reserved a red boat
  - SELECT S.sname
  - FROM Sailors S, Boats B, Reserves R
  - WHERE S.sid=R.sid AND B.bid=R.bid AND B.color='red'
SELECT

- Sailors(sid:integer, sname:string, rating:integer, age:real)
- Boats(bid:integer, bname: string, color: string)
- Reserves(sid:integer, bid: integer, day: date)

• Find the names of the sailors who have reserved at least one boat
  - SELECT S.Sname
  - FROM Sailors S, Reserves R
  - WHERE S.sid=R.sid

SELECT

- Sailors(sid:integer, sname:string, rating:integer, age:real)
- Boats(bid:integer, bname: string, color: string)
- Reserves(sid:integer, bid: integer, day: date)

• Find the names of the sailors who have reserved at least one boat
  - SELECT DISTINCT S.Sname
  - FROM Sailors S, Reserves R
  - WHERE S.sid=R.sid
SELECT

• Sailors(sid:integer, sname:string, rating:integer, age:real)
• Boats(bid:integer, bname: string, color: string)
• Reserves(sid:integer, bid: integer, day: date)

• Compute increments for the rating of people who have sailed two different boats on the same day.

SELECT

• Sailors(sid:integer, sname:string, rating:integer, age:real)
• Boats(bid:integer, bname: string, color: string)
• Reserves(sid:integer, bid: integer, day: date)

• Compute increments for the rating of people who have sailed two different boats on the same day
  — SELECT S.sname, S.rating+1 AS rating
  — FROM Sailors S, Reserves R1, Reserves R2
  — WHERE S.sid = R1.sid AND S.sid = R2.sid AND R1.day = R2.day AND R1.bid <> R2.bid

SELECT

• Sailors(sid:integer, sname:string, rating:integer, age:real)
• Boats(bid:integer, bname: string, color: string)
• Reserves(sid:integer, bid: integer, day: date)

• Find the ages of sailors whose name begins and ends with B and has at least three characters
SELECT
• Sailors(sid:integer, sname:string, rating:integer, age:real)
• Boats(bid:integer, bname: string, color: string)
• Reserves(sid:integer, bid: integer, day: date)

• Find the ages of sailors whose name begins and ends with B and has at least three characters
  — SELECT S.age
  — FROM Sailors S
  — WHERE S.sname LIKE 'B_%B'

SELECT
• Sailors(sid:integer, sname:string, rating:integer, age:real)
• Boats(bid:integer, bname: string, color: string)
• Reserves(sid:integer, bid: integer, day: date)

• Find the ages of sailors whose name begins and ends with B and has at least three characters
  — SELECT S.age
  — FROM Sailors S
  — WHERE S.sname LIKE 'B_B'

SELECT
• Sailors(sid:integer, sname:string, rating:integer, age:real)
• Boats(bid:integer, bname: string, color: string)
• Reserves(sid:integer, bid: integer, day: date)

• Find the ages of sailors whose name begins and ends with B and has at least three characters
  — SELECT S.age
  — FROM Sailors S
  — WHERE S.sname LIKE 'B%B'
SELECT
• Sailors(sid:integer, sname:string, rating:integer, age:real)
• Boats(bid:integer, bname: string, color: string)
• Reserves(sid:integer, bid: integer, day: date)

• Find the names of sailors who have reserved a red or a green boat

SELECT
• Sailors(sid:integer, sname:string, rating:integer, age:real)
• Boats(bid:integer, bname: string, color: string)
• Reserves(sid:integer, bid: integer, day: date)

• Find the names of sailors who have reserved a red or a green boat
  — SELECT S.sname
  — FROM Sailors S, Reserves R, Boats B
  — WHERE S.sid = R.sid AND R.bid = B.bid AND (B.color='red'
  or B.color='green')

SELECT
• Sailors(sid:integer, sname:string, rating:integer, age:real)
• Boats(bid:integer, bname: string, color: string)
• Reserves(sid:integer, bid: integer, day: date)

• Find the names of sailors who have reserved both a red and a green boat
SELECT

- Sailors(sid:integer, sname:string, rating:integer, age:real)
- Boats(bid:integer, bname:string, color:string)
- Reserves(sid:integer, bid:integer, day:date)

- Find the names of sailors who have reserved both a red and a green boat
  - SELECT S.sname
  - FROM Sailors S, Reserves R1, Reserves R2, Boats B1, Boats B2
  - WHERE S.sid = R1.sid AND R1.bid = B1.bid AND B1.color='red'

Note: There isn't INTERSECT in MySQL

SELECT

- Sailors(sid:integer, sname:string, rating:integer, age:real)
- Boats(bid:integer, bname:string, color:string)
- Reserves(sid:integer, bid:integer, day:date)

- Find the names of sailors who have reserved both a red or a green boat
  - SELECT S1.sname
  - FROM Sailors S1, Reserves R1, Boats B1
  - WHERE S1.sid = R1.sid AND R1.bid = B1.bid AND B1.color='red'
  - INTERSECT
  - SELECT S2.sname
  - FROM Sailors S2, Reserves R2, Boats B2
  - WHERE S2.sid = R2.sid AND R2.bid = B2.bid AND B2.color='green'

SELECT

- Sailors(sid:integer, sname:string, rating:integer, age:real)
- Boats(bid:integer, bname:string, color:string)
- Reserves(sid:integer, bid:integer, day:date)

- Find the names of sailors who have reserved both a red or a green boat
  - SELECT S1.sname
  - FROM Sailors S1, Reserves R1, Boats B1
  - WHERE S1.sid = R1.sid AND R1.bid = B1.bid AND B1.color='red'
  - UNION
  - SELECT S2.sname
  - FROM Sailors S2, Reserves R2, Boats B2
  - WHERE S2.sid = R2.sid AND R2.bid = B2.bid AND B2.color='green'
SELECT

• Sailors(sid:integer, sname:string,rating:integer,age:real)
• Boats(bid:integer, bname:string, color:string)
• Reserves(sid:integer, bid:integer, day:date)

• Find the sids of all sailors who have reserved a red boat but not a green boat
  — SELECT R1.sid
  — FROM Reserves R1, Boats B1
  — WHERE R1.bid = B1.bid AND B1.color='red'
  — EXCEPT
  — SELECT R2.sid
  — FROM Reserves R2, Boats B2
  — WHERE R2.bid = B2.bid AND B2.color='green'

• Note: There isn’t EXCEPT in MySQL

SELECT

• Sailors(sid:integer, sname:string,rating:integer,age:real)
• Boats(bid:integer, bname:string, color:string)
• Reserves(sid:integer, bid:integer, day:date)

• Find the sids of all sailors who have a rating of 10 or reserved boat 104
  — SELECT S.sid
  — FROM Sailors S
  — WHERE S.rating = 10
  — UNION
  — SELECT R.sid
  — FROM Reserves R
  — WHERE R.bid = 104