Database Management Systems
L9

Umass Boston
Summer 2023
Cristina Maier

Some slides are based on “Database Management System, 3rd ed., by Ramakrishnan and Gehrke

SQL Exercises

Database Schema:

Singers(sid:int, name:string, city:string, state:string, age:real, rating:int)
Rents(sid:int, mid:int, rentdate:date)
Instruments(mid:int, category:string, myear:int, brand:string, model:string, dailyfee:real)

Exercise 1
WRONG SOLUTION

- Singers(<u>sid:int</u>, name:string, city:string, state:string, age:real, rating:int)
  Rents(<u>sid:int, mid:int</u>, rentdate:date)
  Instruments(<u>mid:int</u>, category:string, myear:int, brand:string, model:string, dailyfee:real)

- Write the SQL statement to find the id and name of singers that rented both guitar and violin instruments.

```
SELECT s.sid, s.name
FROM Singers s, Rents r, Instruments i
WHERE s.sid=r.sid AND r.mid=i.mid AND
          i.category='violin' AND i.category='guitar';
```

Exercise 1
Correct solution

- Singers(<u>sid:int</u>, name:string, city:string, state:string, age:real, rating:int)
  Rents(<u>sid:int, mid:int</u>, rentdate:date)
  Instruments(<u>mid:int</u>, category:string, myear:int, brand:string, model:string, dailyfee:real)

- Write the SQL statement to find the id and name of singers that rented both guitar and violin instruments.

```
(SELECT s.sid, s.name
FROM Singers s, Rents r, Instruments i
WHERE s.sid=r.sid AND r.mid=i.mid AND
          i.category='violin')
INTERSECT
(SELECT s.sid, s.name
FROM Singers s, Rents r, Instruments i
WHERE s.sid=r.sid AND r.mid=i.mid AND
           i.category='guitar’)

;
```

Exercise 2
WRONG SOLUTION

- Singers(sid:int, name:string, city:string, state:string, age:real, rating:int)
  Rents(sid:int, mid:int, rentdate:date)
  Instruments(mid:int, category:string, myear:int, brand:string, model:string, dailyfee:real)

- Write the SQL statement to find the id and names of players who rented ONLY instruments of category guitar.

```
SELECT s.sid, s.name
FROM Singers s, Rents r, Instruments i
WHERE s.sid=r.sid AND r.mid=i.mid
            AND category='guitar'
```

Exercise 2
WRONG SOLUTION

- Singers(sid:int, name:string, city:string, state:string, age:real, rating:int)
  Rents(sid:int, mid:int, rentdate:date)
  Instruments(mid:int, category:string, myear:int, brand:string, model:string, dailyfee:real)

- Write the SQL statement to find the id and names of players who rented only instruments of category guitar.

```
SELECT s.sid, s.name
FROM Singers s, Rents r, Instruments i
WHERE s.sid=r.sid AND r.mid=i.mid
            AND i.category='guitar' AND i.category!
='‘guitar'
```

Exercise 2
Correct solution

- Singers(sid:int, name:string, city:string, state:string, age:real, rating:int)
  Rents(sid:int, mid:int, rentdate:date)
  Instruments(mid:int, category:string, myear:int, brand:string, model:string, dailyfee:real)

- Write the SQL statement to find the id and names of players who rented only instruments of category guitar.

```
(SELECT s.sid, s.name
FROM Singers s, Rents r, Instruments i
WHERE s.sid=r.sid AND r.mid=i.mid
          AND category='guitar')
MINUS
(SELECT s.sid, s.name
FROM Singers s, Rents r, Instruments i
WHERE s.sid=r.sid AND r.mid=i.mid
        AND category!='guitar'  )
```

Exercise 3
WRONG SOLUTION

- Singers(<u>sid:int</u>, name:string, city:string, state:string, age:real, rating:int)
  Rents(<u>sid:int, mid:int</u>, rentdate:date)
  Instruments(<u>mid:int</u>, category:string, myear:int, brand:string, model:string, dailyfee:real)

- Write the SQL query to find the id and name of singers whose name start with letter 'D' and who never rented some instrument having brand 'yamaha'.

```
SELECT s.sid, s.name
FROM Singers s, Rents r, Instruments I
WHERE s.sid=r.sid AND r.mid=i.mid AND
        s.name LIKE ‘D%' AND i.brand <> 'yamaha'
```

Exercise 3
SOLUTION THAT STILL HAS A PROBLEM

- Singers(<u>sid:int</u>, name:string, city:string, state:string, age:real, rating:int)
  Rents(<u>sid:int, mid:int</u>, rentdate:date)
  Instruments(<u>mid:int</u>, category:string, myear:int, brand:string, model:string, dailyfee:real)

- Write the SQL query to find the id and name of singers whose name start with letter 'D' and who never rented some instrument having brand 'yamaha'.

```
(SELECT s.sid, s.name
FROM Singers s, Rents r, Instruments i
WHERE s.sid=r.sid AND r.mid=i.mid AND
        s.name LIKE 'D%')

MINUS
(SELECT s.sid, s.name
FROM Singers s, Rents r, Instruments i
WHERE s.sid=r.sid AND r.mid=i.mid AND
        i.brand='yamaha');
```

Exercise 3
Correct solution

- Singers(<u>sid:int</u>, name:string, city:string, state:string, age:real, rating:int)
  Rents(<u>sid:int, mid:int</u>, rentdate:date)
  Instruments(<u>mid:int</u>, category:string, myear:int, brand:string, model:string, dailyfee:real)

- Write the SQL query to find the id and name of singers whose name start with letter 'D' and who never rented some instrument having brand ‘yamaha'.

```
(SELECT s.sid, s.name
FROM Singers s
WHERE s.name LIKE 'D%')

MINUS

(SELECT s.sid, s.name
FROM Singers s, Rents r, Instruments i
WHERE s.sid=r.sid AND r.mid=i.mid AND
          i.brand='yamaha');
```

Topics

- Introduction to DBMS
- Relational Data Model
- *Relational Algebra*
- Conceptual Design: the Entity-Relationship Model
- Structured Query Language (SQL)
- Database Security and Authorization
- Schema Refinement and Normal Forms
- Application Development (Java, Python)
- Some NoSQL topics (If time permitted)

Application Development

- Connect to DBMS and run queries from Applications:
  - Python
  - Java

Information needed for Connection

- DBMS hostname
- DB name
- Username
- Password
- DRIVER
- Port (different DBMS use specific ports)

Python Oracle DRIVER

- Python-oracledb
- Already installed on UNIX machine
- (https://oracle.github.io/python-oracledb/ .)

Sample Python code

```
import oracledb
connection =
oracledb.connect(user=<username>,
password=<password>, dsn=<hostname/
dbname>)
```

Once you run this command, variable connection contains an established connection to the Oracle DBMS

Python

- You can further use connection var to execute queries against DBMS
- We will see how to do this using
  - Pandas library
  - Cursor

Pandas

- Python library that can be used for data manipulation and analysis
- Very useful for data science, machine learning, data mining

Pandas Dataframes

- Data structure that mimics a table (2d rows and columns)
- Populate a dataframe
  - Read it from a CSV
  - Create it manually
  - Read data from a Database (execute an SQL Query against a DBMS)

Pandas

- Dataframe
- Index
  - index=0 refers to rows
  - index=1 refers to columns

Some operations

- Let's say we have a variable df of type Pandas DataFrame
- df.head(n) returns the first rows
- df.head() returns the first 5 rows
- df.shape returns a tuple (numberRows,numberColumns)
- df.columns returns a list with column names

Some more operations

- df[['col1','col2']] selects only columns col1 and col2 from the dataframe.

- If only one column named col1 is selected you can use df.col1 instead of df[['col1']]

- Note: column names in Pandas are case-sensitive!!!

Some more operations

- Aggregates
- df.agg([listaggregates])
  - Where list aggregates could contain aggregate functions such as: 'sum','min', 'max', ‘mean'
    - E.g. df.agg(['mean','max’])
- If only aggregates on specific columns, we have to select those columns first
  - E.g. to aggregate on rating and age columns
  - df[['rating','age']].agg(['mean','max'])

Python-Oracle Sample Code

- samplePythonAndSQL.py

Some more Pandas operations

- Assuming we have a variable df of type Pandas DataFrame

- df[df['col']=x]
  - Only keeps the records in which column col has value x

- df.groupby([col]).mean()

- df.groupby([col]).agg(['min','max'])

Python Code Session

- Pandas Python code
- Running samplePythonAndSQL

Python

- Pandas
- Cursor

SQL Session

- Practice running .py file

- python3 samplePythonSQLCursor.py

Questions?