WISCONSIN
UNIVERSITY OF WISCONSIN–MADISON

CS 564: Database Management Systems

Lecture 5: Advanced SQL II

Xiangyao Yu

2/2/2024

Announcements

Form groups for later assignments

- Due on **Feb 5 (Monday), 11:59pm**

Assignment #1. SQL

- Due on **Feb 7 (Wednesday), 11:59pm**

Module A1: SQL

SQL: Basics I

SQL: Basics II

Advanced SQL I

Advanced SQL II

- – Set operations and nested queries

Outline of this Lecture

SQL: Set Operators

- UNION/EXCEPT/INTERSECT
- duplicates in SQL

SQL: Nested Queries

- IN/EXISTS/ALL
- correlated queries

Set Operators Refresher

$R = \{1, 2, 3\}$ $S = \{1, 2, 4, 5\}$

Intersection: $R \cap S = \{1, 2\}$

Set Operators Refresher

$R = \{1, 2, 3\}$ $S = \{1, 2, 4, 5\}$

Intersection: $R \cap S = \{1, 2\}$

Union: $R \cup S = \{1, 2, 3, 4, 5\}$

Set Operators Refresher

$R = \{1, 2, 3\}$ $\qquad$ $S = \{1, 2, 4, 5\}$

Intersection: $R \cap S = \{1, 2\}$

Union: $R \cup S = \{1, 2, 3, 4, 5\}$

Difference: $R - S = \{3\}$

$S - R = \{4, 5\}$

Set Operators in SQL

SQL supports set operations between the outputs of subqueries:

(subquery) **`INTERSECT`** (subquery)

(subquery) **`UNION`** (subquery)

(subquery) **`EXCEPT`** (subquery)

Set Operators: INTERSECT

```
SELECT A FROM R
INTERSECT
SELECT A FROM S;
```

R

A
1
1
1
2
3

S

A
1
1
2
2
4
5

output

A
1
2

Returns the tuples that belong in both subquery results

Set Operators: UNION

```
SELECT A FROM R
UNION
SELECT A FROM S;
```

R

A
1
1
1
2
3

S

A
1
1
2
2
4
5

output

A
1
2
3
4
5

Returns the tuples that belong in either subquery results

Set Operators: EXCEPT

```
SELECT A FROM R
EXCEPT
SELECT A FROM S;
```

R

A
1
1
1
2
3

S

A
1
1
2
2
4
5

output

A
3

Returns the tuples that belong in the first and not the second subquery result

Keyword **ALL**

When using set operators, SQL eliminates all duplicate tuples

We can modify the semantics by using the keyword **`ALL`** (e.g. **`UNION ALL`**)

When using **`ALL`**, the operators are evaluated using multiset (or bag) semantics

Set Operators: INTERSECT ALL

```
SELECT A FROM R
INTERSECT ALL
SELECT A FROM S;
```

R

A
1
1
1
2
3

S

A
1
1
2
2
4
5

output

A
1
1
2

The number of copies of each tuple is the minimum of the number of copies in the subqueries

Set Operators: UNION ALL

```
SELECT A FROM R
UNION ALL
SELECT A FROM S;
```

R

A
1
1
1
2
3

S

A
1
1
2
2
4
5

The number of copies of each tuple is the sum of the number of copies in the subqueries

output

A
1
1
1
1
1
2
2
2
3
4
5

Set Operators: EXCEPT ALL

```
SELECT A FROM R
EXCEPT ALL
SELECT A FROM S;
```

R

A
1
1
1
2
3

S

A
1
1
2
2
4
5

output

A
1
3

The number of copies of each tuple is the difference (if positive) of the number of copies in the subqueries

Example Database

Sailors (**sid**: integer, **sname**: string, **rating**: integer, **age**: real)

Boats (**bid**: integer, **bname**: string, **color**: string)

Reserves (**sid**: integer, **bid**: integer, **day**: date)

Sailors

sid	sname	rating	age
22	Dustin	7	45
29	Brutus	1	33
31	Lubber	8	55
32	Andy	8	25
58	Rusty	10	35
64	Horatio	7	35
71	Zorba	10	16
74	Horato	9	35
85	Art	3	25.5
95	Bob	3	63.5

Boats

bid	bname	color
101	Interlake	blue
102	Interlake	red
103	Clipper	green
104	Marine	red

Reserves

sid	bid	day
22	101	10/10/98
22	102	10/10/98
22	103	10/8/98
22	104	10/7/98
31	102	11/10/98
31	103	11/6/98
31	104	11/12/98
64	101	9/5/98
64	102	9/8/98
74	103	9/8/98

Example Queries

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

Sailors

sid	sname	rating	age
22	Dustin	7	45
29	Brutus	1	33
31	Lubber	8	55
32	Andy	8	25
58	Rusty	10	35
64	Horatio	7	35
71	Zorba	10	16
74	Horato	9	35
85	Art	3	25.5
95	Bob	3	63.5

Reserves

sid	bid	day
22	101	10/10/98
22	102	10/10/98
22	103	10/8/98
22	104	10/7/98
31	102	11/10/98
31	103	11/6/98
31	104	11/12/98
64	101	9/5/98
64	102	9/8/98
74	103	9/8/98

Boats

bid	bname	color
101	Interlake	blue
102	Interlake	red
103	Clipper	green
104	Marine	red

Example Queries

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

```
SELECT
FROM
WHERE
```

Sailors

sid	sname	rating	age
22	Dustin	7	45
29	Brutus	1	33
31	Lubber	8	55
32	Andy	8	25
58	Rusty	10	35
64	Horatio	7	35
71	Zorba	10	16
74	Horato	9	35
85	Art	3	25.5
95	Bob	3	63.5

Reserves

sid	bid	day
22	101	10/10/98
22	102	10/10/98
22	103	10/8/98
22	104	10/7/98
31	102	11/10/98
31	103	11/6/98
31	104	11/12/98
64	101	9/5/98
64	102	9/8/98
74	103	9/8/98

Boats

bid	bname	color
101	Interlake	blue
102	Interlake	red
103	Clipper	green
104	Marine	red

Example Queries

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

```
SELECT S.name
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)
```

Sailors

sid	sname	rating	age
22	Dustin	7	45
29	Brutus	1	33
31	Lubber	8	55
32	Andy	8	25
58	Rusty	10	35
64	Horatio	7	35
71	Zorba	10	16
74	Horato	9	35
85	Art	3	25.5
95	Bob	3	63.5

Reserves

sid	bid	day
22	101	10/10/98
22	102	10/10/98
22	103	10/8/98
22	104	10/7/98
31	102	11/10/98
31	103	11/6/98
31	104	11/12/98
64	101	9/5/98
64	102	9/8/98
74	103	9/8/98

Boats

bid	bname	color
101	Interlake	blue
102	Interlake	red
103	Clipper	green
104	Marine	red

Example Queries

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’
UNION
SELECT S.sname
FROM   Sailors S, Reserves R, Boats B
WHERE  S.sid = R.sid AND R.bid = B.bid AND B.color = ‘green’
```

Sailors

sid	sname	rating	age
22	Dustin	7	45
29	Brutus	1	33
31	Lubber	8	55
32	Andy	8	25
58	Rusty	10	35
64	Horatio	7	35
71	Zorba	10	16
74	Horato	9	35
85	Art	3	25.5
95	Bob	3	63.5

Reserves

sid	bid	day
22	101	10/10/98
22	102	10/10/98
22	103	10/8/98
22	104	10/7/98
31	102	11/10/98
31	103	11/6/98
31	104	11/12/98
64	101	9/5/98
64	102	9/8/98
74	103	9/8/98

Boats

bid	bname	color
101	Interlake	blue
102	Interlake	red
103	Clipper	green
104	Marine	red

Example Queries

Find the names of sailors who have reserved red boats but not green boats

Sailors

sid	sname	rating	age
22	Dustin	7	45
29	Brutus	1	33
31	Lubber	8	55
32	Andy	8	25
58	Rusty	10	35
64	Horatio	7	35
71	Zorba	10	16
74	Horato	9	35
85	Art	3	25.5
95	Bob	3	63.5

Reserves

sid	bid	day
22	101	10/10/98
22	102	10/10/98
22	103	10/8/98
22	104	10/7/98
31	102	11/10/98
31	103	11/6/98
31	104	11/12/98
64	101	9/5/98
64	102	9/8/98
74	103	9/8/98

Boats

bid	bname	color
101	Interlake	blue
102	Interlake	red
103	Clipper	green
104	Marine	red

Example Queries

Find the names of sailors who have reserved red boats but not green boats

```
SELECT S.name
FROM   Sailors S, Reserves R, Boats B
WHERE  S.sid = R.sid AND R.bid = B.bid AND B.color = ‘red’
EXCEPT
SELECT S.name
FROM   Sailors S, Reserves R, Boats B
WHERE  S.sid = R.sid AND R.bid = B.bid AND B.color = ‘green’
```

Sailors

sid	sname	rating	age
22	Dustin	7	45
29	Brutus	1	33
31	Lubber	8	55
32	Andy	8	25
58	Rusty	10	35
64	Horatio	7	35
71	Zorba	10	16
74	Horato	9	35
85	Art	3	25.5
95	Bob	3	63.5

Reserves

sid	bid	day
22	101	10/10/98
22	102	10/10/98
22	103	10/8/98
22	104	10/7/98
31	102	11/10/98
31	103	11/6/98
31	104	11/12/98
64	101	9/5/98
64	102	9/8/98
74	103	9/8/98

Boats

bid	bname	color
101	Interlake	blue
102	Interlake	red
103	Clipper	green
104	Marine	red

Outline of this Lecture

SQL: Set Operators

- UNION/EXCEPT/INTERSECT
- duplicates in SQL

SQL: Nested Queries

- IN/EXISTS/ALL
- correlated queries

Nested Queries

A parenthesized SELECT-FROM-WHERE statement (*subquery*) can be used as a value in a:

- **FROM** clause
- **WHERE** clause

Nested Query – Example

Find the names of sailors who have reserved boat 103

```
SELECT S.sname
FROM   Sailors S, Reserves R
WHERE  S.sid = R.sid
  AND  R.bid = 103;
```

Sailors

sid	sname	rating	age
22	Dustin	7	45
29	Brutus	1	33
31	Lubber	8	55
32	Andy	8	25
58	Rusty	10	35
64	Horatio	7	35
71	Zorba	10	16
74	Horato	9	35
85	Art	3	25.5
95	Bob	3	63.5

Reserves

sid	bid	day
22	101	10/10/98
22	102	10/10/98
22	103	10/8/98
22	104	10/7/98
31	102	11/10/98
31	103	11/6/98
31	104	11/12/98
64	101	9/5/98
64	102	9/8/98
74	103	9/8/98

Boats

bid	bname	color
101	Interlake	blue
102	Interlake	red
103	Clipper	green
104	Marine	red

Nested Query – Example

Find the names of sailors who have reserved boat 103

```
SELECT S.sname
FROM   Sailors S, Reserves R
WHERE  S.sid = R.sid
  AND  R.bid = 103;
```

Equivalent to:

```
SELECT S.sname
FROM Sailors S
WHERE S.sid IN (SELECT R.sid
                FROM Reserves R
                WHERE R.bid = 103 );
```

outer query

inner query

Sailors

sid	sname	rating	age
22	Dustin	7	45
29	Brutus	1	33
31	Lubber	8	55
32	Andy	8	25
58	Rusty	10	35
64	Horatio	7	35
71	Zorba	10	16
74	Horato	9	35
85	Art	3	25.5
95	Bob	3	63.5

Reserves

sid	bid	day
22	101	10/10/98
22	102	10/10/98
22	103	10/8/98
22	104	10/7/98
31	102	11/10/98
31	103	11/6/98
31	104	11/12/98
64	101	9/5/98
64	102	9/8/98
74	103	9/8/98

Boats

bid	bname	color
101	Interlake	blue
102	Interlake	red
103	Clipper	green
104	Marine	red

Nesting and Unnesting

Everything in SQL can be represented as a multiset

Hence the output of one query can be used as the input to another (nesting)

Unnesting: find an equivalent SQL query that does not use nesting

Set Comparison Operators

- IN (NOT IN)
- EXISTS (NOT EXISTS)
- op ANY
- op ALL
  - op is one of the arithmetic comparison operators (<, <=, =, <>, >=, >)

Nested Query – EXISTS

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);
```

Sailors

sid	sname	rating	age
22	Dustin	7	45
29	Brutus	1	33
31	Lubber	8	55
32	Andy	8	25
58	Rusty	10	35
64	Horatio	7	35
71	Zorba	10	16
74	Horato	9	35
85	Art	3	25.5
95	Bob	3	63.5

Reserves

sid	bid	day
22	101	10/10/98
22	102	10/10/98
22	103	10/8/98
22	104	10/7/98
31	102	11/10/98
31	103	11/6/98
31	104	11/12/98
64	101	9/5/98
64	102	9/8/98
74	103	9/8/98

Boats

bid	bname	color
101	Interlake	blue
102	Interlake	red
103	Clipper	green
104	Marine	red

Correlated Subqueries

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);
```

correlated subquery

Sailors

sid	sname	rating	age
22	Dustin	7	45
29	Brutus	1	33
31	Lubber	8	55
32	Andy	8	25
58	Rusty	10	35
64	Horatio	7	35
71	Zorba	10	16
74	Horato	9	35
85	Art	3	25.5
95	Bob	3	63.5

Reserves

sid	bid	day
22	101	10/10/98
22	102	10/10/98
22	103	10/8/98
22	104	10/7/98
31	102	11/10/98
31	103	11/6/98
31	104	11/12/98
64	101	9/5/98
64	102	9/8/98
74	103	9/8/98

Boats

bid	bname	color
101	Interlake	blue
102	Interlake	red
103	Clipper	green
104	Marine	red

A correlated subquery uses values defined in the outer query

The inner subquery gets executed multiple times!

Correlated Subqueries

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);
```

Equivalent to:

```
SELECT S.sname
FROM   (SELECT R.sid
        FROM Reserves R
        WHERE R.bid = 103) as Temp, Sailors S
WHERE  S.sid = Temp.sid
```

Sailors

sid	sname	rating	age
22	Dustin	7	45
29	Brutus	1	33
31	Lubber	8	55
32	Andy	8	25
58	Rusty	10	35
64	Horatio	7	35
71	Zorba	10	16
74	Horato	9	35
85	Art	3	25.5
95	Bob	3	63.5

Reserves

sid	bid	day
22	101	10/10/98
22	102	10/10/98
22	103	10/8/98
22	104	10/7/98
31	102	11/10/98
31	103	11/6/98
31	104	11/12/98
64	101	9/5/98
64	102	9/8/98
74	103	9/8/98

Boats

bid	bname	color
101	Interlake	blue
102	Interlake	red
103	Clipper	green
104	Marine	red

Nested Query – Exercise

Find the names of sailors who have reserved red boats but not green boats

Sailors

sid	sname	rating	age
22	Dustin	7	45
29	Brutus	1	33
31	Lubber	8	55
32	Andy	8	25
58	Rusty	10	35
64	Horatio	7	35
71	Zorba	10	16
74	Horato	9	35
85	Art	3	25.5
95	Bob	3	63.5

Reserves

sid	bid	day
22	101	10/10/98
22	102	10/10/98
22	103	10/8/98
22	104	10/7/98
31	102	11/10/98
31	103	11/6/98
31	104	11/12/98
64	101	9/5/98
64	102	9/8/98
74	103	9/8/98

Boats

bid	bname	color
101	Interlake	blue
102	Interlake	red
103	Clipper	green
104	Marine	red

```
SELECT S.sname
FROM   Sailors S, Reserves R, Boats B
WHERE  S.sid = R.sid
  AND  R.bid = B.bid
  AND  B.color = ‘red’
  AND  S.sid NOT IN (SELECT R2.sid
                     FROM   Reserves R2, Boats B2
                     WHERE  R2.bid = B2.bid
                       AND  B2.color = ‘green’);
```

Nested Query – op ANY

Find sailors whose rating is better than some sailor called Horatio

* Note that multiple sailors may be called Horatio

```
SELECT S.sname
FROM Sailors S
WHERE S.rating > ANY (SELECT S2.rating
                      FROM   Sailors S2
                      WHERE  S2.sname = ‘Horatio’);
```

Sailors

sid	sname	rating	age
22	Dustin	7	45
29	Brutus	1	33
31	Lubber	8	55
32	Andy	8	25
58	Rusty	10	35
64	Horatio	7	35
71	Zorba	10	16
74	Horato	9	35
85	Art	3	25.5
95	Bob	3	63.5

Reserves

sid	bid	day
22	101	10/10/98
22	102	10/10/98
22	103	10/8/98
22	104	10/7/98
31	102	11/10/98
31	103	11/6/98
31	104	11/12/98
64	101	9/5/98
64	102	9/8/98
74	103	9/8/98

Boats

bid	bname	color
101	Interlake	blue
102	Interlake	red
103	Clipper	green
104	Marine	red

Nested Query – op ALL

Find sailors whose rating is better than every sailor called Horatio

* Note that multiple sailors may be called Haratio

```
SELECT S.sname
FROM Sailors S
WHERE S.rating > ALL (SELECT S2.rating
                      FROM   Sailors S2
                      WHERE  S2.sname = ‘Horatio’);
```

Sailors

sid	sname	rating	age
22	Dustin	7	45
29	Brutus	1	33
31	Lubber	8	55
32	Andy	8	25
58	Rusty	10	35
64	Horatio	7	35
71	Zorba	10	16
74	Horato	9	35
85	Art	3	25.5
95	Bob	3	63.5

Reserves

sid	bid	day
22	101	10/10/98
22	102	10/10/98
22	103	10/8/98
22	104	10/7/98
31	102	11/10/98
31	103	11/6/98
31	104	11/12/98
64	101	9/5/98
64	102	9/8/98
74	103	9/8/98

Boats

bid	bname	color
101	Interlake	blue
102	Interlake	red
103	Clipper	green
104	Marine	red

Outline of this Lecture

- SQL: Set Operators
  - UNION/EXCEPT/INTERSECT
  - duplicates in SQL
- SQL: Nested Queries
  - IN/EXISTS/ALL
  - correlated queries

Jupyter Notebook

Summary

SQL: Set Operators

- UNION/EXCEPT/INTERSECT
- UNION ALL, EXCEPT ALL, INTERSECT ALL

SQL: Nested Queries

- IN/EXISTS/op ALL/op ANY
- Correlated queries