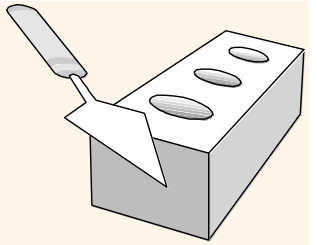
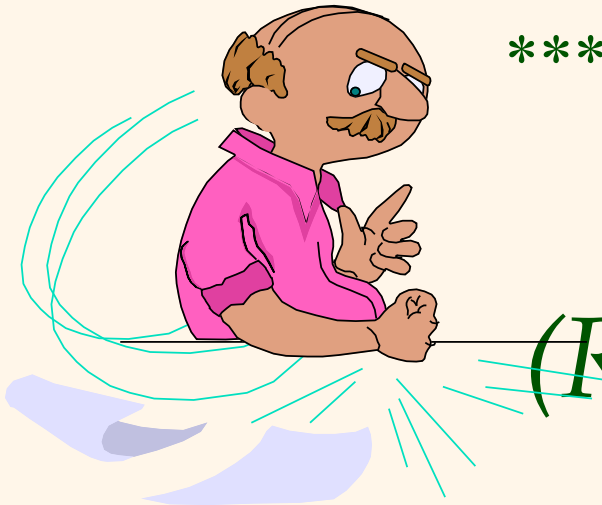


Introduction to Data Management



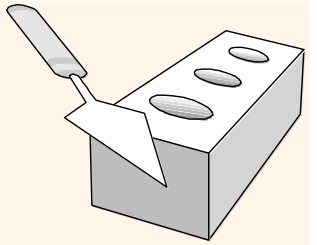
**** The "Flipped" Edition ****

Lecture #10

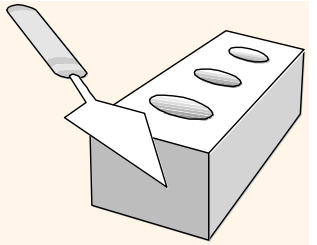
(Relational Languages II)

Instructor: Mike Carey
mjcarey@ics.uci.edu

Today's Notices



- *SWOOSH* HW series status
  - HW1 is almost graded (watch Gradescope)
  - HW2 now over, and HW3 is open for business!
- Midterm 1 info:
  - Relational algebra (etc.) out of scope (Midterm 2)
  - Old exam + solution will be available on the wiki
  - Make yourself a **"cheat sheet"** to use while taking it!
  - You ***must*** be ***physically present*** (w/laptops + masks)!
    - Different 50-minute exams for the two lectures
    - DSC at DSC, remotes only w/explicit (current) permission!
      - No before-exam permission (via e-mail w/me) = **0** (!)



Pre-Midterm Time Check!

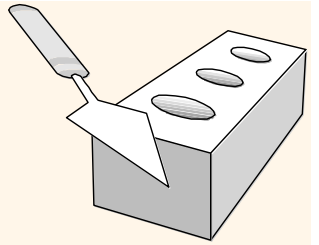
Topic Coverage and Exam Schedule

Syllabus

Topic	Reading (Required!)
Databases and DB Systems	Ch. 1
Entity-Relationship (E-R) Data Model	Ch. 6.1-6.5, 6.8-6.9
Relational Data Model	Ch. 2.1-2.4, 3.1-3.2
E-R to Relational Translation	Ch. 6.6-6.7
Relational Design Theory	Ch. 7.1-7.4.2
Midterm Exam 1	***Fri, Oct 22*** (during lecture time)
Relational Algebra	Ch. 2.5-2.7
Relational Calculus	Wikipedia: Tuple relational calculus
SQL Basics (SPJ and Nested Queries)	Ch. 3.3-3.5
SQL Analytics: Aggregation, Nulls, and Outer Joins	Ch. 3.6-3.9, 4.1
Advanced SQL: Constraints, Triggers, Views, and Security	Ch. 4.2, 4.4-4.5, 4.7
Midterm Exam 2	***Mon, Nov 15*** (during lecture time)
Storage	Ch. 12.1-12.4, 12.6-12.7
Indexing	Ch. 14.1-14.4, 14.5
Physical DB Design	Ch. 14.6-14.7, 15.1-15.3, 15.5.3
Semistructured Data Management (*a.k.a.* NoSQL)	Ch. 8.1, AsterixDB SQL++ Primer, Couchbase SQL++ Book
Data Science 1: Advanced SQL Analytics	Ch. 5.5, 11.3
Data Science 2: Notebooks, Dataframes, and Python/Pandas	Lecture notes and Jupyter notebook
Basics of Transactions	Ch. 4.3, Ch. 17
Endterm Exam	***Fri, Dec 3*** (during lecture time)

Midterm Exam 1

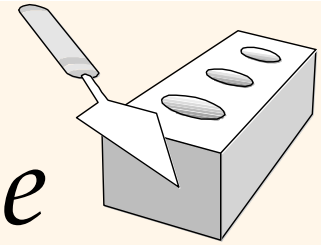
Time: Fri, Oct 22, Lecture Time
Place: SSLH 100



A Few Notes On Honesty...

- Dishonest engineers can severely injure their employers, e.g.,
  - Volkswagen's "benchmark special" ($14.7B!)
  - Uber's self-driving car case ($245M)
- Stanford exams given under an Honor Code
  - Faculty can't even be in the room!
- Be guided by the UCI CS122a Honor Code!
  - I am going to **trust** that you are all mature, grown-up students who want to learn the material
  - What goes around comes around (and pretty quickly...!)
  - I am not the police... (Not ***this*** quarter for sure!)

Ex: *Wisconsin Sailing Club Database*



Sailors

sid	sname	rating	age
22	Dustin	7	45.0
29	Brutus	1	33.0
31	Lubber	8	55.5
32	Andy	8	25.5
58	Rusty	10	35.0
64	Horatio	7	35.0
71	Zorba	10	16.0
74	Horatio	9	35.0
85	Art	4	25.5
95	Bob	3	63.5

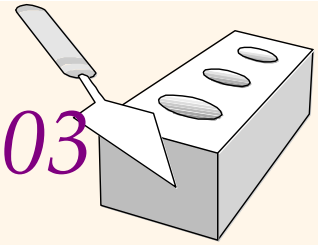
Reserves

sid	bid	date
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	9/8/93

Boats

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

Find names of sailors who've reserved boat #103



Sailors(sid, sname, rating, age)

Reserves(sid, bid, date)

Boats(bid, bname, color)

- Solution 1: $\pi_{sname}((\sigma_{bid=103} Reserves) \bowtie Sailors)$

- Solution 2:

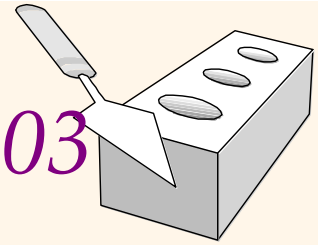
$$\rho(Temp1, \sigma_{bid=103} Reserves)$$

$$\rho(Temp2, Temp1 \bowtie Sailors)$$

$$\pi_{sname}(Temp2)$$

- Solution 3: $\pi_{sname}(\sigma_{bid=103}(Reserves \bowtie Sailors))$

Find names of sailors who've reserved boat #103



Sailors(sid, sname, rating, age)

Reserves(sid, bid, date)

Boats(bid, bname, color)

- Solution 1: $\pi_{sname}((\sigma_{bid=103} \text{Reserves}) \bowtie Sailors)$

- Solution 2:

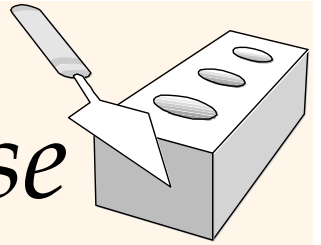
$$Temp1 = \sigma_{bid=103} \text{Reserves}$$

$$Temp2 = Temp1 \bowtie Sailors$$

$$\pi_{sname}(Temp2)$$

- Solution 3: $\pi_{sname}(\sigma_{bid=103}(\text{Reserves} \bowtie Sailors))$

Ex: Wisconsin Sailing Club Database



$\sigma_{bid=103} Reserves$

sid	bid	date
22	103	10/8/98
31	103	11/6/98
74	103	9/8/93

$\pi_{sname}((\sigma_{bid=103} Reserves) \bowtie Sailors)$

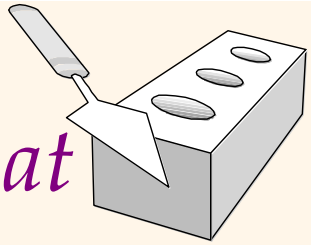
$(\sigma_{bid=103} Reserves) \bowtie Sailors$

sid	bid	date	sname	rating	age
22	103	10/8/98	Dustin	7	45.0
31	103	11/6/98	Lubber	8	55.5
74	103	9/8/93	Horatio	9	35.0

sname
Dustin
Lubber
Horatio

(Solution 1)

Find names of sailors who've reserved a red boat



Sailors(sid, sname, rating, age)

Reserves(sid, bid, date)

Boats(bid, bname, color)

- Information about boat color only available in Boats; so need to do another join:

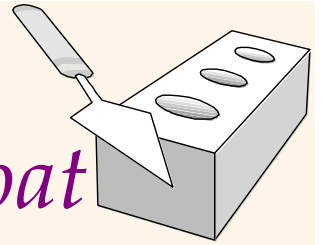
$$\pi_{sname}((\sigma_{color='red'} Boats) \bowtie Reserves \bowtie Sailors)$$

- A more "efficient" solution:

$$\pi_{sname}(\pi_{sid}((\pi_{bid}\sigma_{color='red'} Boats) \bowtie Res) \bowtie Sailors)$$

A query optimizer will find the latter, given the 1st query!

Find sailors who've reserved a red or a green boat



Sailors(sid, sname, rating, age)

Reserves(sid, bid, date)

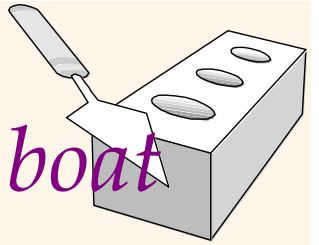
Boats(bid, bname, color)

- Can identify all red or green boats, then find sailors who've reserved one of these boats:

$$\rho\ (Tempboats, (\sigma_{color='red' \vee color='green'} Boats))$$

$$\pi_{sname}(Tempboats \bowtie Reserves \bowtie Sailors)$$

- Could also define Tempboats using *union*! (Q: How?)
- What happens if $\vee$ is replaced by $\wedge$ in this query?



Find sailors who've reserved a red and a green boat

Sailors(sid, sname, rating, age) Reserves(sid, bid, date)

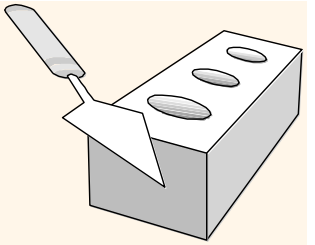
Boats(bid, bname, color)

- Previous form won't work! Must identify sailors who've reserved red boats and sailors who've reserved green boats, then find their intersection (notice that *sid* is a key for Sailors!):

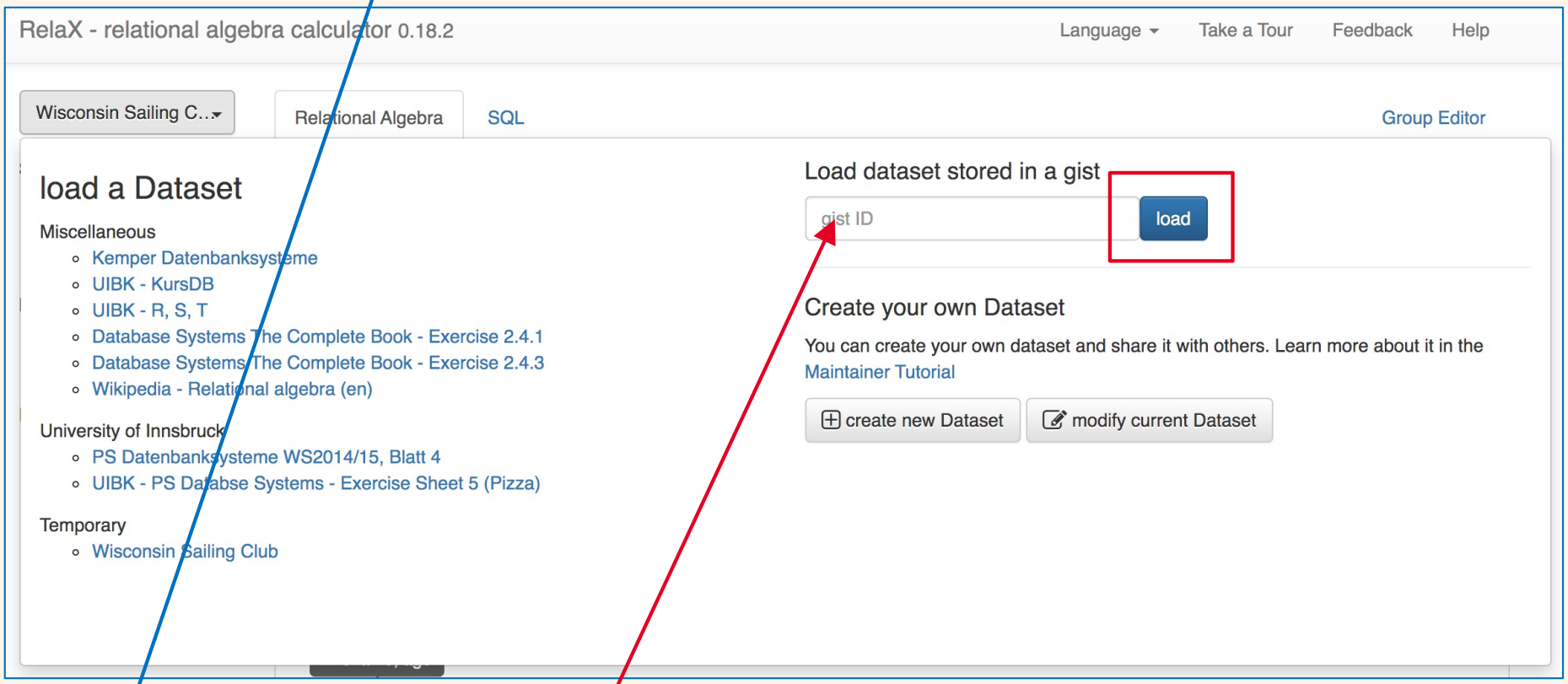
$$\rho\,(Tempred,\ \pi_{sid}((\sigma_{color='red'}\,Boats)\bowtie \mathrm{Re}serves))$$

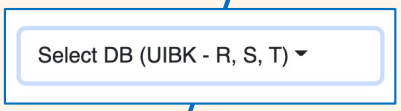
$$\rho\,(Tempgreen,\ \pi_{sid}((\sigma_{color='green'}\,Boats)\bowtie \mathrm{Re}serves))$$

$$\pi_{sname}((Tempred\cap Tempgreen)\bowtie Sailors)$$



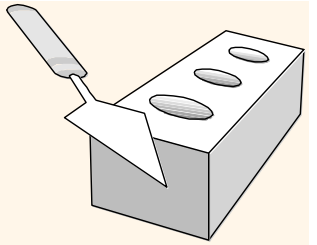
The RelaX "Calculator"





9aa882b7295d64737b8330978c3e0658

https://dbis-uibk.github.io/relax/calc/local/uibk/local/0



The RelaX "Calculator" (cont.)

RelaX - relational algebra calculator 0.18.2

Language ▾ Take a Tour Feedback Help

Wisconsin Sailing C...▾

Sailors
- sid number
- sname string
- rating number
- age number

Boats
- bid number
- bname string
- color string

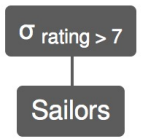
Reserves
- sid number
- bid number
- date date

Relational Algebra | SQL | Group Editor

π σ ρ ← τ γ ∧ ∨ ¬ = ≠ ≥ ≤ ∩ ∪ ÷ - × ⋈ ⋈ ⋈ ⋈ ⋉ ⋊ ▷ = -- /* {}

```
1 σ rating > 7 (Sailors)
```

▶ execute query | download | history ▾

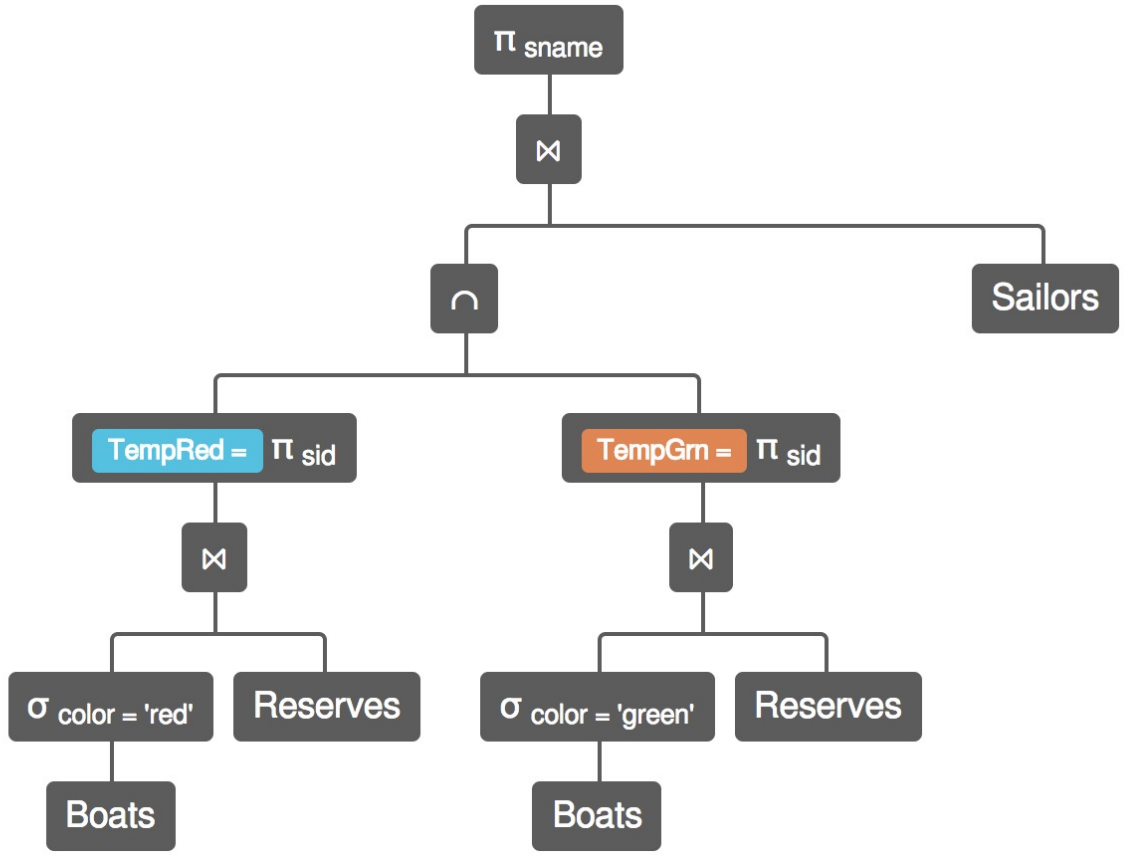


σ rating > 7 (Sailors)

Sailors.sid	Sailors.sname	Sailors.rating	Sailors.age
31	Lubber	8	55.5
32	Andy	8	25.5
58	Rusty	10	35
71	Zorba	10	16
74	Horatio	9	35

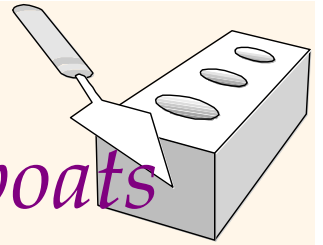
Find sailors who've reserved a red and a green boat

```
1 TempRed = π sid (σ color = 'red' (Boats) ⋈ Reserves)
2 TempGrn = π sid (σ color = 'green' (Boats) ⋈ Reserves)
3 π sname ((TempRed ∩ TempGrn) ⋈ Sailors)
```



$\pi_{sname}(((\pi_{sid}(\sigma_{color='red'}(Boats) \bowtie Reserves)) \cap (\pi_{sid}(\sigma_{color='green'}(Boats) \bowtie Reserves))) \bowtie Sailors)$

Sailors.sname
Dustin
Lubber



Find the names of sailors who've reserved all boats

Sailors(sid, sname, rating, age)

Reserves(sid, bid, date)

Boats(bid, bname, color)

- Uses **division**; schemas of the input relations feeding the / operator must be *carefully chosen*:

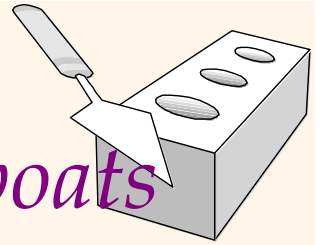
$$\rho\,(Tempsids, (\pi_{sid,bid} Reserves) / (\pi_{bid} Boats))$$

$$\pi_{sname}(Tempsids \bowtie Sailors)$$

- To find sailors who've reserved all 'Interlake' boats:

$$\ldots\ldots \; / \; \pi_{bid}(\sigma_{bname='Interlake'} Boats)$$

Find the names of sailors who've reserved all boats



RelaX - relational algebra calculator 0.18.2

Wisconsin Sailing C...

Sailors
- sid number
- sname string
- raiting number
- age number

Boats
- bid number
- bname string
- color string

Reserves
- sid number
- bid number
- date date

Relational Algebra | SQL

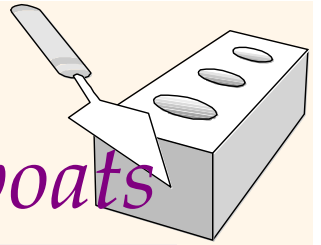
π σ ρ ← τ γ ∧ ∨ ¬ = ≠ ≥ ≤ ∩ ∪

```
1 SBpairs = π sid, bid (Reserves)
2 B = π bid (Boats)
3 AllSlist = SBpairs ÷ B
4 π sname (AllSlist ⋈ Sailors)
```

- U… fe… …en:
- To… …e' boats:

bid` `bname='Interlake'

Find the names of sailors who've reserved all boats



- U
  fe
- To

bid`

RelaX - relational alge

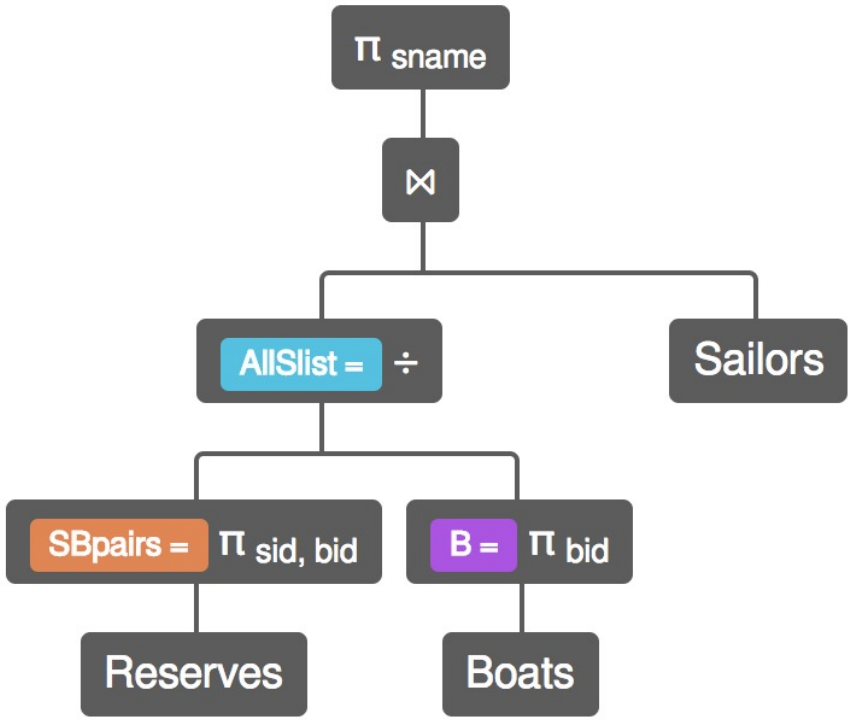
Wisconsin Sailing C...

Sailors
- sid number
- sname string
- raiting number
- age number

Boats
- bid number
- bname string
- color string

Reserves
- sid number
- bid number
- date date

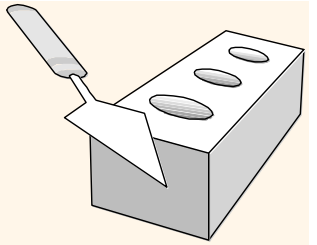
▶ execute query



π sname (((π sid, bid (Reserves)) ÷ (π bid (Boats))) ⋈ Sailors)

Sailors.sname
Dustin

PS: RelaX Renaming Example...

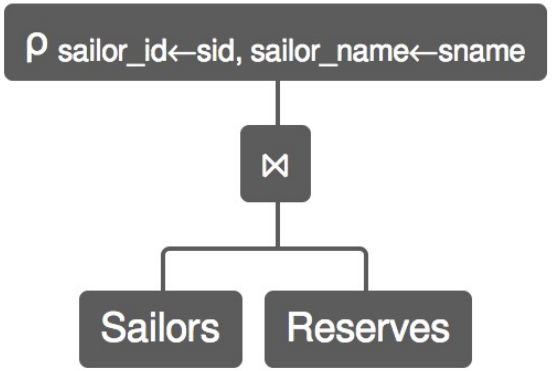


Relational Algebra

π σ ρ ← τ γ ∧

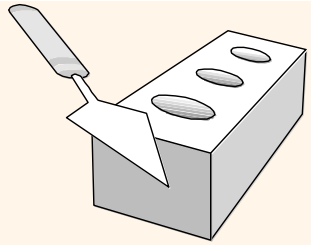
```
1 ρ sailor_id←
```

▶ execute selection



ρ sailor_id←sid, sailor_name←sname (Sailors ⋈ Reserves)

Sailors.sailor_id	Sailors.sailor_name	Sailors.rating	Sailors.age	Reserves.bid	Reserves.date
22	Dustin	7	45	101	1998-10-10
22	Dustin	7	45	102	1998-10-10
22	Dustin	7	45	103	1998-10-08
22	Dustin	7	45	104	1998-10-07
31	Lubber	8	55.5	102	1998-10-11
31	Lubber	8	55.5	103	1998-11-06
31	Lubber	8	55.5	104	1998-11-12
64	Horatio	7	35	101	1998-09-05
64	Horatio	7	35	102	1998-09-02
74	Horatio	9	35	103	1993-09-08



Relational Algebra Summary

- The relational model has (several) rigorously defined query languages that are both simple and powerful in nature.
- Relational algebra is more operational; very useful as an internal representation for *query evaluation plans*.
- Several ways of expressing a given query; a query optimizer should choose the most efficient version. (Take CS122C...! ☺)
- We'll add a few more operators later on…
- Next up for us: *Relational Calculus*