CS275 – Intro to Databases

Making Tables

SELECT: Nested Queries

- 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 boat 103
  - SELECT S.sname
  - FROM Sailors S, Reserves R
  - WHERE S.sid = R.sid AND R.bid = 103

SELECT: Nested Queries

- 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 boat 103
  - SELECT S.sname
  - FROM Sailors S
  - WHERE S.sid IN (SELECT R.sid
    FROM Reserves R
    WHERE R.bid = 103)
SELECT: Nested Queries

- **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 not reserved boat 103
  - SELECT S.sname
  - FROM Sailors S
  - WHERE S.sid NOT IN (SELECT R.sid
    FROM Reserves R
    WHERE R.bid = 103)

- Find the **names** of sailors who have reserved a red boat
  - SELECT:
  - FROM Sailors S
  - WHERE S.sid IN (SELECT R.sid
    FROM Reserves R
    WHERE R.bid IN (SELECT B.bid
      FROM BOAT B
      WHERE B.color = 'red'))
SELECT: Nested Queries

- 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 not reserved a red boat
  - SELECT S.sname
  - FROM Sailors S
  - WHERE S.sid NOT IN (SELECT R.sid
    FROM Reserves R
    WHERE R.bid IN (SELECT B.bid
      FROM Boats B
      WHERE B.color = 'red'))

- Find the names of sailors who have reserved boat 103
  - SELECT S.sname
  - FROM Sailors S
  - WHERE EXISTS (SELECT *
    FROM Reserves R
    WHERE R.bid =103 AND R.sid = S.sid)
SELECT: Nested Queries

- 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 not reserved boat 103
  - SELECT S.sname
  - FROM Sailors S
  - WHERE NOT EXISTS (SELECT *
    FROM Reserves R
    WHERE R.bid = 103 AND R.sid = S.sid )

SELECT: Nested Queries

- 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 sailors whose rating is better than Horatio
  - SELECT S.sid
  - FROM Sailors S
  - WHERE S.rating > ANY (SELECT S2.rating
    FROM Sailors S2
    WHERE S2.sname = 'Horatio' )
SELECT: Nested Queries

- 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 sailors with the highest rating

---

SELECT: Nested Queries

- 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 sailors with the highest rating
  - SELECT S.sid
  - FROM Sailors S
  - WHERE S.rating >= ALL (SELECT S2.rating
  - FROM Sailors S2)

---

SQL Databases

- Create a database
  - CREATE database dbname;
- Set current database
  - USE dbname;
- List the databases
  - SHOW databases;
- Show information about a table
  - DESCRIBE tablename;
Privileges

<table>
<thead>
<tr>
<th>Privilege</th>
<th>Applies To</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>SELECT</td>
<td>Tables, Columns</td>
<td></td>
</tr>
<tr>
<td>INSERT</td>
<td>Tables, Columns</td>
<td></td>
</tr>
<tr>
<td>UPDATE</td>
<td>Tables, Columns</td>
<td></td>
</tr>
<tr>
<td>DELETE</td>
<td>Tables</td>
<td></td>
</tr>
<tr>
<td>INDEX</td>
<td>Tables</td>
<td></td>
</tr>
<tr>
<td>ALTER</td>
<td>Tables</td>
<td></td>
</tr>
<tr>
<td>CREATE</td>
<td>Databases, Tables</td>
<td></td>
</tr>
<tr>
<td>DROP</td>
<td>Databases, Tables</td>
<td></td>
</tr>
</tbody>
</table>

There are more, but not that you will need here.

Privileges

- **ALL**
  - Grants ALL THE THINGS!
- **USAGE**
  - Grants none of the things. 😊
  - Place holder to create accounts but give them no privileges

Privilege Examples

- Assigns all permission on mydb tables
  - GRANT ALL ON mydb.* TO 'someuser';
- Assigns CREATE permissions on mydb
  - GRANT CREATE ROUTINE ON mydb.* TO 'someuser';
- Grants column specific permissions
  - GRANT SELECT (col1), INSERT (col1,col2) ON mydb.mytbl TO 'someuser';
Creating Tables

• You can use the GUI
• You can use the create table command
  – CREATE TABLE example_autoincrement {
    id INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
    data VARCHAR(100)
  };

Table Creation With Foreign Keys

CREATE TABLE parent (id INT NOT NULL,
  PRIMARY KEY (id)
) ENGINE=INNODB;

CREATE TABLE child (id INT,
  parent_id INT,
  FOREIGN KEY (parent_id) REFERENCES parent(id) ON
  DELETE CASCADE ) ENGINE=INNODB;

INSERT

INSERT INTO phonebook(phone, firstname, lastname,
  address) VALUES('+1 123 456 7890', 'John', 'Doe', 'North America');
Data Types

- INT
- FLOAT
- DECIMAL
- DATE
- TIME
- TIMESTAMP
  - Many format options

More Data Types

- CHAR(M)
  - Fixed length
- VARCHAR(M)
  - Variable length up to M
- TEXT