COMP 3311 DATABASE MANAGEMENT SYSTEMS

LECTURE & EXERCISES
STRUCTURED QUERY LANGUAGE (SQL)

EXAMPLE RELATIONAL SCHEMA AND DATABASE

Sailor(sailorld, sName, rating, age)

Boat(boatId, bName, color)

Reserves(sailorld, boatld, rDate)

Attribute names in italics are foreign key attributes.

Sailor

<u>sailorId</u>	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	Horatio	9	35
85	Art	3	25
95	Bob	3	63
99	Chris	10	30

11 tuples

Reserves

<u>sailorld</u>	<u>boatld</u>	<u>rDate</u>
22	101	10/10/17
22	102	10/10/17
22	103	08/10/17
22	104	07/10/17
31	102	10/11/17
31	103	06/11/17
31	104	12/11/17
64	101	05/09/17
64	102	08/09/17
74	103	08/09/17
99	104	08/08/17

Boat

<u>boatld</u>	bName	color
101	Interlake	blue
102	Interlake	red
103	Clipper	green
104	Marine	red
105	Serenity	Cyan

5 tuples

2

11 tuples



Find the name and the number of reservations for each red boat.

(Interlake, 3), (Marine, 3)

Is this a correct solution?

select bName, count(*) reservationCount
from Boat natural join Reserves
where color='red'
group by Boat.boatId;

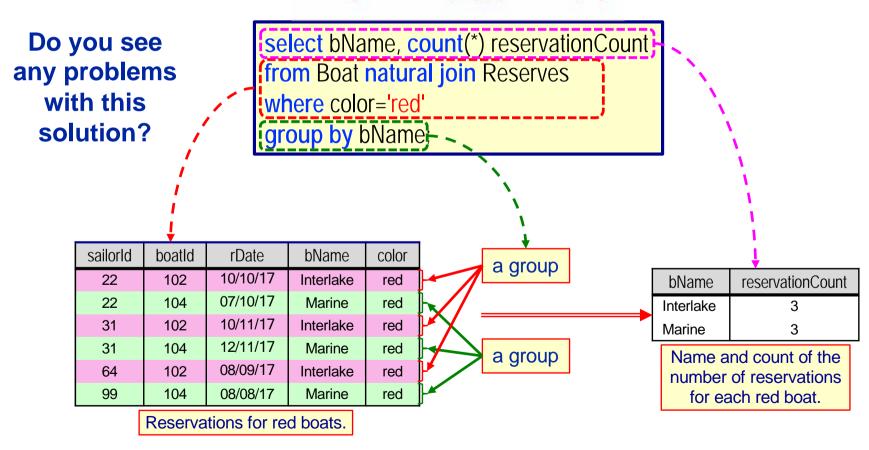
Illegal!!! Why?

All non-aggregate attributes in the select clause must appear in the group by clause (i.e., bName must appear in the group by clause).

EXERCISE I (CONTD)

Find the name and the number of reservations for each red boat.

(Interlake, 3), (Marine, 3)



Boat(boatld, bName, color)

COMP 3311

EXERCISE I (CONTD)

Suppose we change the query to this.

Find the name and the number of reservations for each boat.

What is the result?

select bName, count(*) reservationCount from Boat natural join Reserves group by bName

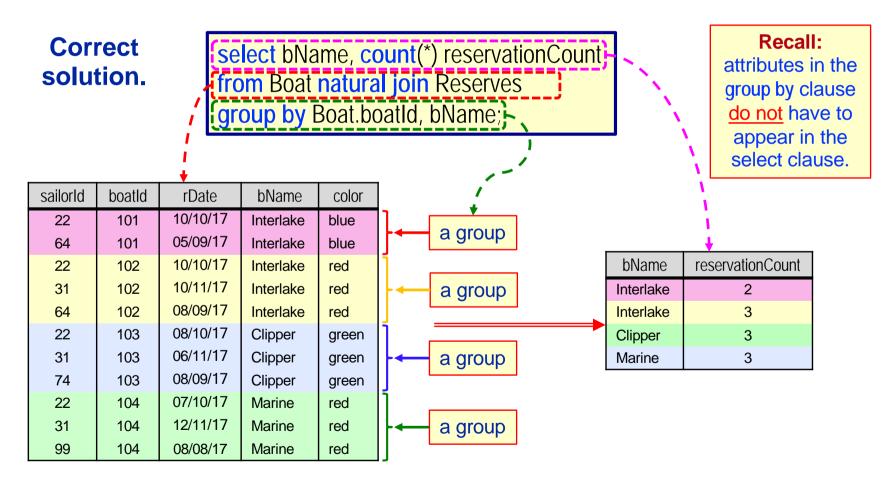
sailorld	boatld	rDate	bName	color	\
22	101	10/10/17	Interlake	blue	1 1
64	101	05/09/17	Interlake	blue	· •
22	102	10/10/17	Interlake	red	a group
31	102	10/11/17	Interlake	red	
64	102	08/09/17	Interlake	red	J =
22	103	08/10/17	Clipper	green]
31	103	06/11/17	Clipper	green	→ a group
74	103	08/09/17	Clipper	green	
22	104	07/10/17	Marine	red	j
31	104	12/11/17	Marine	red	→ a group
ga	104	08/08/17	Marine	red	

bName reservationCount
Interlake 5
Clipper 3
Marine 3

Since bName is not unique, grouping on it can get an incorrect result!

EXERCISE I (CONTD)

Find the name and the number of reservations for each boat.



Find the sailor id and number of reservations made for each sailor.

(22, 4), (29, 0), (31, 3), (32, 0), (58, 0), (64, 2), (71, 0), (74, 1), (85, 0), (95, 0), (99, 1)

select sailorId, count(sailorId) reservationCount
from Reserves
group by sailorId;

How to include all sailors?

sailorid	reservationCount
22	4
31	3
64	2
74	1
99	1

How about joining Sailor and Reserves?

select sailorId, count(sailorId) reservationCount
from Sailor natural join Reserves
group by sailorId;

What's the problem?

Salloriu	reservationCount
22	4
31	3
64	2
74	1
99	1

asilarid rasamustian Count

EXERCISE 2 (CONTD)

Find the sailor id and number of reservations made for each sailor.

(22, 4), (29, 0), (31, 3), (32, 0), (58, 0), (64, 2), (71, 0), (74, 1), (85, 0), (95, 0), (99, 1)

sailorld	sName	rating	age	boatld	rDate
22	Dustin	7	45	101	10/10/17
22	Dustin	7	45	102	10/10/17
22	Dustin	7	45	103	08/10/17
22	Dustin	7	45	104	07/10/17
31	Lubber	8	55	102	10/11/17
31	Lubber	8	55	103	06/11/17
31	Lubber	8	55	104	12/11/17
64	Horatio	7	35	101	05/09/17
64	Horatio	7	35	102	08/09/17
74	Horatio	9	35	103	08/09/17
99	Chris	10	30	104	08/08/17
29	Brutus	1	33	-	-
32	Andy	8	25	-	-
58	Rusty	10	35	-	-
71	Zorba	10	16	-	-
85	Art	3	25	-	-
95	Bob	3	63	-	-

select sailorId, count(sailorId) reservationCount
from Sailor natural join Reserves
group by sailorId;

sailorld	reservationCount
22	4
31	3
64	2
74	1
99	1

Some Sailor tuples have no match in the Reserves relation.

How to deal with this problem?



EXERCISE 2 (CONTD)

Find the sailor id and number of reservations made for each sailor.

(22, 4), (29, 0), (31, 3), (32, 0), (58, 0), (64, 2), (71, 0), (74, 1), (85, 0), (95, 0), (99, 1)

select sailorId, count(boatId) reservationCount
from Sailor natural left outer join Reserves
group by sailorId;

Note: left outer join keeps the common attributes, while natural left outer join removes the common attributes.

Is this a correct solution?

NO! Why?

select sailorId, count(sailorId) reservationCount
from Sailor natural left outer join Reserves
group by sailorId;

Counting is done on the sailor ids and all of them appear at least once in the result.

Find the records (tuples) of the sailors with the highest rating.

(58, Rusty, 10, 35), (71, Zorba, 10, 16), (99, Chris, 10, 30)

Is this a correct solution?

NO! Why?

select *
from Sailor
where rating=max(rating);

There is no max(rating) value to compare in the where clause.

The max rating value has to be obtained by a select statement!

Is this a correct solution?

NO! Why?

select *, max(rating)
from Sailor;

A query that returns multiple tuples cannot contain an aggregate function.

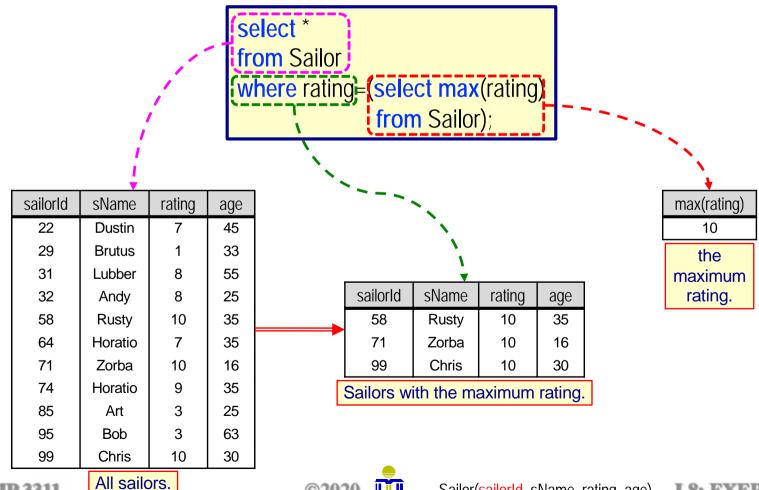
There are multiple tuples in the result, but only one max value!



EXERCISE 3 (CONTD)

Find the records (tuples) of the sailors with the highest rating.

(58, Rusty, 10, 35), (71, Zorba, 10, 16), (99, Chris, 10, 30)

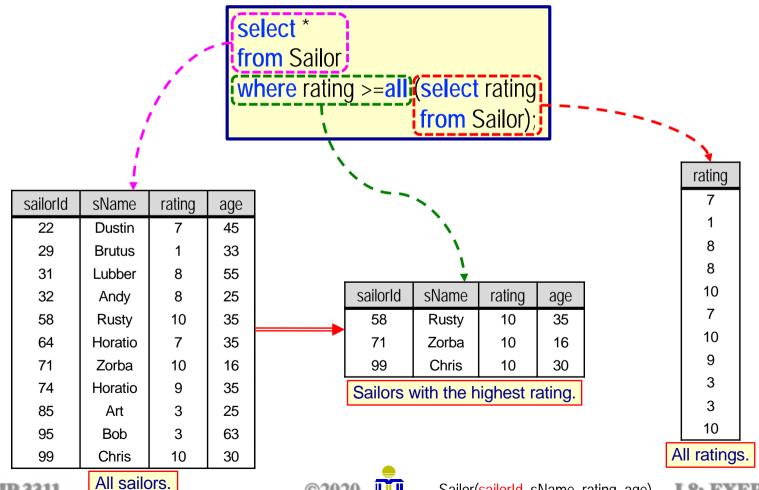




EXERCISE 3 (CONTD)

Find the records (tuples) of the sailors with the highest rating.

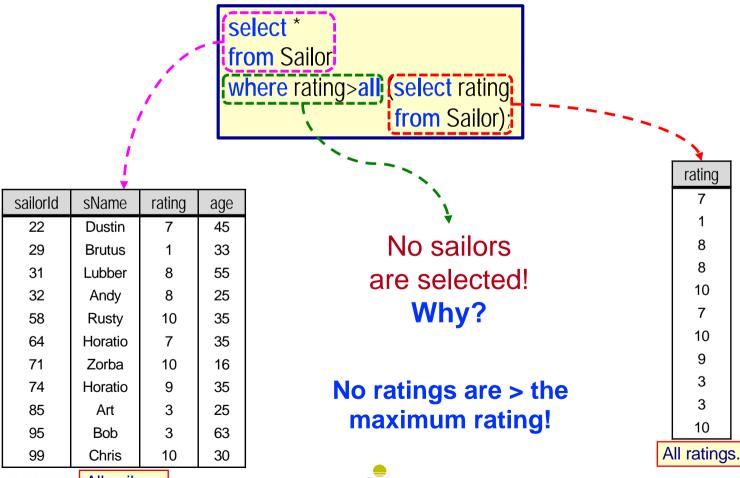
(58, Rusty, 10, 35), (71, Zorba, 10, 16), (99, Chris, 10, 30)



L8: EXERCISES

EXERCISE 3 (CONTO)

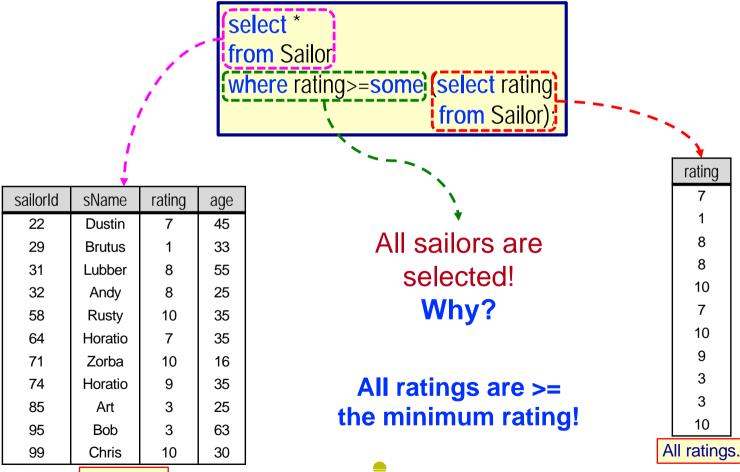
What is the result if we replace ">=all" with ">all"?



COMP 3311

EXERCISE 3 (CONTO)

What is the result if we replace ">=all" with ">some"?



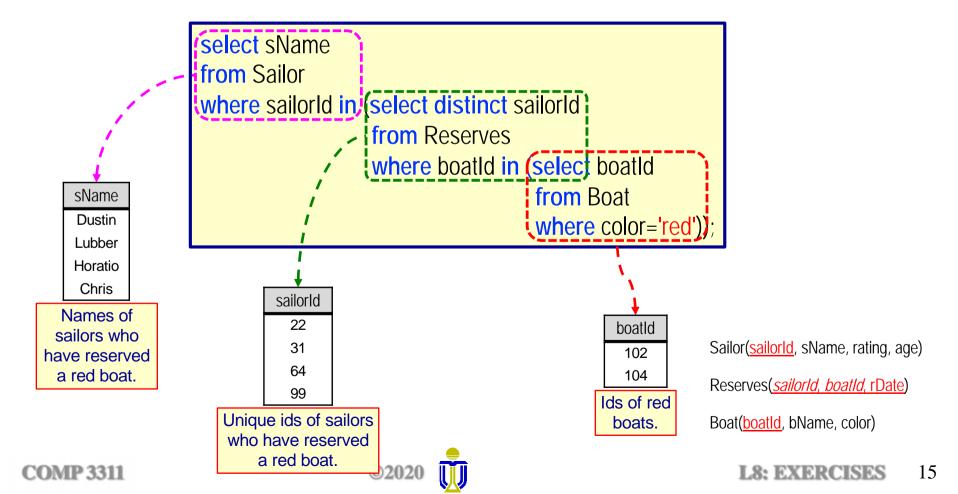
COMP 3311

DO NOT use JOIN

Find the names of sailors who have reserved a red boat.

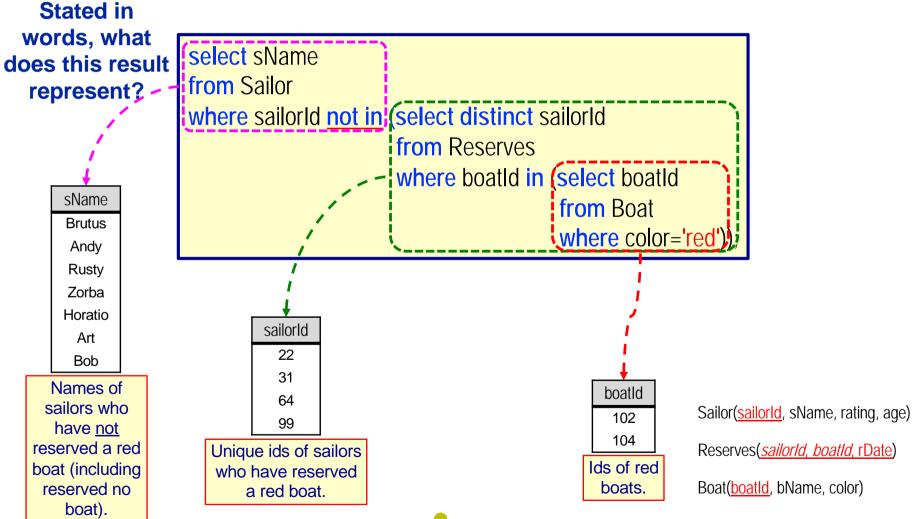
Use only set membership

Dustin, Lubber, Horatio, Chris



EXERCISE 4 (CONTO)

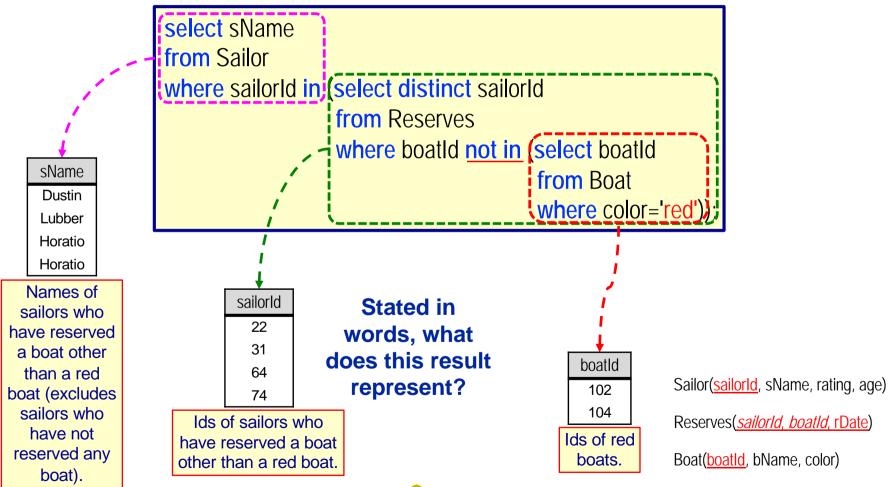
What if we replace the first in with not in?



COMP 3311

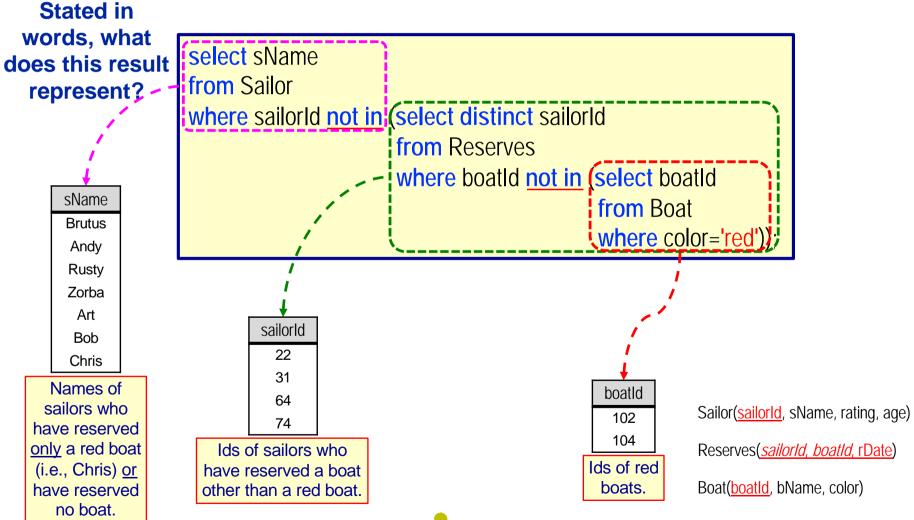
EXERCISE 4 (CONTO)

What if we replace the second in with not in?



EXERCISE 4 (CONTO)

What if we replace both in's with not in?



Find the ratings and the average age of the ratings where a rating's average age is equal to the minimum average age of all ratings.

(10, 27)

Is this a correct solution?

NO! Why?

```
select rating
from Sailor
where avg(age)=min(select avg(age))
from Sailor
group by rating);
```

Cannot use "where avg(age)=" since avg(age) is not an attribute of Sailor!

Cannot use "min(...".
Illegal SQL!

Is this a correct solution?

NO! Why?

```
select rating
from Sailor
group by rating

(having age=(select avg(age))
from Sailor
group by rating);

avgAge
33
44
40
40
35
27
```

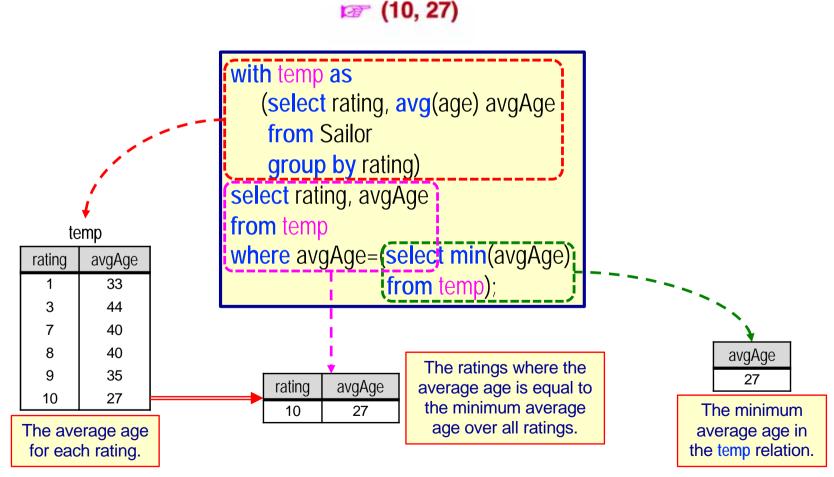
Cannot use "having age=" since age is not in the select or group by clauses.

Illegal SQL!

Subquery returns multiple values.

EXERCISE 5 (CONTD)

Find the ratings and the average age of the ratings where a rating's average age is equal to the minimum average age of all ratings.



EXERCISE 5 (CONTD)

Find the ratings and the average age of the ratings where a rating's average age is equal to the minimum average age of all ratings.

(10, 27)

- This query is correct SQL, but will not execute in Oracle.
 - Returns the error "table or view does not exist".

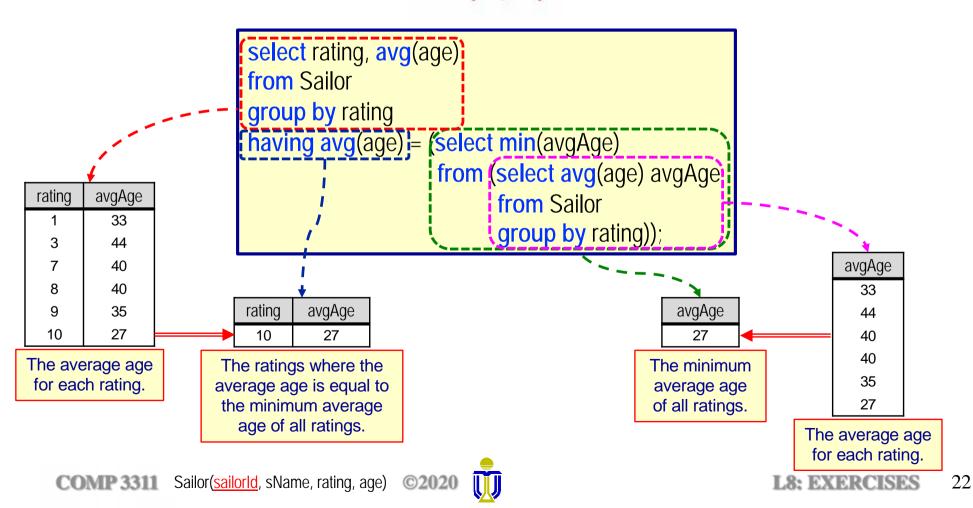
Oracle restricts the scope of the alias temp to the outer select.



EXERCISE 5 (CONTD)

Find the ratings and the average age of the ratings where a rating's average age is equal to the minimum average age of all ratings.

(10, 27)



Find the boat name and number of reservations for each boat.

(Clipper, 3), (Interlake, 2), (Interlake, 3), (Marine, 3), (Serenity, 0)

select bName, count(bName) reservationCount from Boat natural left outer join Reserves group by bName;

The count for Serenity is incorrect. Interlake should have two separate counts.

bName	reservation Count		
Clipper	3		
Interlake	5		
Marine	3		
Serenity	1		

What's the problem?

How about group on boatld, bName; count boatld?

select bName, count(boatId) reservationCount from Boat natural left outer join Reserves group by boatld, bName;

The count for Serenity is still incorrect!

bName	reservation Count
Clipper	3
Interlake	2
Interlake	3
Marine	3
Serenity	1

What's the problem?



Find the boat name and number of reservations for each boat.

(Clipper, 3), (Interlake, 2), (Interlake, 3), (Marine, 3), (Serenity, 0)

select bName, count(boatld) reservationCount from Boat natural left outer join Reserves group by boatld, bName;

	Boat natural left outer join Reserves			
boatld	bName	color	sailorld	rDate
101	Interlake	blue	64	05/09/17
101	Interlake	blue	22	10/10/17
102	Interlake	red	22	10/10/17
102	Interlake	red	64	08/09/17
102	Interlake	red	31	10/11/17
103	Clipper	green	22	08/10/17
103	Clipper	green	31	06/11/17
103	Clipper	green	74	08/09/17
104	Marine	red	22	07/10/17
104	Marine	red	99	08/08/17
104	Marine	red	31	12/11/17
105	Serenity	cyan	(null)	(null)

We need to count sailorld or rDate!





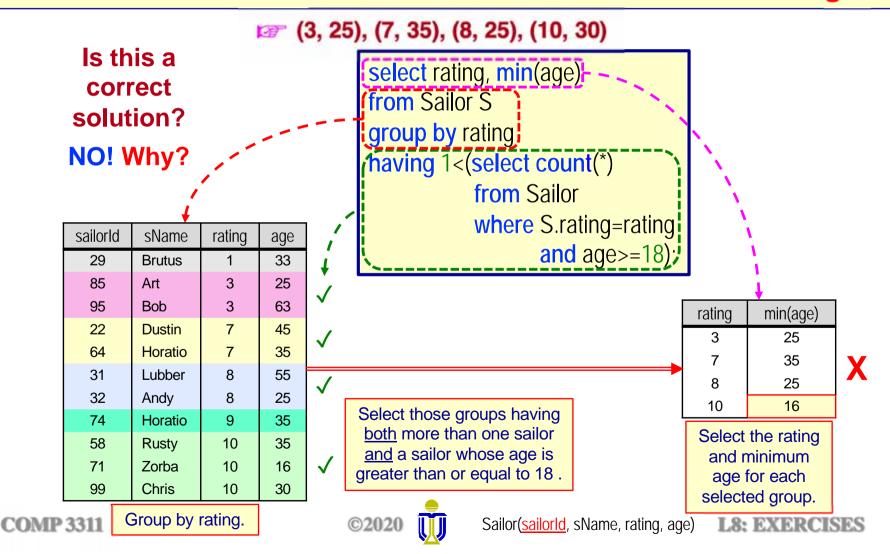
Find the boat name and number of reservations for each boat.

(Clipper, 3), (Interlake, 2), (Interlake, 3), (Marine, 3), (Serenity, 0)

select bName, count(sailorId) reservationCount from Boat natural left outer join Reserves group by boatld, bName;

	V
bName	reservation Count
Clipper	3
Interlake	2
Interlake	3
Marine	3
Serenity	0

Find the age of the youngest adult sailor (i.e., age≥18) for each rating for which there are at least 2 adult sailors with the same rating.



EXERCISE 7 (CONTD)

Find the age of the youngest adult sailor (i.e., age≥18) for each rating for which there are at least 2 <u>adult</u> sailors with the same rating.

