

DSAA 5012

Advanced Data Management for Data Science

LECTURE 6 EXERCISES

STRUCTURED QUERY LANGUAGE (SQL)

EXAMPLE RELATIONAL SCHEMA AND DATABASE

Sailor(sailorId, sName, rating, age)

Boat(boatId, bName, color)

Reserves(sailorId, boatId, 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>sailorId</u>	<u>boatId</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

11 tuples

Boat

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

5 tuples

EXERCISE 1

Find the names of sailors who have reserved boat 103.

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>sailorId</u>	<u>boatId</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

11 tuples

Boat

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

5 tuples

EXERCISE 1

Find the names of sailors who have reserved boat 103.

☞ **Dustin, Lubber, Horatio**

How to eliminate duplicate columns in the join result?

```
select sName
from Sailor, Reserves
where Sailor.sailorId=Reserves.sailorId
and boatId=103;
```

sName
Dustin
Lubber
Horatio

Project on sName.

sailorId	sName	rating	age	sailorId1	boatId	rDate
22	Dustin	7	45	22	101	10/10/17
22	Dustin	7	45	22	102	10/10/17
22	Dustin	7	45	22	103	08/10/17
22	Dustin	7	45	22	104	07/10/17
31	Lubber	8	55	31	102	10/11/17
31	Lubber	8	55	31	103	06/11/17
31	Lubber	8	55	31	104	12/11/17
64	Horatio	7	35	64	101	05/09/17
64	Horatio	7	35	64	102	08/09/17
74	Horatio	9	35	74	103	08/09/17
99	Chris	10	30	99	104	08/08/17

sailorId	sName	rating	age	sailorId1	boatId	rDate
22	Dustin	7	45	22	103	08/10/17
31	Lubber	8	55	31	103	06/11/17
74	Horatio	9	35	74	103	08/09/17

Keep only those tuples where the boatId is 103.

Join Sailor and Reserves on sailorId.



EXERCISE 1 (cont'd)

Find the names of sailors who have reserved boat 103.

☞ **Dustin, Lubber, Horatio**

Natural join eliminates duplicate columns in the join result.

```
select sName
from Sailor natural join Reserves
where boatId=103
```

sName
Dustin
Lubber
Horatio

Project on sName.

sailorId	sName	rating	age	boatId	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

sailorId	sName	rating	age	boatId	rDate
22	Dustin	7	45	103	08/10/17
31	Lubber	8	55	103	06/11/17
74	Horatio	9	35	103	08/09/17

Keep only those tuples where the boatId is 103.

Join Sailor and Reserves on sailorId.



EXERCISE 2

Find the ids and names of sailors who have reserved either a red or a green boat.

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>sailorId</u>	<u>boatId</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

11 tuples

Boat

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

5 tuples

EXERCISE 2

Find the ids and names of sailors who have reserved either a red or a green boat.

👉 (22, Dustin), (31, Lubber), (64, Horatio), (74, Horatio), (99, Chris)

```
select distinct Sailor.sailorId, sName
from Sailor, Reserves, Boat
where Sailor.sailorId=Reserves.sailorId
and Reserves.boatId=Boat.boatId
and (color='red' or color='green');
```

Sailor(sailorId, sName, rating, age)

Reserves(sailorId, boatId, rDate)

Boat(boatId, bName, color)

sailorId	sName	rating	age	sailorId1	boatId	rDate	boatId1	bName	color
22	Dustin	7	45	22	101	10/10/17	101	Interlake	blue
22	Dustin	7	45	22	102	10/10/17	102	Interlake	red
22	Dustin	7	45	22	103	08/10/17	103	Clipper	green
22	Dustin	7	45	22	104	07/10/17	104	Marine	red
31	Lubber	8	55	31	102	10/11/17	102	Interlake	red
31	Lubber	8	55	31	103	06/11/17	103	Clipper	green
31	Lubber	8	55	31	104	12/11/17	104	Marine	red
64	Horatio	7	35	64	101	05/09/17	101	Interlake	blue
64	Horatio	7	35	64	102	08/09/17	102	Interlake	red
74	Horatio	9	35	74	103	08/09/17	103	Clipper	green
99	Chris	10	30	99	104	08/08/17	104	Marine	red

Join Sailor and Reserves on sailorId and Reserves and Boat on boatId.

Keep only those tuples where the boat color is red or green.

EXERCISE 2 (cont'd)

Find the ids and names of sailors who have reserved either a red or a green boat.

☞ (22, Dustin), (31, Lubber), (64, Horatio), (74, Horatio), (99, Chris)

Sailor(sailorId, sName, rating, age)

Reserves(sailorId, boatId, rDate)

Boat(boatId, bName, color)

```
select distinct Sailor.sailorId, sName
from Sailor, Reserves, Boat
where Sailor.sailorId=Reserves.sailorId
and Reserves.boatId=Boat.boatId
and (color='red' or color='green');
```

Keep only unique tuples.

sailorId	sName
22	Dustin
31	Lubber
64	Horatio
74	Horatio
99	Chris

sailorId	sName	rating	age	sailorId1	boatId	rDate	boatId1	bName	color
22	Dustin	7	45	22	102	10/10/17	102	Interlake	red
22	Dustin	7	45	22	103	08/10/17	103	Clipper	green
22	Dustin	7	45	22	104	07/10/17	104	Marine	red
31	Lubber	8	55	31	102	10/11/17	102	Interlake	red
31	Lubber	8	55	31	103	06/11/17	103	Clipper	green
31	Lubber	8	55	31	104	12/11/17	104	Marine	red
64	Horatio	7	35	64	102	08/09/17	102	Interlake	red
74	Horatio	9	35	74	103	08/09/17	103	Clipper	green
99	Chris	10	30	99	104	08/08/17	104	Marine	red

sailorId	sName
22	Dustin
22	Dustin
22	Dustin
31	Lubber
31	Lubber
31	Lubber
64	Horatio
74	Horatio
99	Chris

Project on sailorId and sName.

EXERCISE 2 (cont'd)

Find the ids and names of sailors who have reserved either a red or a green boat.

☞ (22, Dustin), (31, Lubber), (64, Horatio), (74, Horatio), (99, Chris)

```
select distinct Sailor.sailorId, sName
from Sailor, Reserves, Boat
where Sailor.sailorId=Reserves.sailorId
and Reserves.boatId=Boat.boatId
and (color='red' and color='green');
```

Sailor(sailorId, sName, rating, age)

Reserves(sailorId, boatId, rDate)

Boat(boatId, bName, color)

Why is it necessary to qualify `sailorId` in the `select` clause?

☞ `sailorId` is ambiguous in the join result.
Should we take it from `Sailor` or `Reserves`?
(For some operations it will make a difference!)

What do we get if we replace `or` with `and` in the query?

☞ No result since there is no boat whose color is both red and green!

EXERCISE 2 (cont'd)

Find the ids and names of sailors who have reserved either a red or a green boat.

☞ (22, Dustin), (31, Lubber), (64, Horatio), (74, Horatio), (99, Chris)

```
select distinct sailorId, sName
from Sailor natural join Reserves natural join Boat
where color='red' or color='green'
```

Keep only unique tuples.

sailorId	sName
22	Dustin
31	Lubber
64	Horatio
74	Horatio
99	Chris

Use natural join to eliminate duplicate columns in the join result.

sailorId	sName	rating	age	boatId	rDate	bName	color
22	Dustin	7	45	102	10/10/17	Interlake	red
22	Dustin	7	45	103	08/10/17	Clipper	green
22	Dustin	7	45	104	07/10/17	Marine	red
31	Lubber	8	55	102	10/11/17	Interlake	red
31	Lubber	8	55	103	06/11/17	Clipper	green
31	Lubber	8	55	104	12/11/17	Marine	red
64	Horatio	7	35	102	08/09/17	Interlake	red
74	Horatio	9	35	103	08/09/17	Clipper	green
99	Chris	10	30	104	08/08/17	Marine	red

Keep only those tuples where the boat color is red or green.

sailorId	sName
22	Dustin
22	Dustin
22	Dustin
31	Lubber
31	Lubber
31	Lubber
64	Horatio
74	Horatio
99	Chris

Project on sailorId and sName.

EXERCISE 3

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

Use intersect

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>sailorId</u>	<u>boatId</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

11 tuples

Boat

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

5 tuples



EXERCISE 3

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

Use intersect

 Dustin, Lubber

```
select sName
from (select sailorId, sName
      from Sailor natural join Reserves natural join Boat
      where color='red'
      intersect
      select sailorId, sName
      from Sailor natural join Reserves natural join Boat
      where color='green');
```

Sailors who have reserved red boats.

Sailors who have reserved both a red and a green boat.

Sailors who have reserved green boats.

Sailor(sailorId, sName, rating, age)

Reserves(sailorId, boatId, rDate)

Boat(boatId, bName, color)



EXERCISE 3 (cont'd)

Sailor(sailorId, sName, rating, age)

Reserves(sailorId, boatId, rDate)

Boat(boatId, bName, color)

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

Use intersect

 **Dustin, Lubber**

from Sailor, Reserves, Boat **where** Sailor.sailorId=Reserves.sailorId **and** Reserves.boatId=Boat.boatId **and** color='red'

sailorId	sName	rating	age	boatId	rDate	bName	color
22	Dustin	7	45	102	10/10/17	Interlake	red
22	Dustin	7	45	104	07/10/17	Marine	red
31	Lubber	8	55	102	10/11/17	Interlake	red
31	Lubber	8	55	104	12/11/17	Marine	red
64	Horatio	7	35	102	08/09/17	Interlake	red
99	Chris	10	30	104	08/08/17	Marine	red

Sailors who have reserved red boats.

select Sailor.sailorId, sName

sailorId	sName
22	Dustin
22	Dustin
31	Lubber
31	Lubber
64	Horatio
99	Chris

Why are there no duplicates in the result?

from Sailor, Reserves, Boat **where** Sailor.sailorId=Reserves.sailorId **and** Reserves.boatId=Boat.boatId **and** color='green'

sailorId	sName	rating	age	boatId	rDate	bName	color
22	Dustin	7	45	103	08/10/17	Clipper	green
31	Lubber	8	55	103	06/11/17	Clipper	green
74	Horatio	9	35	103	08/09/17	Clipper	green

Sailors who have reserved green boats.

select Sailor.sailorId, sName

sailorId	sName
22	Dustin
31	Lubber
74	Horatio

∩

select sName

sName
Dustin
Lubber



EXERCISE 3 (cont'd)

What happens if we remove sailorId from the two inner select clauses?

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

Use intersect

 Dustin, Lubber

```
select sName
from (select sName
      from Sailor natural join Reserves natural join Boat
      where color='red'
      intersect
      select sName
      from Sailor natural join Reserves natural join Boat
      where color='green');
```

Sailors who have reserved red boats.

Sailors who have reserved both a red and a green boat.

Sailors who have reserved green boats.

Sailor(sailorId, sName, rating, age)

Reserves(sailorId, boatId, rDate)

Boat(boatId, bName, color)

EXERCISE 3 (cont'd)

What happens if we remove sailorId from the inner select clauses?

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

Use intersect

☞ Dustin, Lubber

from Sailor, Reserves, Boat **where** Sailor.sailorId=Reserves.sailorId **and** Reserves.boatId=Boat.boatId **and** color='red'

sailorId	sName	rating	age	boatId	rDate	bName	color
22	Dustin	7	45	102	10/10/17	Interlake	red
22	Dustin	7	45	104	07/10/17	Marine	red
31	Lubber	8	55	102	10/11/17	Interlake	red
31	Lubber	8	55	104	12/11/17	Marine	red
64	Horatio	7	35	102	08/09/17	Interlake	red
99	Chris	10	30	104	08/08/17	Marine	red

Sailors who have reserved red boats.

from Sailor, Reserves, Boat **where** Sailor.sailorId=Reserves.sailorId **and** Reserves.boatId=Boat.boatId **and** color='green'

sailorId	sName	rating	age	boatId	rDate	bName	color
22	Dustin	7	45	103	08/10/17	Clipper	green
31	Lubber	8	55	103	06/11/17	Clipper	green
74	Horatio	9	35	103	08/09/17	Clipper	green

Sailors who have reserved green boats.

select sName

sName
Dustin
Dustin
Lubber
Lubber
Horatio
Chris

What is the problem?

☞ sName is not unique!

∩

select sName

sName
Dustin
Lubber
Horatio

select sName

sName
Dustin
Lubber
Horatio

X

Sailor(sailorId, sName, rating, age)

Reserves(sailorId, boatId, rDate)

Boat(boatId, bName, color)



EXERCISE 3 (cont'd)

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

Use Join

👉 Dustin, Lubber

👉 **Hint:** You need to use correlation names.

```
select distinct sName
from Sailor S, Reserves R1, Boat B1, Reserves R2, Boat B2
where S.sailorId=R1.sailorId
and R1.boatId=B1.boatId
and B1.color='red'
and S.sailorId=R2.sailorId
and R2.boatId=B2.boatId
and B2.color='green';
```

The same sailor id's have to be in Sailor and in both join results.

Join Reserves and Boat
where color='red'.

Join Reserves and Boat
where color='green'.

Sailor(sailorId, sName, rating, age)

Reserves(sailorId, boatId, rDate)

Boat(boatId, bName, color)

EXERCISE 3 (cont'd)

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

Use Join

 **Dustin, Lubber**

Only 22 and 31 are in both join results and in Sailor.

Sailor			
sailorId	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

Result of join Reserves and Boat where color='red'.

R1.boatId=B1.boatId and B1.color='red'				
sailorId	boatId	rDate	bName	color
22	102	10/10/17	Interlake	red
22	104	07/10/17	Marine	red
31	102	10/11/17	Interlake	red
31	104	12/11/17	Marine	red
64	102	08/09/17	Interlake	red
99	104	08/08/17	Marine	red

Note
Duplicate columns are not shown in the join result.

JOIN_{sailorId}

R2.boatId=B2.boatId and B2.color='green'				
sailorId	boatId	rDate	bName	color
22	103	08/10/17	Clipper	green
31	103	06/11/17	Clipper	green
74	103	08/09/17	Clipper	green

select distinct sName

sName
Dustin
Lubber

Result of join Reserves and Boat where color='green'.

Sailor(sailorId, sName, rating, age)

Reserves(sailorId, boatId, rDate)

Boat(boatId, bName, color)



EXERCISES 4, 5, 6

Sailor(sailorId, sName, rating, age)

Boat(boatId, bName, color)

Reserves(sailorId, boatId, rDate)

Exercise 4: Find the ids and names of boats that have never been reserved.

Exercise 5: Find the ids and names of sailors who have not reserved boat 103.

Exercise 6: Find the names and ids of those sailors who have the same name.



EXERCISE 4

Find the ids and names of boats that have never been reserved.

👉 (105, Serenity)

```

select boatId, bName
from Boat
minus
select boatId, bName
from Boat natural join Reserves;
    
```

boatId	bName

boatId	bName
105	Serenity

boatId	rDate	bName	color
101	10/10/17	Interlake	blue
102	10/10/17	Interlake	red
103	08/10/17	Clipper	green
104	07/10/17	Marine	red
102	10/11/17	Interlake	red
103	06/11/17	Clipper	green
104	12/11/17	Marine	red
101	05/09/17	Interlake	blue
102	08/09/17	Interlake	red
103	08/09/17	Clipper	green
104	08/08/17	Marine	red

Join Boat and Reserves on boatId.

Project on boatId and bName.

boatId	bName
101	Interlake
102	Interlake
103	Clipper
104	Marine
102	Interlake
103	Clipper
104	Marine
101	Interlake
102	Interlake
103	Clipper
104	Marine



EXERCISE 4 (cont'd)

Find the ids and names of boats that have never been reserved.

👉 (105, Serenity)

Is this a correct solution?

Yes!

```
select Boat.boatId, bName
from Boat left outer join Reserves
on Boat.boatId=Reserves.boatId
where Reserves.boatId is null;
```

left outer join

Boat

boatId	bName	color
101	Interlake	blue
102	Interlake	red
103	Clipper	green
104	Marine	red
105	Serenity	Cyan

Reserves

sailorId	boatId	rDate
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 -----> <----- Reserves ----->

from Boat left outer join Reserves on Boat.boatId=Reserves.boatId

boatId	bName	color	sailorId	boatId	rDate
101	Interlake	blue	64	101	05/09/17
101	Interlake	blue	22	101	10/10/17
102	Interlake	red	22	102	10/10/17
102	Interlake	red	64	102	08/09/17
102	Interlake	red	31	102	10/11/17
103	Clipper	green	22	103	08/10/17
103	Clipper	green	31	103	06/11/17
103	Clipper	green	74	103	08/09/17
104	Marine	red	22	104	07/10/17
104	Marine	red	99	104	08/08/17
104	Marine	red	31	104	12/11/17
105	Serenity	cyan	(null)	(null)	(null)



EXERCISE 5

Find the ids and names of sailors who have not reserved boat 103.

👉 (29, Brutus), (32, Andy), (58, Rusty), (64, Horatio),
(71, Zorba), (85, Art), (95, Bob), (99, Chris)

Is this a
correct
solution?
No! Why?

```
select distinct Sailor.sailorId, sName
from Sailor, Reserves
where Sailor.sailorId=Reserves.sailorId
and boatId <> 103;
```

Does not include
sailors who have
not reserved any
boat (i.e., sailors
who do not appear
in Reserves).

sailorId	sName	rating	age	boatId	rDate
22	Dustin	7	45	101	10/10/17
22	Dustin	7	45	102	10/10/17
22	Dustin	7	45	104	07/10/17
31	Lubber	8	55	102	10/11/17
31	Lubber	8	55	104	12/11/17
64	Horatio	7	35	101	05/09/17
64	Horatio	7	35	102	08/09/17
99	Chris	10	30	104	08/08/17

sailorId	sName
22	Dustin
31	Lubber
64	Horatio
99	Chris

X

EXERCISE 5 (cont'd)

Find the ids and names of sailors who have not reserved boat 103.

👉 (29, Brutus), (32, Andy), (58, Rusty), (64, Horatio),
(71, Zorba), (85, Art), (95, Bob), (99, Chris)

sailorId	sName
22	Dustin
29	Brutus
31	Lubber
32	Andy
58	Rusty
64	Horatio
71	Zorba
74	Horatio
85	Art
95	Bob
99	Chris

All unique combinations of sailorId and sName.

```
select sailorId, sName
from Sailor
minus
select Sailor.sailorId, sName
from Sailor, Reserves
where Sailor.sailorId=Reserves.sailorId
and boatId=103;
```

sailorId	sName
22	Dustin
31	Lubber
74	Horatio

Sailors who have reserved boat 103.

sailorId	sName
29	Brutus
32	Andy
58	Rusty
64	Horatio
71	Zorba
85	Art
95	Bob
99	Chris

Sailors who have not reserved boat 103.

Sailor(sailorId, sName, rating, age)

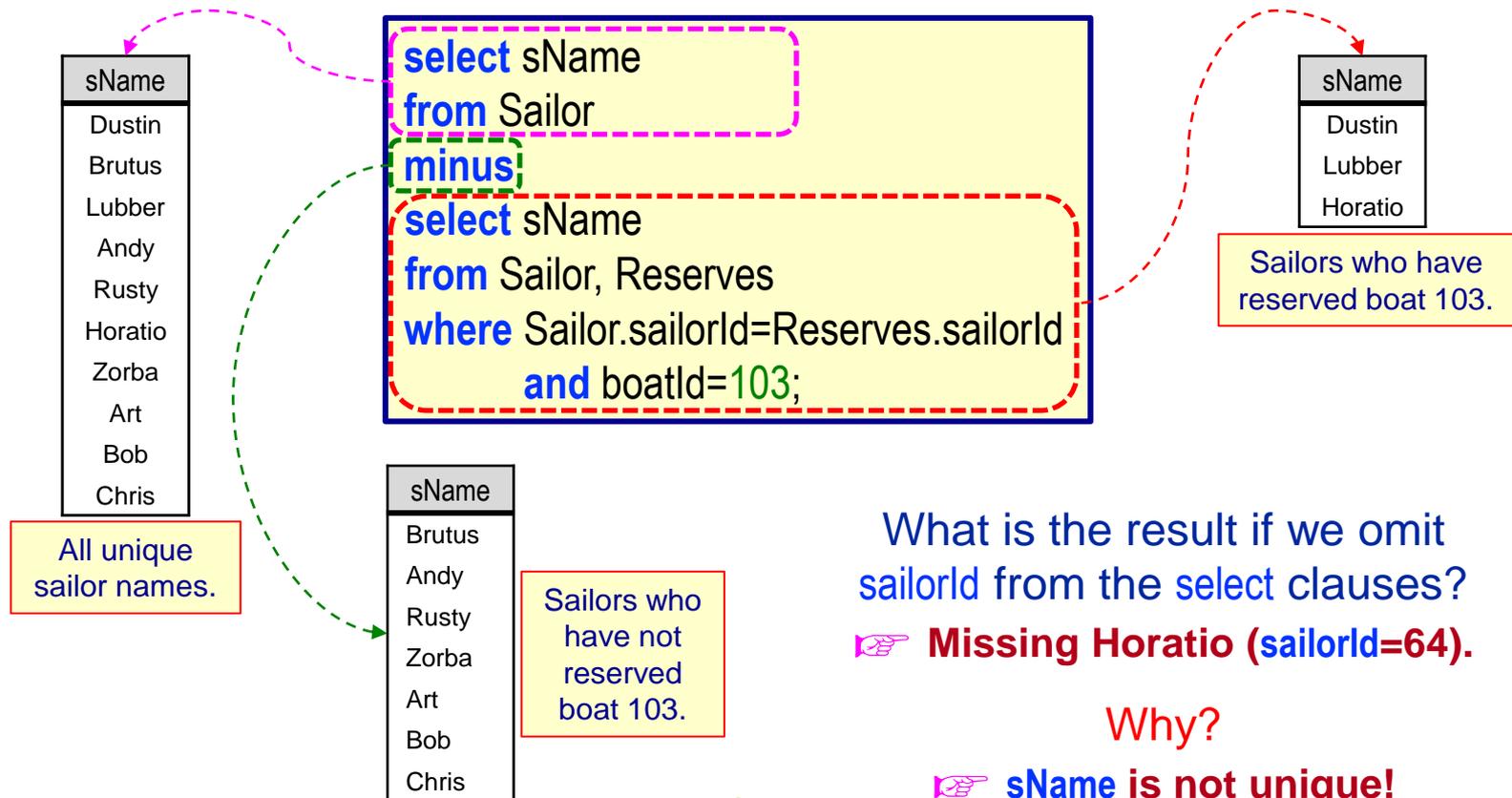
Reserves(sailorId, boatId, rDate)



EXERCISE 5 (cont'd)

Find the ids and names of sailors who have not reserved boat 103.

☞ (29, Brutus), (32, Andy), (58, Rusty), (64, Horatio),
(71, Zorba), (85, Art), (95, Bob), (99, Chris)



EXERCISE 6

Find the names and ids of those sailors who have the same name.

☞ (Horatio, 64), (Horatio, 74)

Join Sailor
with itself.

```
select S1.sName, S1.sailorId  
from Sailor S1, Sailor S2  
where S1.sName=S2.sName  
and S1.sailorId<>S2.sailorId;
```

S1.sailorId	S1.sName	S1.rating	S1.age	S2.sailorId	S2.sName	S2.rating	S2.age
22	Dustin	7	45	22	Dustin	7	45
29	Brutus	1	33	29	Brutus	1	33
31	Lubber	8	55	31	Lubber	8	55
32	Andy	8	25	32	Andy	8	25
58	Rusty	10	35	58	Rusty	10	35
64	Horatio	7	35	64	Horatio	7	35
64	Horatio	7	35	74	Horatio	9	35
71	Zorba	10	16	71	Zorba	10	16
74	Horatio	9	35	74	Horatio	9	35
74	Horatio	9	35	64	Horatio	7	35
85	Art	3	25	85	Art	3	25
95	Bob	3	63	95	Bob	3	63
99	Chris	10	30	99	Chris	10	30

Keep only
those tuples
where the
sailor names
are the same.

EXERCISE 6 (cont'd)

Find the names and ids of those sailors who have the same name.

☞ (Horatio, 64), (Horatio, 74)

```
select S1.sName, S1.sailorId
from Sailor S1, Sailor S2
where S1.sName=S2.sName
and S1.sailorId<>S2.sailorId;
```

S1.sailorId	S1.sName	S1.rating	S1.age	S2.sailorId	S2.sName	S2.rating	S2.age
64	Horatio	7	35	74	Horatio	9	35
74	Horatio	9	35	64	Horatio	7	35

S1.sName	S1.sailorId
Horatio	64
Horatio	74

Project on sName
and sailorId.