BZAN 6354

Lecture 9

March 18, 2024

Dr. Mark Grimes, Ph.D.
gmgrimes@bauer.uh.edu

UNIVERSITY of HOUSTON
C. T. BAUER COLLEGE of BUSINESS
Department of Decision & Information Sciences

Agenda

- Quick discussion of:
  - Schedule
  - Exam1
- Databases in the “real world”
- Module 11.1 – 11.2: Relational Algebra
- Break
- Module 12.1: Single table SQL functions

Schedule

- Where we've been and where we're going
  - First half of the semester
    - Introduction
    - Conceptual Modeling
    - Relational Modeling
    - Database Creation
  - Second half of the semester
    - Relational Algebra
    - Structured Query Language (SQL)
    - Normalization

- Exam 2 is not "comprehensive", but everything builds on what we've learned so far, so…

Schedule

- SQL Project will be made available next week
- We will not have class on April 8 – all content will just be shifted back by one week, and April 22 will be Advanced/Applied SQL + Wrap up/Review

9: 3/18	Relational Algebra Structured Query Language **Modules 11.1, 11.2, 12.1**	
10: 3/25	Structured Query Language **Module 12.2, 12.3**	**Assignment 3 assigned** **SQL Project Assigned**
11: 4/1	Normalization **Modules 7.1, 7.2, 7.3**	
12: 4/8	~~Normalization~~ **No Class** ~~**Modules 8.1, 8.2, 8.3, 8.4**~~	~~**Assignment 4 assigned**~~
13: 4/15	~~Advanced/Applied SQL~~ **Modules 8.1 – 8.4** ~~**Modules 13.1, 13.2, 13.3**~~	**Assignment 4 Assigned**
14: 4/22	Wrap up/Review **Modules 13.1 – 13.3** **Wrap up / Review**	
15: 4/29	Exam 2 – SQL and Normalization	**Exam 2**

Exam 1

- Average score: 80.1
  - n=30
  - SD=12.1
  - Max = 97
  - Min = 56

- Last semester's average was 78.9 (n=32, SD=11.4, max=91, min=48)
  - Not a statistically significant difference, t(60)=0.4021, p=0.6890
  - (this is a good thing)

- Let me know if you want to review your exam
  - In person
  - On Zoom

Databases in the real world

- Has anyone ever ordered anything online?

- Has anyone ever worked in a warehouse?

Databases in the real world

- Order is placed by customer

Databases in the real world

- A “picker” finds the product in the warehouse

Databases in the real world

- Picker takes the item to a packer, who boxes it up along with other items in the same order

Databases in the real world

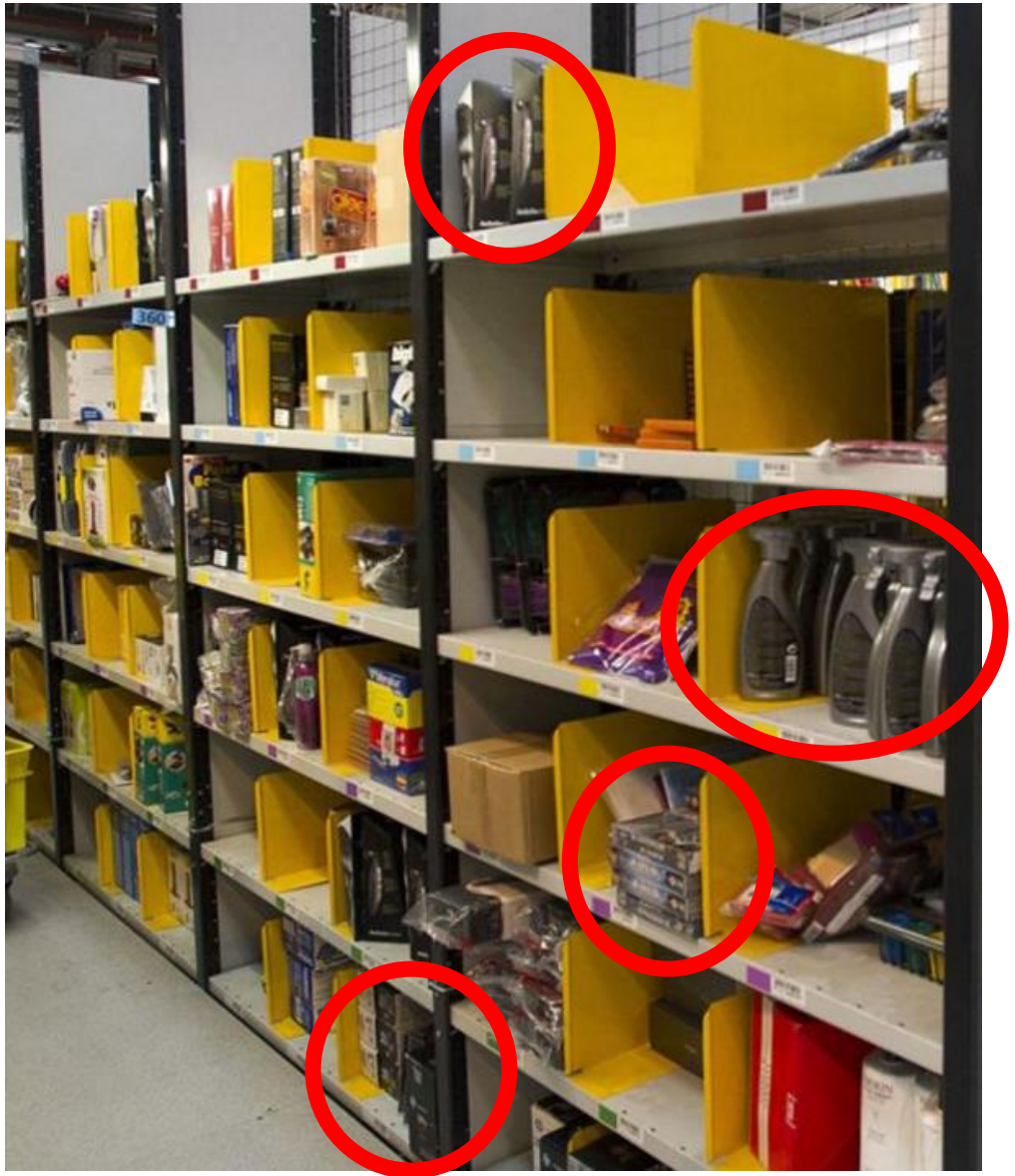
- Order is shipped to you

Databases in the real world

- Obviously lots of ways databases are used here
  - Customers
  - Products
  - Orders
  - Employees
  - Work assignments
  - Product Locations

Databases in the real world

- Notice that products are stored in groups
- Pro: Keeps the pickers sane
- Con: Huge waste of space, inefficient, slow, etc…



A day in the life of an Amazon robot

https://www.youtube.com/watch?v=6KRjuuEVEZs (3:35)

Other similar videos if you are interested
https://www.youtube.com/watch?v=8nKPC-WmLjU (11:08)

https://www.youtube.com/watch?v=i0fEiw4ycyY (6:44)

https://www.youtube.com/watch?v=8gy5tYVR-28 (3:18)

https://www.youtube.com/watch?v=cLVCGEmkJs0 (3:40)

Databases in the real world

- Now products may be “randomly” placed
  - Better use of space
  - Can determine optimal grouping of items
  - MUCH faster/more efficient
- Must track location of individual items – not classes of products
- Your database must be highly reliable!

Some quick thoughts on this system

- Pods, Items, Workers, Customers, Orders, Robots, Pack Stations, Chargers, Pod Locations
- **Customers** place **Orders**
- **Orders** contain **items**
- **Pods** house **items**
- **Pods** stored in **Pod Locations**
- **Robots** fetch **pods**
- **Pods** transported to **Stations** by **robots**
- **Workers** assigned to S**tations**
- **Workers** pack **items** into **orders**
- **Robots** charge at **Chargers**

In summary

- We have discussed a lot of topics, which require a lot of detail and precision
- If you're doing "small potatoes" stuff, maybe you can get away with cutting some corners
- If you're processing 30,000 transactions per day, you've got to do this right – 1% failure = 300 mistakes!
- Millions of transactions a day?
- High dollar transactions?

Module 11.1
Unary Operators

- Select
- Project

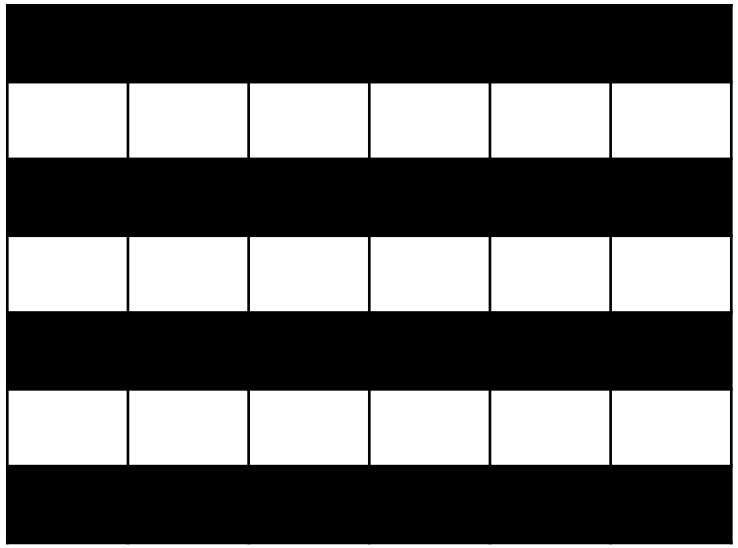
The data used in the upcoming examples are found on pages 540-541 of your book

We talked briefly before the exam

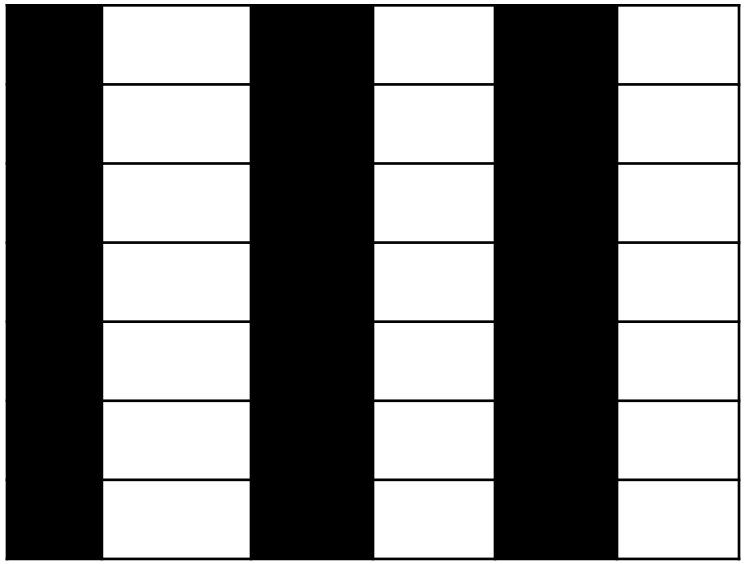
- Two unary operations
  - Selection (σ)
  - Projection (π)
- Six binary operators
  - Union ($\cup$)
  - Intersection ($\cap$)
  - Difference (-)
  - Join ($\bowtie$)
  - Cartesian product (X)
  - Division ($\div$)

Unary Operations

Selection (σ)



Projection (π)



Select Operator

- Selects a horizontal subset of tuples that satisfy a selection condition from the relation

$$\sigma_{\text{<selection condition>}}R$$

- Lower case sigma (σ) designates “select”

- <selection condition> is a Boolean expression specified on the attributes of relation R

Select Operator

$$\sigma_{\text{<selection condition>}}R$$

- R is generally a *relational algebra expression* whose result is a relation – the simplest form of R is a single relation
  - This means relational algebra operations can be “nested”

- The Boolean expression is of the form
  - <attribute name> <comparison operator> <constant value>
    - i.e., Lastname='Smith' or Salary > 30000

    or
  - <attribute name> <comparison operator> <attribute name>
    - i.e., CreditsNeeded >= CreditsCompleted

Select Operator

$\sigma_{<\text{selection condition}>}R$

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	John	Grey	No	M	1,000	mgrimes
4	Trotty	Brown	Yes	M	1,300	mgrimes
5	Rio	Grey	No	F	1,700	tswift
6	Robin	Yellow	No	M	1,100	jisbell
7	Katy	Brown	No	F	1,200	jisbell
8	Pegasus	Brown	No	M	1,750	mgrimes
9	Sammy	Black	Yes	M	2,200	mgrimes
10	Pinky	Red	No	M	1,050	tswift
11	Hulk	Grey	No	M	2,050	mgrimes
12	Pat	White	No	F	1,400	mgrimes
13	Betty	White	Yes	F	1,250	tswift
14	Shamrock	Black	No	M	1,400	[NULL]

- $\sigma_{(\text{Sex} = \text{'F'})}$ Horses ← How many tuples?

Select Operator

$\sigma_{<\text{selection condition}>}R$

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	John	Grey	No	M	1,000	mgrimes
4	Trotty	Brown	Yes	M	1,300	mgrimes
5	Rio	Grey	No	F	1,700	tswift
6	Robin	Yellow	No	M	1,100	jisbell
7	Katy	Brown	No	F	1,200	jisbell
8	Pegasus	Brown	No	M	1,750	mgrimes
9	Sammy	Black	Yes	M	2,200	mgrimes
10	Pinky	Red	No	M	1,050	tswift
11	Hulk	Grey	No	M	2,050	mgrimes
12	Pat	White	No	F	1,400	mgrimes
13	Betty	White	Yes	F	1,250	tswift
14	Shamrock	Black	No	M	1,400	[NULL]

- $\sigma_{(\text{Sex} = \text{'F'})}$ Horses
- $\sigma_{(\text{Sex} = \text{'F' AND Spots} = \text{'No'})}$ Horses ← Now How many tuples?

Select Operator

$\sigma_{<\text{selection condition}>}R$

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	John	Grey	No	M	1,800	mgrimes
4	Trotty	Brown	Yes	M	1,300	mgrimes
5	Ric	Grey	No	F	1,700	tswift
6	Robin	Yellow	No	M	1,100	jisbell
7	Katy	Brown	No	F	1,200	jisbell
8	Pegasus	Brown	No	M	1,750	mgrimes
9	Sammy	Black	Yes	M	2,200	mgrimes
10	Pinky	Red	No	M	1,050	tswift
11	Hulk	Grey	No	M	2,050	mgrimes
12	Pat	White	No	F	1,400	mgrimes
13	Betty	White	Yes	F	1,250	tswift
14	Shamrock	Black	No	M	1,400	[NULL]

- $\sigma_{(\text{Color='Brown' OR Color='White'})}$ Course

Select Operator

$\sigma_{<\text{selection condition}>}R$

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	John	Grey	No	M	1,800	mgrimes
4	Trotty	Brown	Yes	M	1,300	mgrimes
5	Rio	Grey	No	F	1,700	tswift
6	Robin	Yellow	No	M	1,100	jisbell
7	Katy	Brown	No	F	1,200	jisbell
8	Pegasus	Brown	No	M	1,750	mgrimes
9	Sammy	Black	Yes	M	2,200	mgrimes
10	Pinky	Red	No	M	1,050	tswift
11	Hulk	Grey	No	M	2,050	mgrimes
12	Pat	White	No	F	1,400	mgrimes
13	Betty	White	Yes	F	1,250	tswift
14	Shamrock	Black	No	M	1,400	[NULL]

- $\sigma_{((\text{Color}=\text{'Brown'}\ \text{OR}\ \text{Color}=\text{'White'})\ \text{AND}\ \text{Spots}=\text{'No'})}$ Course

Select Operator

$\sigma_{<\text{selection condition}>}R$

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	John	Grey	No	M	1,800	mgrimes
4	Trotty	Brown	Yes	M	1,300	mgrimes
5	Ric	Grey	No	F	1,700	tswift
6	Robin	Yellow	No	M	1,100	jisbell
7	Katy	Brown	No	F	1,200	jisbell
8	Pegasus	Brown	No	M	1,750	mgrimes
9	Sammy	Black	Yes	M	2,200	mgrimes
10	Pinky	Red	No	M	1,050	tswift
11	Hulk	Grey	No	M	2,050	mgrimes
12	Pat	White	No	F	1,400	mgrimes
13	Betty	White	Yes	F	1,250	tswift
14	Shamrock	Black	No	M	1,400	[NULL]

- $\sigma_{(\text{Color='Brown' OR (Color='White' AND Spots ='No'))}}$ Course

Select Operator

- Be careful with multiple operators – these are all different!

(Color='Brown' OR Color='White') AND Spots ='No'

	NAME	COLOR	SPOTS
1	Sam	Brown	No
2	Erica	~~Yellow~~	~~Yes~~
3	John	~~Grey~~	~~No~~
4	Trotty	~~Brown~~	~~Yes~~
5	Rio	~~Grey~~	~~No~~
6	Robin	~~Yellow~~	~~No~~
7	Katy	Brown	No
8	Pegasus	Brown	No
9	Sammy	~~Black~~	~~Yes~~
10	Pinky	~~Red~~	~~No~~
11	Hulk	~~Grey~~	~~No~~
12	Pat	White	No
13	Betty	~~White~~	~~Yes~~
14	Shamrock	~~Black~~	~~No~~

Horses that are **either** brown or white but no spots in either case

Color='Brown' OR (Color='White' AND Spots ='No')

	NAME	COLOR	SPOTS
1	Sam	Brown	No
2	Erica	~~Yellow~~	~~Yes~~
3	John	~~Grey~~	~~No~~
4	Trotty	Brown	Yes
5	Rio	~~Grey~~	~~No~~
6	Robin	~~Yellow~~	~~No~~
7	Katy	Brown	No
8	Pegasus	Brown	No
9	Sammy	~~Black~~	~~Yes~~
10	Pinky	~~Red~~	~~No~~
11	Hulk	~~Grey~~	~~No~~
12	Pat	White	No
13	Betty	~~White~~	~~Yes~~
14	Shamrock	~~Black~~	~~No~~

Horses that are brown (and may have spots) **or** are white with no spots

Color='White' OR (Color='Brown' AND Spots ='No')

	NAME	COLOR	SPOTS
1	Sam	Brown	No
2	Erica	~~Yellow~~	~~Yes~~
3	John	~~Grey~~	~~No~~
4	Trotty	~~Brown~~	~~Yes~~
5	Rio	~~Grey~~	~~No~~
6	Robin	~~Yellow~~	~~No~~
7	Katy	Brown	No
8	Pegasus	Brown	No
9	Sammy	~~Black~~	~~Yes~~
10	Pinky	~~Red~~	~~No~~
11	Hulk	~~Grey~~	~~No~~
12	Pat	White	No
13	Betty	White	Yes
14	Shamrock	~~Black~~	~~No~~

Horses that are white (and may have spots) **or** are brown with no spots

- If you do not include parentheses “AND” is evaluated before “OR”

Project Operator

- Selects a vertical subset of attributes from a relation

$$\pi_{\text{<attribute list>}}R$$

- Lower case pi (π) designates “project”
- <attribute list> is a subset of attributes of relation R

Project Operator

$$\pi_{\text{<attribute list>}}R$$

- As before, R is generally a *relational algebra expression*, the simplest form being a single relation
  - i.e., relational algebra operations can be “nested”
- The result of the projection operation contains the attributes specific in <attribute list> in the same order they are specified in the list
- If the attribute list in a projection is not a superkey (i.e. is not unique), duplicate tuples will NOT be displayed
  - Remember: Relations do not allow duplicate tuples!

Project Operator

$\pi_{<\text{attribute list}>}R$

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	John	Grey	No	M	1,800	mgrimes
4	Trotty	Brown	Yes	M	1,300	mgrimes
5	Rio	Grey	No	F	1,700	tswift
6	Robin	Yellow	No	M	1,100	jisbell
7	Katy	Brown	No	F	1,200	jisbell
8	Pegasus	Brown	No	M	1,750	mgrimes
9	Sammy	Black	Yes	M	2,200	mgrimes
10	Pinky	Red	No	M	1,050	tswift
11	Hulk	Grey	No	M	2,050	mgrimes
12	Pat	White	No	F	1,400	mgrimes
13	Betty	White	Yes	F	1,250	tswift
14	Shamrock	Black	No	M	1,400	[NULL]

- $\pi_{(\text{owner})}\text{Horses}$ ← How many tuples?

	OWNER
1	canderson
2	[NULL]
3	jisbell
4	mgrimes
5	tswift

- Five tuples returned, including NULL
  - Fewer than the 14 tuples in the original relation, since Owner is not a superkey

Project Operator

$\pi_{<\text{attribute list}>}R$

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	John	Grey	No	M	1,800	mgrimes
4	Trotty	Brown	Yes	M	1,300	mgrimes
5	Rio	Grey	No	F	1,700	tswift
6	Robin	Yellow	No	M	1,100	jisbell
7	Katy	Brown	No	F	1,200	jisbell
8	Pegasus	Brown	No	M	1,750	mgrimes
9	Sammy	Black	Yes	M	2,200	mgrimes
10	Pinky	Red	No	M	1,050	tswift
11	Hulk	Grey	No	M	2,050	mgrimes
12	Pat	White	No	F	1,400	mgrimes
13	Betty	White	Yes	F	1,250	tswift
14	Shamrock	Black	No	M	1,400	[NULL]

- $\pi_{(\text{name, owner})}\text{Horses}$ ← How many tuples?

- Fourteen tuples returned
  - Same as the source relation since {name, owner} is a super key!

	NAME	OWNER
1	Sam	mgrimes
2	Erica	canderson
3	John	mgrimes
4	Trotty	mgrimes
5	Rio	tswift
6	Robin	jisbell
7	Katy	jisbell
8	Pegasus	mgrimes
9	Sammy	mgrimes
10	Pinky	tswift
11	Hulk	mgrimes
12	Pat	mgrimes
13	Betty	tswift
14	Shamrock	[NULL]

Module 11.1
Unary Operators

- Select
- Project

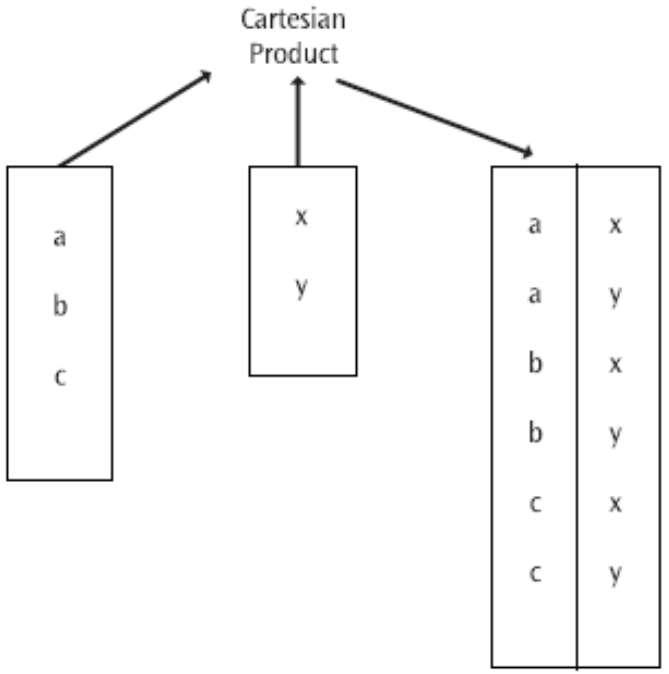
The data used in the upcoming examples are found on pages 540-541 of your book

Module 11.2
Binary Operators

- Cartesian Product
- Union, Intersection, and Difference
- Join
- Division

Cartesian Product

- Also known as “Cross-Product” or just “Product”
- All combinations of tuples from two relations
  - Syntax: R1 X R2
- Will result in n_{R1} x n_{R2} tuples



Cartesian Product

- Note: For the upcoming examples we have created two new (simplified) relations using the project operation
  - H = $\pi_{(Name, Color, Weight, Owner)}$Horses
  - C = $\pi_{(Username, Fname, Lname)}$Customers

  - I have also removed "ssimpson" as the owner for Shamrock – If you want to follow along you can update your table with:

```
UPDATE Horses SET owner=NULL where Name='Shamrock';
```

- H has 14 tuples
- C has 5 tuples

- How many tuples will the Cartesian product (H X C) have?

	NAME	COLOR	WEIGHT	OWNER
1	Sam	Brown	1,500	mgrimes
2	Erica	Yellow	920	canderson
3	John	Grey	1,800	mgrimes
4	Trotty	Brown	1,300	mgrimes
5	Rio	Grey	1,700	tswift
6	Robin	Yellow	1,100	jisbell
7	Katy	Brown	1,200	jisbell
8	Pegasus	Brown	1,750	mgrimes
9	Sammy	Black	2,200	mgrimes
10	Pinky	Red	1,050	tswift
11	Hulk	Grey	2,050	mgrimes
12	Pat	White	1,400	mgrimes
13	Betty	White	1,250	tswift
14	Shamrock	Black	1,400	[NULL]

	USERNAME	FNAME	LNAME
1	mgrimes	Marvin	Grimes
2	canderson	Christine	Anderson
3	tswift	Tina	Swift
4	jisbell	Jason	Isbell
5	ssimpson	Sam	Simpson

Cartesian Product

- Results of H X C: 70 tuples (only first 28 shown…)

	NAME	COLOR	WEIGHT	OWNER
1	Sam	Brown	1,500	mgrimes
2	Erica	Yellow	920	canderson
3	John	Grey	1,800	mgrimes
4	Trotty	Brown	1,300	mgrimes
5	Rio	Grey	1,700	tswift
6	Robin	Yellow	1,100	jisbell
7	Katy	Brown	1,200	jisbell
8	Pegasus	Brown	1,750	mgrimes
9	Sammy	Black	2,200	mgrimes
10	Pinky	Red	1,050	tswift
11	Hulk	Grey	2,050	mgrimes
12	Pat	White	1,400	mgrimes
13	Betty	White	1,250	tswift
14	Shamrock	Black	1,400	[NULL]

	USERNAME	FNAME	LNAME
1	mgrimes	Marvin	Grimes
2	canderson	Christine	Anderson
3	tswift	Tina	Swift
4	jisbell	Jason	Isbell
5	ssimpson	Sam	Simpson

	NAME	COLOR	WEIGHT	OWNER	USERNAME	FNAME	LNAME
1	Sam	Brown	1,500	mgrimes	mgrimes	Marvin	Grimes
2	Erica	Yellow	920	canderson	mgrimes	Marvin	Grimes
3	John	Grey	1,800	mgrimes	mgrimes	Marvin	Grimes
4	Trotty	Brown	1,300	mgrimes	mgrimes	Marvin	Grimes
5	Rio	Grey	1,700	tswift	mgrimes	Marvin	Grimes
6	Robin	Yellow	1,100	jisbell	mgrimes	Marvin	Grimes
7	Katy	Brown	1,200	jisbell	mgrimes	Marvin	Grimes
8	Pegasus	Brown	1,750	mgrimes	mgrimes	Marvin	Grimes
9	Sammy	Black	2,200	mgrimes	mgrimes	Marvin	Grimes
10	Pinky	Red	1,050	tswift	mgrimes	Marvin	Grimes
11	Hulk	Grey	2,050	mgrimes	mgrimes	Marvin	Grimes
12	Pat	White	1,400	mgrimes	mgrimes	Marvin	Grimes
13	Betty	White	1,250	tswift	mgrimes	Marvin	Grimes
14	Shamrock	Black	1,400	[NULL]	mgrimes	Marvin	Grimes
15	Sam	Brown	1,500	mgrimes	canderson	Christine	Anderson
16	Erica	Yellow	920	canderson	canderson	Christine	Anderson
17	John	Grey	1,800	mgrimes	canderson	Christine	Anderson
18	Trotty	Brown	1,300	mgrimes	canderson	Christine	Anderson
19	Rio	Grey	1,700	tswift	canderson	Christine	Anderson
20	Robin	Yellow	1,100	jisbell	canderson	Christine	Anderson
21	Katy	Brown	1,200	jisbell	canderson	Christine	Anderson
22	Pegasus	Brown	1,750	mgrimes	canderson	Christine	Anderson
23	Sammy	Black	2,200	mgrimes	canderson	Christine	Anderson
24	Pinky	Red	1,050	tswift	canderson	Christine	Anderson
25	Hulk	Grey	2,050	mgrimes	canderson	Christine	Anderson
26	Pat	White	1,400	mgrimes	canderson	Christine	Anderson
27	Betty	White	1,250	tswift	canderson	Christine	Anderson
28	Shamrock	Black	1,400	[NULL]	canderson	Christine	Anderson

(next would be every horse with tswift…)

Cartesian Product

- The Cartesian product is of little value by itself
- When combined with the select operator it can be equivalent to a join operation, which is much more useful
- We previously mentioned RA operations can be nested:

  $\sigma_{(H.Owner=C.Username)}\ H \times C$

  - H X C is a relation that is the result of a Cartesian product operation
  - This will return all tuples from the product of course and department WHERE the values of H.Owner match C.Username

	NAME	COLOR	WEIGHT	OWNER
1	Sam	Brown	1,500	mgrimes
2	Erica	Yellow	920	canderson
3	John	Grey	1,800	mgrimes
4	Trotty	Brown	1,300	mgrimes
5	Rio	Grey	1,700	tswift
6	Robin	Yellow	1,100	jisbell
7	Katy	Brown	1,200	jisbell
8	Pegasus	Brown	1,750	mgrimes
9	Sammy	Black	2,200	mgrimes
10	Pinky	Red	1,050	tswift
11	Hulk	Grey	2,050	mgrimes
12	Pat	White	1,400	mgrimes
13	Betty	White	1,250	tswift
14	Shamrock	Black	1,400	[NULL]

	USERNAME	FNAME	LNAME
1	mgrimes	Marvin	Grimes
2	canderson	Christine	Anderson
3	tswift	Tina	Swift
4	jisbell	Jason	Isbell
5	ssimpson	Sam	Simpson

Cartesian Product with selection

- This is more useful: $\sigma_{(H.Owner=C.Username)}\ H \times C$

	NAME	COLOR	WEIGHT	OWNER
1	Sam	Brown	1,500	mgrimes
2	Erica	Yellow	920	canderson
3	John	Grey	1,800	mgrimes
4	Trotty	Brown	1,300	mgrimes
5	Rio	Grey	1,700	tswift
6	Robin	Yellow	1,100	jisbell
7	Katy	Brown	1,200	jisbell
8	Pegasus	Brown	1,750	mgrimes
9	Sammy	Black	2,200	mgrimes
10	Pinky	Red	1,050	tswift
11	Hulk	Grey	2,050	mgrimes
12	Pat	White	1,400	mgrimes
13	Betty	White	1,250	tswift
14	Shamrock	Black	1,400	[NULL]

	USERNAME	FNAME	LNAME
1	mgrimes	Marvin	Grimes
2	canderson	Christine	Anderson
3	tswift	Tina	Swift
4	jisbell	Jason	Isbell
5	ssimpson	Sam	Simpson

Remember, this keeps going for 70 rows→

	NAME	COLOR	WEIGHT	OWNER	USERNAME	FNAME	LNAME
1	Sam	Brown	1,500	mgrimes	mgrimes	Marvin	Grimes
2	Erica	Yellow	920	canderson	mgrimes	Marvin	Grimes
3	John	Grey	1,800	mgrimes	mgrimes	Marvin	Grimes
4	Trotty	Brown	1,300	mgrimes	mgrimes	Marvin	Grimes
5	Rio	Grey	1,700	tswift	mgrimes	Marvin	Grimes
6	Robin	Yellow	1,100	jisbell	mgrimes	Marvin	Grimes
7	Katy	Brown	1,200	jisbell	mgrimes	Marvin	Grimes
8	Pegasus	Brown	1,750	mgrimes	mgrimes	Marvin	Grimes
9	Sammy	Black	2,200	mgrimes	mgrimes	Marvin	Grimes
10	Pinky	Red	1,050	tswift	mgrimes	Marvin	Grimes
11	Hulk	Grey	2,050	mgrimes	mgrimes	Marvin	Grimes
12	Pat	White	1,400	mgrimes	mgrimes	Marvin	Grimes
13	Betty	White	1,250	tswift	mgrimes	Marvin	Grimes
14	Shamrock	Black	1,400	[NULL]	mgrimes	Marvin	Grimes
15	Sam	Brown	1,500	mgrimes	canderson	Christine	Anderson
16	Erica	Yellow	920	canderson	canderson	Christine	Anderson
17	John	Grey	1,800	mgrimes	canderson	Christine	Anderson
18	Trotty	Brown	1,300	mgrimes	canderson	Christine	Anderson
19	Rio	Grey	1,700	tswift	canderson	Christine	Anderson
20	Robin	Yellow	1,100	jisbell	canderson	Christine	Anderson
21	Katy	Brown	1,200	jisbell	canderson	Christine	Anderson
22	Pegasus	Brown	1,750	mgrimes	canderson	Christine	Anderson
23	Sammy	Black	2,200	mgrimes	canderson	Christine	Anderson
24	Pinky	Red	1,050	tswift	canderson	Christine	Anderson
25	Hulk	Grey	2,050	mgrimes	canderson	Christine	Anderson
26	Pat	White	1,400	mgrimes	canderson	Christine	Anderson
27	Betty	White	1,250	tswift	canderson	Christine	Anderson
28	Shamrock	Black	1,400	[NULL]	canderson	Christine	Anderson

Cartesian Product with selection

$$\sigma_{(H.Owner=C.Username)}\ H \times C$$

	NAME	COLOR	WEIGHT	OWNER	USERNAME	FNAME	LNAME
1	Sam	Brown	1,500	mgrimes	mgrimes	Marvin	Grimes
2	Erica	Yellow	920	canderson	canderson	Christine	Anderson
3	John	Grey	1,800	mgrimes	mgrimes	Marvin	Grimes
4	Trotty	Brown	1,300	mgrimes	mgrimes	Marvin	Grimes
5	Rio	Grey	1,700	tswift	tswift	Tina	Swift
6	Robin	Yellow	1,100	jisbell	jisbell	Jason	Isbell
7	Katy	Brown	1,200	jisbell	jisbell	Jason	Isbell
8	Pegasus	Brown	1,750	mgrimes	mgrimes	Marvin	Grimes
9	Sammy	Black	2,200	mgrimes	mgrimes	Marvin	Grimes
10	Pinky	Red	1,050	tswift	tswift	Tina	Swift
11	Hulk	Grey	2,050	mgrimes	mgrimes	Marvin	Grimes
12	Pat	White	1,400	mgrimes	mgrimes	Marvin	Grimes
13	Betty	White	1,250	tswift	tswift	Tina	Swift

Cartesian Product with selection & projection

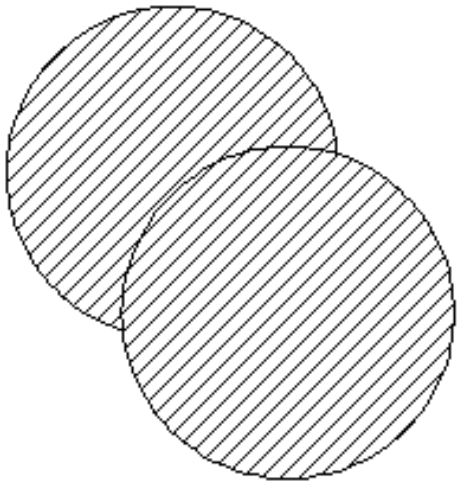
- We could add a projection operation if we are only interested in certain attributes:

$$\pi_{(H.Name,H.Color,\ C.Fname,\ C.Lname)}\sigma_{(H.Owner=C.Username)}(H \times C)$$

	NAME	COLOR	FNAME	LNAME
1	Sam	Brown	Marvin	Grimes
2	Erica	Yellow	Christine	Anderson
3	John	Grey	Marvin	Grimes
4	Trotty	Brown	Marvin	Grimes
5	Rio	Grey	Tina	Swift
6	Robin	Yellow	Jason	Isbell
7	Katy	Brown	Jason	Isbell
8	Pegasus	Brown	Marvin	Grimes
9	Sammy	Black	Marvin	Grimes
10	Pinky	Red	Tina	Swift
11	Hulk	Grey	Marvin	Grimes
12	Pat	White	Marvin	Grimes
13	Betty	White	Tina	Swift

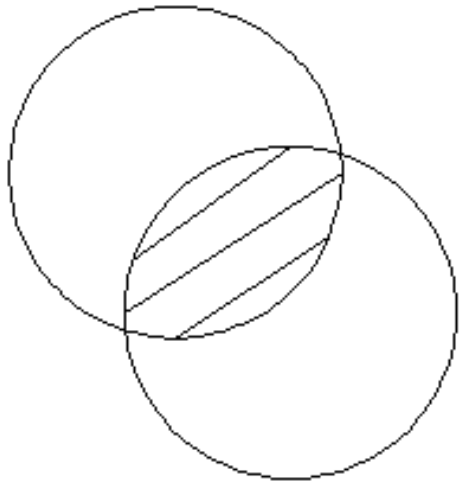
Set Operations

Union



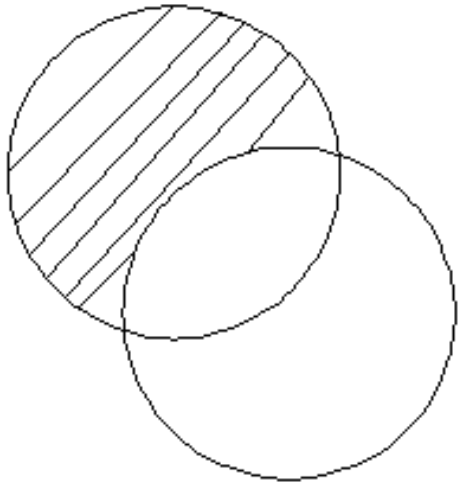
“OR”

Intersection



“AND”

Difference



“NOT”

Set Operations

- Two relations (R and S) are “union compatible” if they:
  - Have the same degree (number of attributes)
  - Pairs of attributes from R and S have the same domain

- Union compatibility is a requirement for all set operations (union, intersection, and difference)

Set Operations

- Relation F: Horses that are Female

	NAME	COLOR	SPOTS	SEX	WEIGHT
1	Sam	Brown	No	F	1,500
2	Erica	Yellow	Yes	F	920
3	Rio	Grey	No	F	1,700
4	Katy	Brown	No	F	1,200
5	Pat	White	No	F	1,400
6	Betty	White	Yes	F	1,250

- Relation S: Horses that have spots

	NAME	COLOR	SPOTS	SEX	WEIGHT
1	Erica	Yellow	Yes	F	920
2	Trotty	Brown	Yes	M	1,300
3	Sammy	Black	Yes	M	2,200
4	Betty	White	Yes	F	1,250

- Union compatible?
  - Note: I created these relations by just SELECTing from the Horses relation

Union

- Horses that are EITHER Female OR have spots

	NAME	COLOR	SPOTS	SEX	WEIGHT
1	Sam	Brown	No	F	1,500
2	Erica	Yellow	Yes	F	920
3	Rio	Grey	No	F	1,700
4	Katy	Brown	No	F	1,200
5	Pat	White	No	F	1,400
6	Betty	White	Yes	F	1,250

	NAME	COLOR	SPOTS	SEX	WEIGHT
1	Erica	Yellow	Yes	F	920
2	Trotty	Brown	Yes	M	1,300
3	Sammy	Black	Yes	M	2,200
4	Betty	White	Yes	F	1,250

F ∪ S

	NAME	COLOR	SPOTS	SEX	WEIGHT
1	Betty	White	Yes	F	1,250
2	Erica	Yellow	Yes	F	920
3	Katy	Brown	No	F	1,200
4	Pat	White	No	F	1,400
5	Rio	Grey	No	F	1,700
6	Sam	Brown	No	F	1,500
7	Sammy	Black	Yes	M	2,200
8	Trotty	Brown	Yes	M	1,300

- Note that only unique tuples are returned
  - The resulting relation has 8, not 10 tuples
  - Remember: Relations do not allow duplicates!

Intersection

- Horses that are Female AND have spots

	NAME	COLOR	SPOTS	SEX	WEIGHT
1	Sam	Brown	No	F	1,500
2	Erica	Yellow	Yes	F	920
3	Rio	Grey	No	F	1,700
4	Katy	Brown	No	F	1,200
5	Pat	White	No	F	1,400
6	Betty	White	Yes	F	1,250

	NAME	COLOR	SPOTS	SEX	WEIGHT
1	Erica	Yellow	Yes	F	920
2	Trotty	Brown	Yes	M	1,300
3	Sammy	Black	Yes	M	2,200
4	Betty	White	Yes	F	1,250

$F \cap S$

	NAME	COLOR	SPOTS	SEX	WEIGHT
1	Betty	White	Yes	F	1,250
2	Erica	Yellow	Yes	F	920

Difference

- Horses that are female but do NOT have spots

	NAME	COLOR	SPOTS	SEX	WEIGHT
1	Sam	Brown	No	F	1,500
2	Erica	Yellow	Yes	F	920
3	Rio	Grey	No	F	1,700
4	Katy	Brown	No	F	1,200
5	Pat	White	No	F	1,400
6	Betty	White	Yes	F	1,250

	NAME	COLOR	SPOTS	SEX	WEIGHT
1	Erica	Yellow	Yes	F	920
2	Trotty	Brown	Yes	M	1,300
3	Sammy	Black	Yes	M	2,200
4	Betty	White	Yes	F	1,250

F - S

	NAME	COLOR	SPOTS	SEX	WEIGHT
1	Katy	Brown	No	F	1,200
2	Pat	White	No	F	1,400
3	Rio	Grey	No	F	1,700
4	Sam	Brown	No	F	1,500

Difference

- Horses that have spots but are NOT female

	NAME	COLOR	SPOTS	SEX	WEIGHT
1	Sam	Brown	No	F	1,500
2	Erica	Yellow	Yes	F	920
3	Rio	Grey	No	F	1,700
4	Katy	Brown	No	F	1,200
5	Pat	White	No	F	1,400
6	Betty	White	Yes	F	1,250

	NAME	COLOR	SPOTS	SEX	WEIGHT
1	Erica	Yellow	Yes	F	920
2	Trotty	Brown	Yes	M	1,300
3	Sammy	Black	Yes	M	2,200
4	Betty	White	Yes	F	1,250

S - F

	NAME	COLOR	SPOTS	SEX	WEIGHT
1	Sammy	Black	Yes	M	2,200
2	Trotty	Brown	Yes	M	1,300

Join Operators

- The JOIN operation is used to combine related tuples from two relations into a single tuple when the join condition is satisfied.
  - Symbol is ⋈, written in the book as $[\mathrm{X}]$
- In order to join the two relations R and S, they must be join compatible
  - The join operation must involve attributes from R and S which share the same domain
- General form: R ⋈ $_{\text{<join condition>}}$ S
- Resulting relation will have:
  - Number of attributes (degrees) equal to the number of attributes in R plus the number of attributes in S
  - Number of tuples (cardinality) equal to number of combinations of R and S where the combination satisfies the join condition

Join Operators

- The primary difference in Join and Cartesian product is that with join, only combinations of tuples satisfying the join condition will appear in the result, whereas in the Cartesian product, all combinations are in the result

- This Cartesian product returns 70 fairly useless tuples:
  - $H \times C$

- This Cartesian product with selection is equivalent to an equijoin operation:
  - $\sigma_{(H.Owner=C.Username)} H \times C$

- This is how the same RA is written as a join:
  - $H \bowtie_{(H.Owner=C.Username)} C$

- These are two DIFFERENT processes that result in the same outcome!

	NAME	COLOR	WEIGHT	OWNER
1	Sam	Brown	1,500	mgrimes
2	Erica	Yellow	920	canderson
3	John	Grey	1,800	mgrimes
4	Trotty	Brown	1,300	mgrimes
5	Rio	Grey	1,700	tswift
6	Robin	Yellow	1,100	jisbell
7	Katy	Brown	1,200	jisbell
8	Pegasus	Brown	1,750	mgrimes
9	Sammy	Black	2,200	mgrimes
10	Pinky	Red	1,050	tswift
11	Hulk	Grey	2,050	mgrimes
12	Pat	White	1,400	mgrimes
13	Betty	White	1,250	tswift
14	Shamrock	Black	1,400	[NULL]

	USERNAME	FNAME	LNAME
1	mgrimes	Marvin	Grimes
2	canderson	Christine	Anderson
3	tswift	Tina	Swift
4	jisbell	Jason	Isbell
5	ssimpson	Sam	Simpson

Join Operations

- We can further refine our selection criteria, and add a projection operation if we are only interested in certain attributes:

$$\pi_{(H.Name,\ H.Weight,\ C.Fname,\ C.Lname)}\ \sigma_{(Weight<1500)}\ H \bowtie_{(H.Owner=C.Username)} C$$

	NAME	COLOR	WEIGHT	OWNER
1	Sam	Brown	1,500	mgrimes
2	Erica	Yellow	920	canderson
3	John	Grey	1,800	mgrimes
4	Trotty	Brown	1,300	mgrimes
5	Rio	Grey	1,700	tswift
6	Robin	Yellow	1,100	jisbell
7	Katy	Brown	1,200	jisbell
8	Pegasus	Brown	1,750	mgrimes
9	Sammy	Black	2,200	mgrimes
10	Pinky	Red	1,050	tswift
11	Hulk	Grey	2,050	mgrimes
12	Pat	White	1,400	mgrimes
13	Betty	White	1,250	tswift
14	Shamrock	Black	1,400	[NULL]

	USERNAME	FNAME	LNAME
1	mgrimes	Marvin	Grimes
2	canderson	Christine	Anderson
3	tswift	Tina	Swift
4	jisbell	Jason	Isbell
5	ssimpson	Sam	Simpson

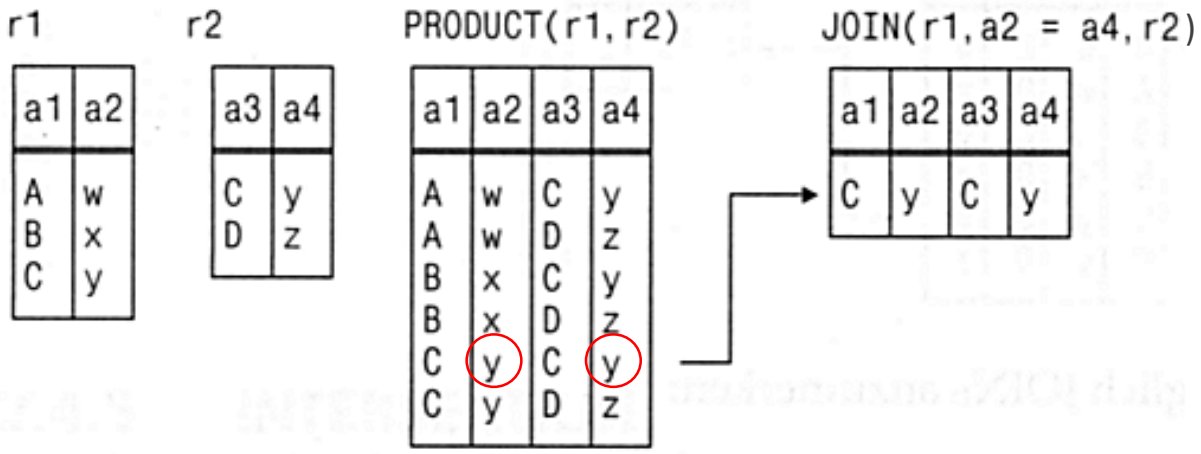
	NAME	WEIGHT	FNAME	LNAME
1	Erica	920	Christine	Anderson
2	Trotty	1,300	Marvin	Grimes
3	Robin	1,100	Jason	Isbell
4	Katy	1,200	Jason	Isbell
5	Pinky	1,050	Tina	Swift
6	Pat	1,400	Marvin	Grimes
7	Betty	1,250	Tina	Swift

Variations on Join

- Equijoin (⋈)
  - Most common
- Natural Join (*)
  - Special case of equijoin
- Theta Join
  - Uncommon, opposite of equijoin
- Outer Join
  - Left outer join (⟕)
  - Right outer join (⟖)
  - Full outer join (⟗)

Equijoin Operation

- Involves join conditions with equality comparisons only. The result of an Equijoin includes all attributes from both relations participating in the join operation. This implies duplication of the joining attributes in the result.



Equijoin Operation

- Much more useful than the Cartesian product, but the attribute used to join the relations is repeated, which is not really useful…

$$H \bowtie_{(H.Owner=C.Username)} C$$

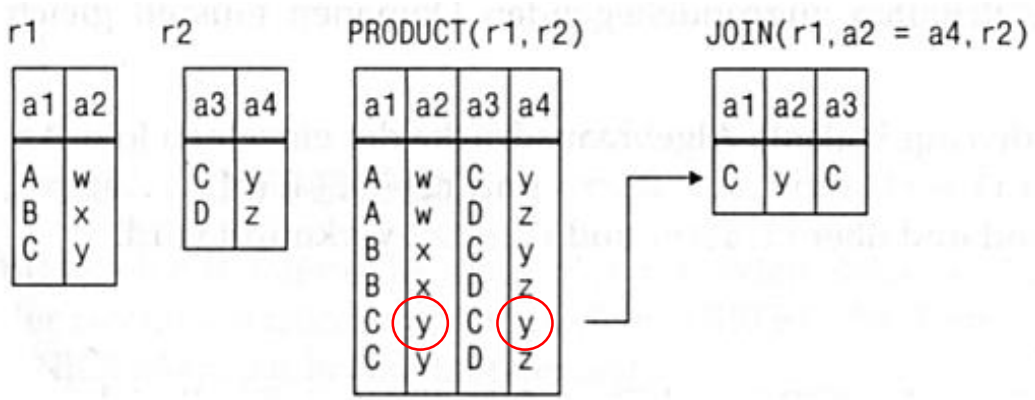
	NAME	COLOR	WEIGHT	OWNER
1	Sam	Brown	1,500	mgrimes
2	Erica	Yellow	920	canderson
3	John	Grey	1,800	mgrimes
4	Trotty	Brown	1,300	mgrimes
5	Rio	Grey	1,700	tswift
6	Robin	Yellow	1,100	jisbell
7	Katy	Brown	1,200	jisbell
8	Pegasus	Brown	1,750	mgrimes
9	Sammy	Black	2,200	mgrimes
10	Pinky	Red	1,050	tswift
11	Hulk	Grey	2,050	mgrimes
12	Pat	White	1,400	mgrimes
13	Betty	White	1,250	tswift
14	Shamrock	Black	1,400	[NULL]

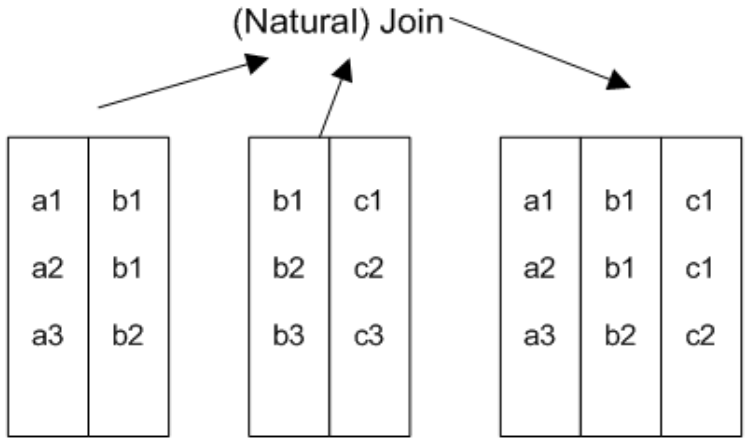
	USERNAME	FNAME	LNAME
1	mgrimes	Marvin	Grimes
2	canderson	Christine	Anderson
3	tswift	Tina	Swift
4	jisbell	Jason	Isbell
5	ssimpson	Sam	Simpson

	NAME	COLOR	WEIGHT	OWNER	USERNAME	FNAME	LNAME
1	Sam	Brown	1,500	mgrimes	mgrimes	Marvin	Grimes
2	Erica	Yellow	920	canderson	canderson	Christine	Anderson
3	John	Grey	1,800	mgrimes	mgrimes	Marvin	Grimes
4	Trotty	Brown	1,300	mgrimes	mgrimes	Marvin	Grimes
5	Rio	Grey	1,700	tswift	tswift	Tina	Swift
6	Robin	Yellow	1,100	jisbell	jisbell	Jason	Isbell
7	Katy	Brown	1,200	jisbell	jisbell	Jason	Isbell
8	Pegasus	Brown	1,750	mgrimes	mgrimes	Marvin	Grimes
9	Sammy	Black	2,200	mgrimes	mgrimes	Marvin	Grimes
10	Pinky	Red	1,050	tswift	tswift	Tina	Swift
11	Hulk	Grey	2,050	mgrimes	mgrimes	Marvin	Grimes
12	Pat	White	1,400	mgrimes	mgrimes	Marvin	Grimes
13	Betty	White	1,250	tswift	tswift	Tina	Swift

Natural Join Operation

- Because the result of an Equijoin results in pairs of attributes with identical values in all the tuples, a new relational algebra operation called a Natural Join, denoted by *, was created to omit the second (and superfluous) attribute in an EQUIJOIN condition.





Natural Join Operation

- The Natural join gets rid of the repeated attribute

$$H *_{(H.Owner=C.Username)} C$$

	NAME	COLOR	WEIGHT	OWNER
1	Sam	Brown	1,500	mgrimes
2	Erica	Yellow	920	canderson
3	John	Grey	1,800	mgrimes
4	Trotty	Brown	1,300	mgrimes
5	Rio	Grey	1,700	tswift
6	Robin	Yellow	1,100	jisbell
7	Katy	Brown	1,200	jisbell
8	Pegasus	Brown	1,750	mgrimes
9	Sammy	Black	2,200	mgrimes
10	Pinky	Red	1,050	tswift
11	Hulk	Grey	2,050	mgrimes
12	Pat	White	1,400	mgrimes
13	Betty	White	1,250	tswift
14	Shamrock	Black	1,400	[NULL]

	USERNAME	FNAME	LNAME
1	mgrimes	Marvin	Grimes
2	canderson	Christine	Anderson
3	tswift	Tina	Swift
4	jisbell	Jason	Isbell
5	ssimpson	Sam	Simpson

	NAME	COLOR	WEIGHT	OWNER	FNAME	LNAME
1	Sam	Brown	1,500	mgrimes	Marvin	Grimes
2	Erica	Yellow	920	canderson	Christine	Anderson
3	John	Grey	1,800	mgrimes	Marvin	Grimes
4	Trotty	Brown	1,300	mgrimes	Marvin	Grimes
5	Rio	Grey	1,700	tswift	Tina	Swift
6	Robin	Yellow	1,100	jisbell	Jason	Isbell
7	Katy	Brown	1,200	jisbell	Jason	Isbell
8	Pegasus	Brown	1,750	mgrimes	Marvin	Grimes
9	Sammy	Black	2,200	mgrimes	Marvin	Grimes
10	Pinky	Red	1,050	tswift	Tina	Swift
11	Hulk	Grey	2,050	mgrimes	Marvin	Grimes
12	Pat	White	1,400	mgrimes	Marvin	Grimes
13	Betty	White	1,250	tswift	Tina	Swift

Natural Join Operation

- Weakness of the natural join is that the attributes used for the join must have the same name in both relations!
- How else could we do this?

$H *_{(H.Owner=C.Username)} C$

$\pi_{(H.NAME,H.Color,H.Weight,H.Owner,C.Fname,C.Lname)} H \bowtie_{(H.Owner=C.Username)} C$

$\pi_{(H.NAME,H.Color,H.Weight,H.Owner,C.Fname,C.Lname)} \sigma_{(H.Owner=C.Username)} H \, X \, C$

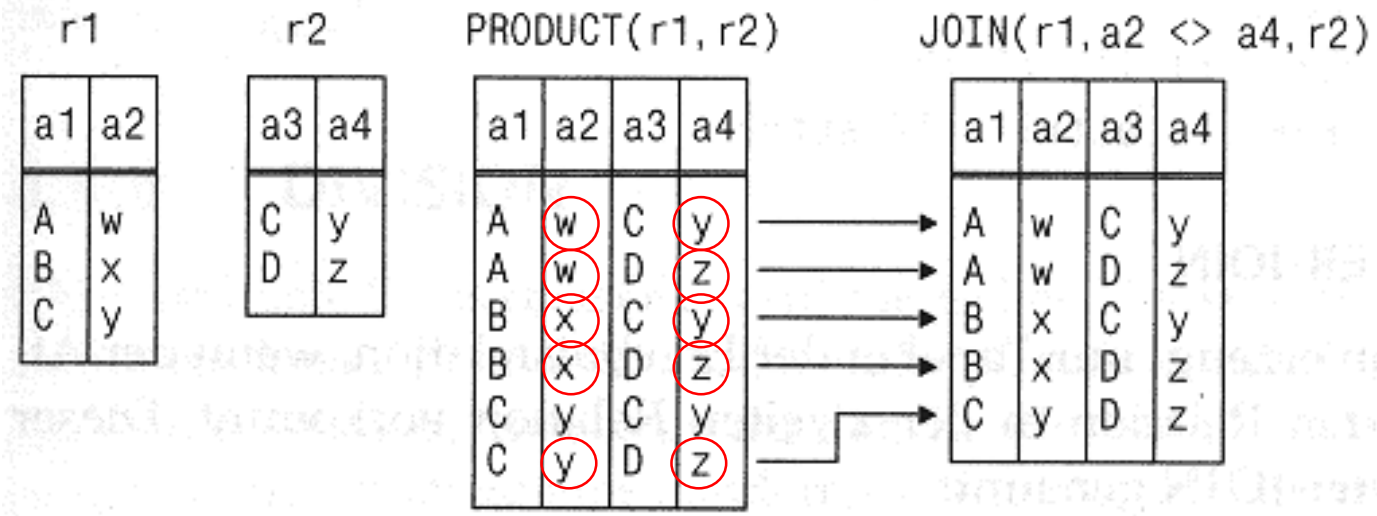
	NAME	COLOR	WEIGHT	OWNER
1	Sam	Brown	1,500	mgrimes
2	Erica	Yellow	920	canderson
3	John	Grey	1,800	mgrimes
4	Trotty	Brown	1,300	mgrimes
5	Rio	Grey	1,700	tswift
6	Robin	Yellow	1,100	jisbell
7	Katy	Brown	1,200	jisbell
8	Pegasus	Brown	1,750	mgrimes
9	Sammy	Black	2,200	mgrimes
10	Pinky	Red	1,050	tswift
11	Hulk	Grey	2,050	mgrimes
12	Pat	White	1,400	mgrimes
13	Betty	White	1,250	tswift
14	Shamrock	Black	1,400	[NULL]

	USERNAME	FNAME	LNAME
1	mgrimes	Marvin	Grimes
2	canderson	Christine	Anderson
3	tswift	Tina	Swift
4	jisbell	Jason	Isbell
5	ssimpson	Sam	Simpson

	NAME	COLOR	WEIGHT	OWNER	FNAME	LNAME
1	Sam	Brown	1,500	mgrimes	Marvin	Grimes
2	Erica	Yellow	920	canderson	Christine	Anderson
3	John	Grey	1,800	mgrimes	Marvin	Grimes
4	Trotty	Brown	1,300	mgrimes	Marvin	Grimes
5	Rio	Grey	1,700	tswift	Tina	Swift
6	Robin	Yellow	1,100	jisbell	Jason	Isbell
7	Katy	Brown	1,200	jisbell	Jason	Isbell
8	Pegasus	Brown	1,750	mgrimes	Marvin	Grimes
9	Sammy	Black	2,200	mgrimes	Marvin	Grimes
10	Pinky	Red	1,050	tswift	Tina	Swift
11	Hulk	Grey	2,050	mgrimes	Marvin	Grimes
12	Pat	White	1,400	mgrimes	Marvin	Grimes
13	Betty	White	1,250	tswift	Tina	Swift

Theta Join Operation

- Occurs infrequently in practical applications, Theta Joins do not involve equality but inequality conditions for the join condition involving attributes that share the same domain.



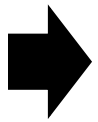
Theta Join Operation

- Performing this Theta Join results in this mess… Showing every horse with every customer that does NOT own it…

$$H \bowtie_{(H.Owner<>C.Username)} C$$

	NAME	COLOR	WEIGHT	OWNER
1	Sam	Brown	1,500	mgrimes
2	Erica	Yellow	920	canderson
3	John	Grey	1,800	mgrimes
4	Trotty	Brown	1,300	mgrimes
5	Rio	Grey	1,700	tswift
6	Robin	Yellow	1,100	jisbell
7	Katy	Brown	1,200	jisbell
8	Pegasus	Brown	1,750	mgrimes
9	Sammy	Black	2,200	mgrimes
10	Pinky	Red	1,050	tswift
11	Hulk	Grey	2,050	mgrimes
12	Pat	White	1,400	mgrimes
13	Betty	White	1,250	tswift
14	Shamrock	Black	1,400	[NULL]

	USERNAME	FNAME	LNAME
1	mgrimes	Marvin	Grimes
2	canderson	Christine	Anderson
3	tswift	Tina	Swift
4	jisbell	Jason	Isbell
5	ssimpson	Sam	Simpson



	NAME	COLOR	WEIGHT	OWNER	USERNAME	FNAME	LNAME
1	Erica	Yellow	920	canderson	mgrimes	Marvin	Grimes
2	Rio	Grey	1,700	tswift	mgrimes	Marvin	Grimes
3	Robin	Yellow	1,100	jisbell	mgrimes	Marvin	Grimes
4	Katy	Brown	1,200	jisbell	mgrimes	Marvin	Grimes
5	Pinky	Red	1,050	tswift	mgrimes	Marvin	Grimes
6	Betty	White	1,250	tswift	mgrimes	Marvin	Grimes
7	Sam	Brown	1,500	mgrimes	canderson	Christine	Anderson
8	John	Grey	1,800	mgrimes	canderson	Christine	Anderson
9	Trotty	Brown	1,300	mgrimes	canderson	Christine	Anderson
10	Rio	Grey	1,700	tswift	canderson	Christine	Anderson
11	Robin	Yellow	1,100	jisbell	canderson	Christine	Anderson
12	Katy	Brown	1,200	jisbell	canderson	Christine	Anderson
13	Pegasus	Brown	1,750	mgrimes	canderson	Christine	Anderson
14	Sammy	Black	2,200	mgrimes	canderson	Christine	Anderson
15	Pinky	Red	1,050	tswift	canderson	Christine	Anderson

(truncated – this goes on for 37 more tuples)

Outer Join Operation

- In Inner Join operations, tuples without a matching (or related) tuple are eliminated from the Join result.
- Tuples with null values in the joining attributes are also eliminated.
- A set of operations, called Outer Joins, can be used when we want to keep all the tuples in R, or those in S, or those in both relations in the result of the Join, whether or not they have matching tuples in the other relation.

The Three Outer Joins

- Left Outer Join (⟕ or]X| in the book)
  - The Left Outer Join operation keeps every tuple in the first or left relation R in R ⟕ S.
  - If no matching tuple is found in S, then the attributes of S in the join result are filled or "padded" with null values.

- Right Outer Join (⟖ or |X[in the book)
  - Right Outer Join keeps every tuple in the second or right relation S in the result R ⟖ S.
  - If no matching tuple is found in R, then the attributes of R in the join result are filled or "padded" with null values.

- Full Outer Join (⟗ or]X[in the book)
  - Full Outer Join keeps all tuples in both the left and right relations when no matching tuples are found, padding them with null values as needed.

Left Outer Join

- $H \bowtie_{(H.Owner=C.Username)} C$

	NAME	COLOR	WEIGHT	OWNER
1	Sam	Brown	1,500	mgrimes
2	Erica	Yellow	920	canderson
3	John	Grey	1,800	mgrimes
4	Trotty	Brown	1,300	mgrimes
5	Rio	Grey	1,700	tswift
6	Robin	Yellow	1,100	jisbell
7	Katy	Brown	1,200	jisbell
8	Pegasus	Brown	1,750	mgrimes
9	Sammy	Black	2,200	mgrimes
10	Pinky	Red	1,050	tswift
11	Hulk	Grey	2,050	mgrimes
12	Pat	White	1,400	mgrimes
13	Betty	White	1,250	tswift
14	Shamrock	Black	1,400	[NULL]

	USERNAME	FNAME	LNAME
1	mgrimes	Marvin	Grimes
2	canderson	Christine	Anderson
3	tswift	Tina	Swift
4	jisbell	Jason	Isbell
5	ssimpson	Sam	Simpson

	NAME	COLOR	WEIGHT	OWNER	USERNAME	FNAME	LNAME
1	Pat	White	1,400	mgrimes	mgrimes	Marvin	Grimes
2	Hulk	Grey	2,050	mgrimes	mgrimes	Marvin	Grimes
3	Sammy	Black	2,200	mgrimes	mgrimes	Marvin	Grimes
4	Pegasus	Brown	1,750	mgrimes	mgrimes	Marvin	Grimes
5	Trotty	Brown	1,300	mgrimes	mgrimes	Marvin	Grimes
6	John	Grey	1,800	mgrimes	mgrimes	Marvin	Grimes
7	Sam	Brown	1,500	mgrimes	mgrimes	Marvin	Grimes
8	Erica	Yellow	920	canderson	canderson	Christine	Anderson
9	Betty	White	1,250	tswift	tswift	Tina	Swift
10	Pinky	Red	1,050	tswift	tswift	Tina	Swift
11	Rio	Grey	1,700	tswift	tswift	Tina	Swift
12	Katy	Brown	1,200	jisbell	jisbell	Jason	Isbell
13	Robin	Yellow	1,100	jisbell	jisbell	Jason	Isbell
14	Shamrock	Black	1,400	[NULL]	[NULL]	[NULL]	[NULL]

Right Outer Join

- $H \bowtie_{(H.Owner=C.Username)} C$

	NAME	COLOR	WEIGHT	OWNER
1	Sam	Brown	1,500	mgrimes
2	Erica	Yellow	920	canderson
3	John	Grey	1,800	mgrimes
4	Trotty	Brown	1,300	mgrimes
5	Rio	Grey	1,700	tswift
6	Robin	Yellow	1,100	jisbell
7	Katy	Brown	1,200	jisbell
8	Pegasus	Brown	1,750	mgrimes
9	Sammy	Black	2,200	mgrimes
10	Pinky	Red	1,050	tswift
11	Hulk	Grey	2,050	mgrimes
12	Pat	White	1,400	mgrimes
13	Betty	White	1,250	tswift
14	Shamrock	Black	1,400	[NULL]

	USERNAME	FNAME	LNAME
1	mgrimes	Marvin	Grimes
2	canderson	Christine	Anderson
3	tswift	Tina	Swift
4	jisbell	Jason	Isbell
5	ssimpson	Sam	Simpson

	NAME	COLOR	WEIGHT	OWNER	USERNAME	FNAME	LNAME
1	Sam	Brown	1,500	mgrimes	mgrimes	Marvin	Grimes
2	Erica	Yellow	920	canderson	canderson	Christine	Anderson
3	John	Grey	1,800	mgrimes	mgrimes	Marvin	Grimes
4	Trotty	Brown	1,300	mgrimes	mgrimes	Marvin	Grimes
5	Rio	Grey	1,700	tswift	tswift	Tina	Swift
6	Robin	Yellow	1,100	jisbell	jisbell	Jason	Isbell
7	Katy	Brown	1,200	jisbell	jisbell	Jason	Isbell
8	Pegasus	Brown	1,750	mgrimes	mgrimes	Marvin	Grimes
9	Sammy	Black	2,200	mgrimes	mgrimes	Marvin	Grimes
10	Pinky	Red	1,050	tswift	tswift	Tina	Swift
11	Hulk	Grey	2,050	mgrimes	mgrimes	Marvin	Grimes
12	Pat	White	1,400	mgrimes	mgrimes	Marvin	Grimes
13	Betty	White	1,250	tswift	tswift	Tina	Swift
14	[NULL]	[NULL]	[NULL]	[NULL]	ssimpson	Sam	Simpson

Full Outer Join

- $H \bowtie_{(H.Owner=C.Username)} C$

	NAME	COLOR	WEIGHT	OWNER
1	Sam	Brown	1,500	mgrimes
2	Erica	Yellow	920	canderson
3	John	Grey	1,800	mgrimes
4	Trotty	Brown	1,300	mgrimes
5	Rio	Grey	1,700	tswift
6	Robin	Yellow	1,100	jisbell
7	Katy	Brown	1,200	jisbell
8	Pegasus	Brown	1,750	mgrimes
9	Sammy	Black	2,200	mgrimes
10	Pinky	Red	1,050	tswift
11	Hulk	Grey	2,050	mgrimes
12	Pat	White	1,400	mgrimes
13	Betty	White	1,250	tswift
14	Shamrock	Black	1,400	[NULL]

	USERNAME	FNAME	LNAME
1	mgrimes	Marvin	Grimes
2	canderson	Christine	Anderson
3	tswift	Tina	Swift
4	jisbell	Jason	Isbell
5	ssimpson	Sam	Simpson

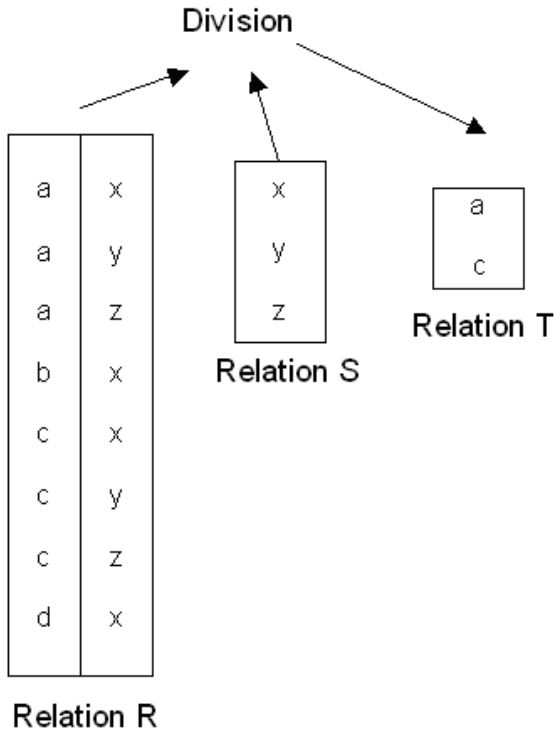
	NAME	COLOR	WEIGHT	OWNER	USERNAME	FNAME	LNAME
1	Sam	Brown	1,500	mgrimes	mgrimes	Marvin	Grimes
2	Erica	Yellow	920	canderson	canderson	Christine	Anderson
3	John	Grey	1,800	mgrimes	mgrimes	Marvin	Grimes
4	Trotty	Brown	1,300	mgrimes	mgrimes	Marvin	Grimes
5	Rio	Grey	1,700	tswift	tswift	Tina	Swift
6	Robin	Yellow	1,100	jisbell	jisbell	Jason	Isbell
7	Katy	Brown	1,200	jisbell	jisbell	Jason	Isbell
8	Pegasus	Brown	1,750	mgrimes	mgrimes	Marvin	Grimes
9	Sammy	Black	2,200	mgrimes	mgrimes	Marvin	Grimes
10	Pinky	Red	1,050	tswift	tswift	Tina	Swift
11	Hulk	Grey	2,050	mgrimes	mgrimes	Marvin	Grimes
12	Pat	White	1,400	mgrimes	mgrimes	Marvin	Grimes
13	Betty	White	1,250	tswift	tswift	Tina	Swift
14	Shamrock	Black	1,400	[NULL]	[NULL]	[NULL]	[NULL]
15	[NULL]	[NULL]	[NULL]	[NULL]	ssimpson	Sam	Simpson

The Divide Operation

- The DIVIDE operator is useful when there is a need to identify tuples in one relation that match all tuples in another relation.
  - Which person has account in all the banks of a particular city?
  - Which students have taken all the courses required to graduate?

- Division is effectively the opposite of the Cartesian Product

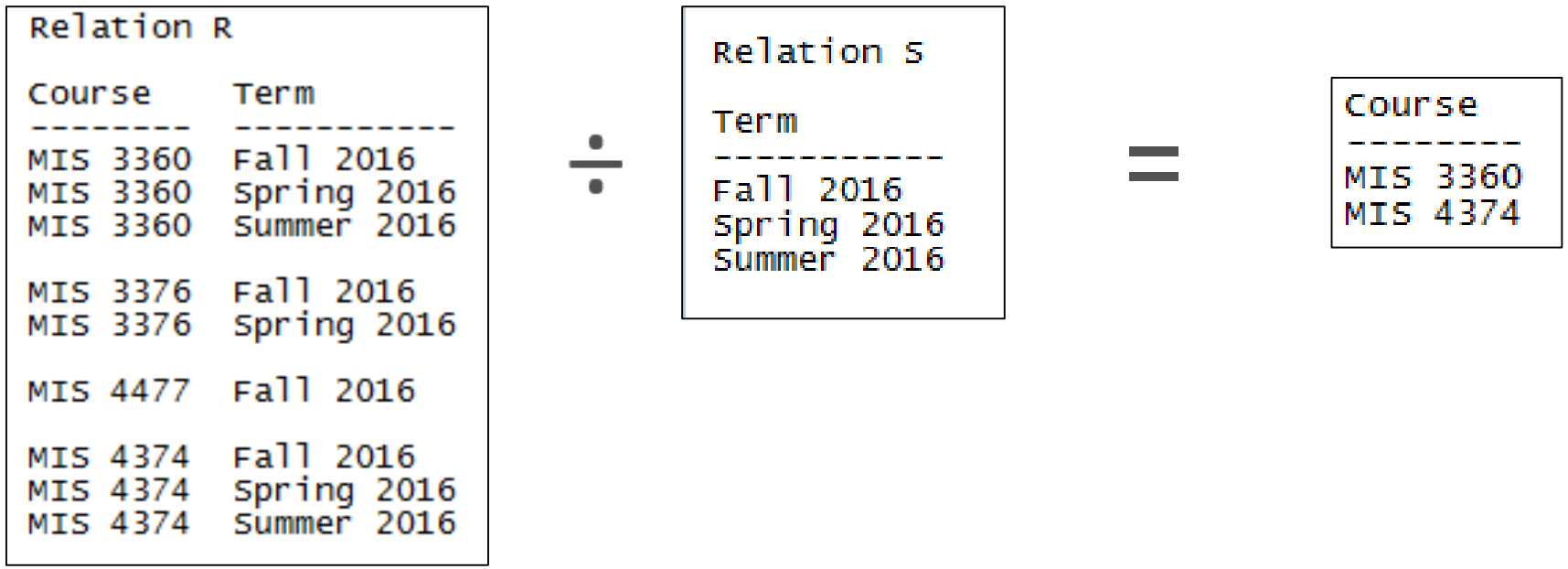
- Division is not supported in SQL
  - However, it can be represented using combinations of cross joins, difference, and IN operations.



The Divide Operation

- The DIVIDE operator is useful when there is a need to identify tuples in one relation that match all tuples in another relation.

$$R \div S$$



Break – 10 minutes

Module 12.1
Single table SQL operations

- SELECT-FROM-WHERE block
- Comparison and logical operators
- GROUP BY, HAVING, and ORDER BY clauses
- Aggregate functions

Structured Query Language

- SQL is a programming language used to facilitate implementation of relational algebra operations in a database.

- We will be converting some of the relational algebra expressions we did earlier into SQL code

- Focusing in this module on our unary operations:
  - Select: $\sigma_{\text{<section condition>}}R$
  - Project: $\pi_{\text{<attribute list>}}R$

General form of SELECT FROM WHERE

- SELECT <column list>
  FROM <table list>
  WHERE <condition> ← Optional

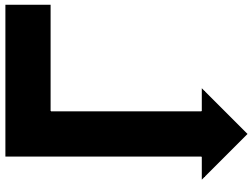
Foundational Knowledge for using SQL! All read operations will follow this basic form!

- <column list> is a list of column names (attributes) whose values are to be projected
- <table list> is a list of the table names (relations) required to process the query
- <condition> is a conditional (Boolean) expression that identifies the rows to be retrieved by the query.

Select Operation

- σ Horses
  - SELECT * FROM Horses;

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	John	Grey	No	M	1,800	mgrimes
4	Trotty	Brown	Yes	M	1,300	mgrimes
5	Rio	Grey	No	F	1,700	tswift
6	Robin	Yellow	No	M	1,100	jisbell
7	Katy	Brown	No	F	1,200	jisbell
8	Pegasus	Brown	No	M	1,750	mgrimes
9	Sammy	Black	Yes	M	2,200	mgrimes
10	Pinky	Red	No	M	1,050	tswift
11	Hulk	Grey	No	M	2,050	mgrimes
12	Pat	White	No	F	1,400	mgrimes
13	Betty	White	Yes	F	1,250	tswift
14	Shamrock	Black	No	M	1,400	[NULL]

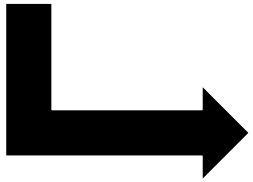


	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	John	Grey	No	M	1,800	mgrimes
4	Trotty	Brown	Yes	M	1,300	mgrimes
5	Rio	Grey	No	F	1,700	tswift
6	Robin	Yellow	No	M	1,100	jisbell
7	Katy	Brown	No	F	1,200	jisbell
8	Pegasus	Brown	No	M	1,750	mgrimes
9	Sammy	Black	Yes	M	2,200	mgrimes
10	Pinky	Red	No	M	1,050	tswift
11	Hulk	Grey	No	M	2,050	mgrimes
12	Pat	White	No	F	1,400	mgrimes
13	Betty	White	Yes	F	1,250	tswift
14	Shamrock	Black	No	M	1,400	[NULL]

Select Operation with criteria

- $\sigma_{(Sex = 'F')}$ Horses
  - SELECT * FROM Horses WHERE Sex='F';

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	John	Grey	No	M	1,800	mgrimes
4	Trotty	Brown	Yes	M	1,300	mgrimes
5	Rio	Grey	No	F	1,700	tswift
6	Robin	Yellow	No	M	1,100	jisbell
7	Katy	Brown	No	F	1,200	jisbell
8	Pegasus	Brown	No	M	1,750	mgrimes
9	Sammy	Black	Yes	M	2,200	mgrimes
10	Pinky	Red	No	M	1,050	tswift
11	Hulk	Grey	No	M	2,050	mgrimes
12	Pat	White	No	F	1,400	mgrimes
13	Betty	White	Yes	F	1,250	tswift
14	Shamrock	Black	No	M	1,400	[NULL]



	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	Rio	Grey	No	F	1,700	tswift
4	Katy	Brown	No	F	1,200	jisbell
5	Pat	White	No	F	1,400	mgrimes
6	Betty	White	Yes	F	1,250	tswift

Select Operation specifying all attributes

- Instead of *, you can specify all attributes if you like:
  - SELECT name, color, spots, sex, weight, owner FROM Horses WHERE Sex='F';

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	John	Grey	No	M	1,800	mgrimes
4	Trotty	Brown	Yes	M	1,300	mgrimes
5	Rio	Grey	No	F	1,700	tswift
6	Robin	Yellow	No	M	1,100	jisbell
7	Katy	Brown	No	F	1,200	jisbell
8	Pegasus	Brown	No	M	1,750	mgrimes
9	Sammy	Black	Yes	M	2,200	mgrimes
10	Pinky	Red	No	M	1,050	tswift
11	Hulk	Grey	No	M	2,050	mgrimes
12	Pat	White	No	F	1,400	mgrimes
13	Betty	White	Yes	F	1,250	tswift
14	Shamrock	Black	No	M	1,400	[NULL]

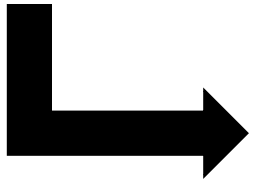
(exact same result as previous slide)

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	Rio	Grey	No	F	1,700	tswift
4	Katy	Brown	No	F	1,200	jisbell
5	Pat	White	No	F	1,400	mgrimes
6	Betty	White	Yes	F	1,250	tswift

Select Operation with multiple criteria

- $\sigma_{(Sex = 'F' \text{ AND } Spots='No')}$ Horses
  - SELECT * FROM Horses WHERE Sex='F' AND Spots='No';

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	John	Grey	No	M	1,800	mgrimes
4	Trotty	Brown	Yes	M	1,300	mgrimes
5	Rio	Grey	No	F	1,700	tswift
6	Robin	Yellow	No	M	1,100	jisbell
7	Katy	Brown	No	F	1,200	jisbell
8	Pegasus	Brown	No	M	1,750	mgrimes
9	Sammy	Black	Yes	M	2,200	mgrimes
10	Pinky	Red	No	M	1,050	tswift
11	Hulk	Grey	No	M	2,050	mgrimes
12	Pat	White	No	F	1,400	mgrimes
13	Betty	White	Yes	F	1,250	tswift
14	Shamrock	Black	No	M	1,400	[NULL]



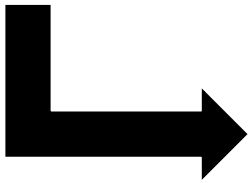
	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Rio	Grey	No	F	1,700	tswift
3	Katy	Brown	No	F	1,200	jisbell
4	Pat	White	No	F	1,400	mgrimes

Select Operation with multiple criteria

- $\sigma_{(Color='Brown' \text{ OR } Color='White')}$ Horses
  - SELECT * FROM Horses WHERE Color='Brown' OR Color='White';

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	John	Grey	No	M	1,800	mgrimes
4	Trotty	Brown	Yes	M	1,300	mgrimes
5	Rio	Grey	No	F	1,700	tswift
6	Robin	Yellow	No	M	1,100	jisbell
7	Katy	Brown	No	F	1,200	jisbell
8	Pegasus	Brown	No	M	1,750	mgrimes
9	Sammy	Black	Yes	M	2,200	mgrimes
10	Pinky	Red	No	M	1,050	tswift
11	Hulk	Grey	No	M	2,050	mgrimes
12	Pat	White	No	F	1,400	mgrimes
13	Betty	White	Yes	F	1,250	tswift
14	Shamrock	Black	No	M	1,400	[NULL]

Would Color='Brown' AND Color='White' ever evaluate to true?



	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Trotty	Brown	Yes	M	1,300	mgrimes
3	Katy	Brown	No	F	1,200	jisbell
4	Pegasus	Brown	No	M	1,750	mgrimes
5	Pat	White	No	F	1,400	mgrimes
6	Betty	White	Yes	F	1,250	tswift

Be careful with order of operations

SELECT * FROM Horses WHERE (Color='Brown' OR Color='White') AND Spots='No';

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Katy	Brown	No	F	1,200	jisbell
3	Pegasus	Brown	No	M	1,750	mgrimes
4	Pat	White	No	F	1,400	mgrimes

SELECT * FROM Horses WHERE Color='Brown' OR (Color='White' AND Spots='No');

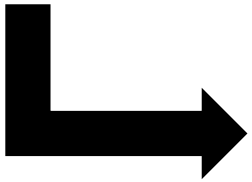
	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Trotty	Brown	Yes	M	1,300	mgrimes
3	Katy	Brown	No	F	1,200	jisbell
4	Pegasus	Brown	No	M	1,750	mgrimes
5	Pat	White	No	F	1,400	mgrimes

Order of operations dictates that AND is processed first, so without parentheses you should get the second result? … Is that what you wanted?

Projection

- $\pi_{(Name,\ Color,\ Owner)}Horses$
  - SELECT name, color, owner FROM horses;

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	John	Grey	No	M	1,800	mgrimes
4	Trotty	Brown	Yes	M	1,300	mgrimes
5	Rio	Grey	No	F	1,700	tswift
6	Robin	Yellow	No	M	1,100	jisbell
7	Katy	Brown	No	F	1,200	jisbell
8	Pegasus	Brown	No	M	1,750	mgrimes
9	Sammy	Black	Yes	M	2,200	mgrimes
10	Pinky	Red	No	M	1,050	tswift
11	Hulk	Grey	No	M	2,050	mgrimes
12	Pat	White	No	F	1,400	mgrimes
13	Betty	White	Yes	F	1,250	tswift
14	Shamrock	Black	No	M	1,400	[NULL]

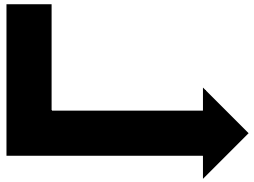


	NAME	COLOR	OWNER
1	Sam	Brown	mgrimes
2	Erica	Yellow	canderson
3	John	Grey	mgrimes
4	Trotty	Brown	mgrimes
5	Rio	Grey	tswift
6	Robin	Yellow	jisbell
7	Katy	Brown	jisbell
8	Pegasus	Brown	mgrimes
9	Sammy	Black	mgrimes
10	Pinky	Red	tswift
11	Hulk	Grey	mgrimes
12	Pat	White	mgrimes
13	Betty	White	tswift
14	Shamrock	Black	[NULL]

Projection and Selection

- $\pi_{(Name,\ Color,\ Owner)}\sigma_{(Color='Brown'\ OR\ Color='White')}$ Horses
  - SELECT name, color, owner FROM Horses WHERE Color='Brown' OR Color='White';

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	John	Grey	No	M	1,800	mgrimes
4	Trotty	Brown	Yes	M	1,300	mgrimes
5	Rio	Grey	No	F	1,700	tswift
6	Robin	Yellow	No	M	1,100	jisbell
7	Katy	Brown	No	F	1,200	jisbell
8	Pegasus	Brown	No	M	1,750	mgrimes
9	Sammy	Black	Yes	M	2,200	mgrimes
10	Pinky	Red	No	M	1,050	tswift
11	Hulk	Grey	No	M	2,050	mgrimes
12	Pat	White	No	F	1,400	mgrimes
13	Betty	White	Yes	F	1,250	tswift
14	Shamrock	Black	No	M	1,400	[NULL]



	NAME	COLOR	OWNER
1	Sam	Brown	mgrimes
2	Trotty	Brown	mgrimes
3	Katy	Brown	jisbell
4	Pegasus	Brown	mgrimes
5	Pat	White	mgrimes
6	Betty	White	tswift

An instance where RA and SQL do not align perfectly

- The expression $\pi_{(Owner)}$Horses
  - Creates a relation that does not have duplicate tuples

- SELECT owner FROM Horses;
  - Will return duplicate rows ☹

	ABC OWNER
1	mgrimes
2	canderson
3	mgrimes
4	mgrimes
5	tswift
6	jisbell
7	jisbell
8	mgrimes
9	mgrimes
10	tswift
11	mgrimes
12	mgrimes
13	tswift
14	[NULL]

- SELECT DISTINCT owner FROM Horses;
  - Faithfully reproduces the results of the relational algebra expression

	ABC OWNER
1	canderson
2	jisbell
3	mgrimes
4	tswift
5	[NULL]

- This is fundamentally due to the nuanced difference in a table, relation, and entity…

Optional Add-ons To SELECT Clause

- GROUP BY *expression*
  - Groups rows with a common value of a specified attribute
  - Typically used with **aggregate functions** (next slide)

- HAVING *condition*
  - filters the groups subject to some condition
  - HAVING is very similar to WHERE, but is used on the results of aggregate functions

- ORDER BY *column name(s)*
  - specifies the order of the output.

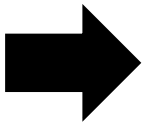
Relational Algebra Aggregate Functions

- One final relational algebra operation: Aggregate Functions
- Symbol is $\mathfrak{F}$ (“Script F”)
- General form is: $_{\text{<GROUPING ATTRIBUTE(S)>}}\mathfrak{F}_{\text{<Function List>}}$ R
- Common functions include:
  - COUNT(), AVG(), MIN(), MAX(), SUM()
- Aggregate functions ignore **null** values!
- All attributes you wish to see in the result MUST BE listed as a grouping attribute
- Upcoming examples for AVG and Count, but they all work basically the same way

Average function

- $_{\text{Sex}}\Im_{\text{AVG Weight}}$ Horses
  - SELECT Sex, AVG(weight) FROM Horses GROUP BY Sex;

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	John	Grey	No	M	1,800	mgrimes
4	Trotty	Brown	Yes	M	1,300	mgrimes
5	Rio	Grey	No	F	1,700	tswift
6	Robin	Yellow	No	M	1,100	jisbell
7	Katy	Brown	No	F	1,200	jisbell
8	Pegasus	Brown	No	M	1,750	mgrimes
9	Sammy	Black	Yes	M	2,200	mgrimes
10	Pinky	Red	No	M	1,050	tswift
11	Hulk	Grey	No	M	2,050	mgrimes
12	Pat	White	No	F	1,400	mgrimes
13	Betty	White	Yes	F	1,250	tswift
14	Shamrock	Black	No	M	1,400	[NULL]

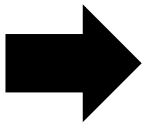


	SEX	AVG(WEIGHT)
1	M	1,581.25
2	F	1,328.3333333333

Count function

- $_{\text{Color}}\Im_{\text{COUNT}}$ Horses
  - SELECT Color, COUNT(*) FROM Horses GROUP BY Color;

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	John	Grey	No	M	1,800	mgrimes
4	Trotty	Brown	Yes	M	1,300	mgrimes
5	Rio	Grey	No	F	1,700	tswift
6	Robin	Yellow	No	M	1,100	jisbell
7	Katy	Brown	No	F	1,200	jisbell
8	Pegasus	Brown	No	M	1,750	mgrimes
9	Sammy	Black	Yes	M	2,200	mgrimes
10	Pinky	Red	No	M	1,050	tswift
11	Hulk	Grey	No	M	2,050	mgrimes
12	Pat	White	No	F	1,400	mgrimes
13	Betty	White	Yes	F	1,250	tswift
14	Shamrock	Black	No	M	1,400	[NULL]

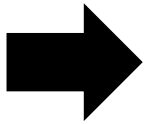


	COLOR	COUNT(*)
1	White	2
2	Yellow	2
3	Grey	3
4	Brown	4
5	Black	2
6	Red	1

Count function with selection

- $\sigma_{(Count(*)>2)}(_{Color}\Im_{Count}$ Horses)
  - SELECT color, Count(*) FROM Horses GROUP BY color HAVING Count(*) > 2;

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	John	Grey	No	M	1,800	mgrimes
4	Trotty	Brown	Yes	M	1,300	mgrimes
5	Rio	Grey	No	F	1,700	tswift
6	Robin	Yellow	No	M	1,100	jisbell
7	Katy	Brown	No	F	1,200	jisbell
8	Pegasus	Brown	No	M	1,750	mgrimes
9	Sammy	Black	Yes	M	2,200	mgrimes
10	Pinky	Red	No	M	1,050	tswift
11	Hulk	Grey	No	M	2,050	mgrimes
12	Pat	White	No	F	1,400	mgrimes
13	Betty	White	Yes	F	1,250	tswift
14	Shamrock	Black	No	M	1,400	[NULL]

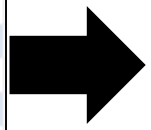


	COLOR	COUNT(*)
1	Grey	3
2	Brown	4

Count function – multiple grouping (and ordering)

- Color, Sex $\Im$ COUNT Horses
  - SELECT Color, Sex, COUNT(*) FROM Horses
    GROUP BY Color, Sex
    ORDER BY Color, Sex;

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	John	Grey	No	M	1,800	mgrimes
4	Trotty	Brown	Yes	M	1,300	mgrimes
5	Rio	Grey	No	F	1,700	tswift
6	Robin	Yellow	No	M	1,100	jisbell
7	Katy	Brown	No	F	1,200	jisbell
8	Pegasus	Brown	No	M	1,750	mgrimes
9	Sammy	Black	Yes	M	2,200	mgrimes
10	Pinky	Red	No	M	1,050	tswift
11	Hulk	Grey	No	M	2,050	mgrimes
12	Pat	White	No	F	1,400	mgrimes
13	Betty	White	Yes	F	1,250	tswift
14	Shamrock	Black	No	M	1,400	[NULL]



	COLOR	SEX	COUNT(*)
1	Black	M	2
2	Brown	F	2
3	Brown	M	2
4	Grey	F	1
5	Grey	M	2
6	Red	M	1
7	White	F	2
8	Yellow	F	1
9	Yellow	M	1

Using AS to create aliases

- Can be used for any attribute, but is particularly useful for aggregate attributes, which the DBMS will automatically assign hard to use names to.
  - SELECT Color, Sex, COUNT(*) AS “Number of Horses” FROM Horses
    GROUP BY Color, Sex
    ORDER BY Color, Sex;

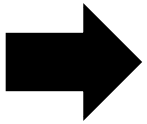
	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	John	Grey	No	M	1,800	mgrimes
4	Trotty	Brown	Yes	M	1,300	mgrimes
5	Rio	Grey	No	F	1,700	tswift
6	Robin	Yellow	No	M	1,100	jisbell
7	Katy	Brown	No	F	1,200	jisbell
8	Pegasus	Brown	No	M	1,750	mgrimes
9	Sammy	Black	Yes	M	2,200	mgrimes
10	Pinky	Red	No	M	1,050	tswift
11	Hulk	Grey	No	M	2,050	mgrimes
12	Pat	White	No	F	1,400	mgrimes
13	Betty	White	Yes	F	1,250	tswift
14	Shamrock	Black	No	M	1,400	[NULL]

	COLOR	SEX	Number of Horses
1	Black	M	2
2	Brown	F	2
3	Brown	M	2
4	Grey	F	1
5	Grey	M	2
6	Red	M	1
7	White	F	2
8	Yellow	F	1
9	Yellow	M	1

Count function – no grouping

- $\Im_{COUNT}$ Horses
  - SELECT COUNT(*) FROM horses;

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	John	Grey	No	M	1,800	mgrimes
4	Trotty	Brown	Yes	M	1,300	mgrimes
5	Rio	Grey	No	F	1,700	tswift
6	Robin	Yellow	No	M	1,100	jisbell
7	Katy	Brown	No	F	1,200	jisbell
8	Pegasus	Brown	No	M	1,750	mgrimes
9	Sammy	Black	Yes	M	2,200	mgrimes
10	Pinky	Red	No	M	1,050	tswift
11	Hulk	Grey	No	M	2,050	mgrimes
12	Pat	White	No	F	1,400	mgrimes
13	Betty	White	Yes	F	1,250	tswift
14	Shamrock	Black	No	M	1,400	[NULL]

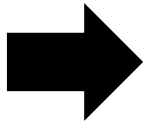


	COUNT(*)
1	14

Count function – no grouping, NULL values

- Aggregate functions (like count) ignore NULL values
- In this previous example we were counting tuples. If we count values in an attribute we get a different result
- $\Im_{\text{COUNT(Owner)}}$ Horses
  - SELECT COUNT(owner) FROM Horses;

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	John	Grey	No	M	1,800	mgrimes
4	Trotty	Brown	Yes	M	1,300	mgrimes
5	Rio	Grey	No	F	1,700	tswift
6	Robin	Yellow	No	M	1,100	jisbell
7	Katy	Brown	No	F	1,200	jisbell
8	Pegasus	Brown	No	M	1,750	mgrimes
9	Sammy	Black	Yes	M	2,200	mgrimes
10	Pinky	Red	No	M	1,050	tswift
11	Hulk	Grey	No	M	2,050	mgrimes
12	Pat	White	No	F	1,400	mgrimes
13	Betty	White	Yes	F	1,250	tswift
14	Shamrock	Black	No	M	1,400	[NULL]

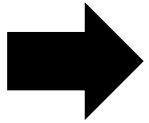


	COUNT(OWNER)
1	13

Count function – no grouping, no NULL values

- Aggregate functions (like count) ignore NULL values
- In this previous example we were counting tuples. If we count values in an attribute we get a different result
- $\Im_{\text{COUNT(Name)}}$ Horses
  - SELECT COUNT(Name) FROM Horses;

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	John	Grey	No	M	1,800	mgrimes
4	Trotty	Brown	Yes	M	1,300	mgrimes
5	Rio	Grey	No	F	1,700	tswift
6	Robin	Yellow	No	M	1,100	jisbell
7	Katy	Brown	No	F	1,200	jisbell
8	Pegasus	Brown	No	M	1,750	mgrimes
9	Sammy	Black	Yes	M	2,200	mgrimes
10	Pinky	Red	No	M	1,050	tswift
11	Hulk	Grey	No	M	2,050	mgrimes
12	Pat	White	No	F	1,400	mgrimes
13	Betty	White	Yes	F	1,250	tswift
14	Shamrock	Black	No	M	1,400	[NULL]



	COUNT(NAME)
1	14

Ways to be more efficient with selections

- IN and BETWEEN can be used as comparison operators.
- IN is evaluated in the context of being equal to any member of a set of values
- A nested query can follow the IN/NOT IN operator
- Between is evaluated as a pair of ≥ and ≤ operators

Using IN

- For example, instead of this:

```
SELECT * FROM Horses
WHERE spots = 'No'
AND (color = 'Brown' OR color = 'White');
```

- We can do this:

```
SELECT * FROM Horses
WHERE spots = 'No'
AND color IN ('Brown', 'White');
```

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	John	Grey	No	M	1,800	mgrimes
4	Trotty	Brown	Yes	M	1,300	mgrimes
5	Rio	Grey	No	F	1,700	tswift
6	Robin	Yellow	No	M	1,100	jisbell
7	Katy	Brown	No	F	1,200	jisbell
8	Pegasus	Brown	No	M	1,750	mgrimes
9	Sammy	Black	Yes	M	2,200	mgrimes
10	Pinky	Red	No	M	1,050	tswift
11	Hulk	Grey	No	M	2,050	mgrimes
12	Pat	White	No	F	1,400	mgrimes
13	Betty	White	Yes	F	1,250	tswift
14	Shamrock	Black	No	M	1,400	[NULL]

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Katy	Brown	No	F	1,200	jisbell
3	Pegasus	Brown	No	M	1,750	mgrimes
4	Pat	White	No	F	1,400	mgrimes

Using NOT IN

- This gets even more useful when negating things, which can get a little confusing…
  - If we want all horses that are NOT Brown or White…
- Instead of this:

```
SELECT * FROM horses
WHERE color <> 'Brown' AND color <> 'White';
```

- We can do this:

```
SELECT * FROM horses
WHERE color NOT IN ('Brown', 'White');
```

Negation gets confusing because it's not Brown OR White, it's now Brown AND White… If you accidentally said
`color <> 'Brown' OR color <> 'White'`
Then all tuples would be returned!

That's why NOT IN is so much simpler (in my opinion…)

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	John	Grey	No	M	1,800	mgrimes
4	Trotty	Brown	Yes	M	1,300	mgrimes
5	Rio	Grey	No	F	1,700	tswift
6	Robin	Yellow	No	M	1,100	jisbell
7	Katy	Brown	No	F	1,200	jisbell
8	Pegasus	Brown	No	M	1,750	mgrimes
9	Sammy	Black	Yes	M	2,200	mgrimes
10	Pinky	Red	No	M	1,050	tswift
11	Hulk	Grey	No	M	2,050	mgrimes
12	Pat	White	No	F	1,400	mgrimes
13	Betty	White	Yes	F	1,250	tswift
14	Shamrock	Black	No	M	1,400	[NULL]

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Erica	Yellow	Yes	F	920	canderson
2	John	Grey	No	M	1,800	mgrimes
3	Rio	Grey	No	F	1,700	tswift
4	Robin	Yellow	No	M	1,100	jisbell
5	Sammy	Black	Yes	M	2,200	mgrimes
6	Pinky	Red	No	M	1,050	tswift
7	Hulk	Grey	No	M	2,050	mgrimes
8	Shamrock	Black	No	M	1,400	[NULL]

Using IN with a subquery

- SELECT * FROM horses
  WHERE name IN (SELECT ord_Horsename FROM orders);

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	John	Grey	No	M	1,800	mgrimes
4	Trotty	Brown	Yes	M	1,300	mgrimes
5	Rio	Grey	No	F	1,700	tswift
6	Robin	Yellow	No	M	1,100	jisbell
7	Katy	Brown	No	F	1,200	jisbell
8	Pegasus	Brown	No	M	1,750	mgrimes
9	Sammy	Black	Yes	M	2,200	mgrimes
10	Pinky	Red	No	M	1,050	tswift
11	Hulk	Grey	No	M	2,050	mgrimes
12	Pat	White	No	F	1,400	mgrimes
13	Betty	White	Yes	F	1,250	tswift
14	Shamrock	Black	No	M	1,400	[NULL]

	RXNUM	ORD_MEDCODE	ORD_HORSENAME	DOSE	FREQUENCY
1	108	Dexa	Betty	2	1
2	107	Dexa	Shamrock	2	1
3	105	Doxy	Trotty	1	3
4	110	Enro	Erica	3	2
5	101	Flux	Sam	1	2
6	104	Iver	Trotty	2	1
7	103	Iver	John	2	1
8	102	Pen	Sam	1	1
9	106	Pres	Shamrock	1	1
10	109	Xyl	Hulk	1	1

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	John	Grey	No	M	1,800	mgrimes
3	Trotty	Brown	Yes	M	1,300	mgrimes
4	Shamrock	Black	No	M	1,400	[NULL]
5	Betty	White	Yes	F	1,250	tswift
6	Hulk	Grey	No	M	2,050	mgrimes
7	Erica	Yellow	Yes	F	920	canderson

…So this query is effectively the same as:

```
SELECT * FROM horses WHERE name IN ('Betty', 'Shamrock', 'Trotty',
'Erica', 'Sam', 'Trotty', 'John', 'Sam', 'Shamrock', 'Hulk');
```

But even better, since list of names is created dynamically based on the subquery!

What must be true about ORD_Horsename?
What is the cardinality of this relationship?
Which relation is the parent?

BETWEEN and NOT BETWEEN

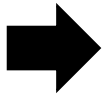
- For example, instead of this:

SELECT * FROM horses WHERE weight >= 1000 AND weight <= 1500;

- We can do this:

SELECT * FROM horses WHERE weight BETWEEN 1000 AND 1500;

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	John	Grey	No	M	1,800	mgrimes
4	Trotty	Brown	Yes	M	1,300	mgrimes
5	Rio	Grey	No	F	1,700	tswift
6	Robin	Yellow	No	M	1,100	jisbell
7	Katy	Brown	No	F	1,200	jisbell
8	Pegasus	Brown	No	M	1,750	mgrimes
9	Sammy	Black	Yes	M	2,200	mgrimes
10	Pinky	Red	No	M	1,050	tswift
11	Hulk	Grey	No	M	2,050	mgrimes
12	Pat	White	No	F	1,400	mgrimes
13	Betty	White	Yes	F	1,250	tswift
14	Shamrock	Black	No	M	1,400	[NULL]

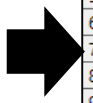


	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Trotty	Brown	Yes	M	1,300	mgrimes
3	Robin	Yellow	No	M	1,100	jisbell
4	Katy	Brown	No	F	1,200	jisbell
5	Pinky	Red	No	M	1,050	tswift
6	Pat	White	No	F	1,400	mgrimes
7	Betty	White	Yes	F	1,250	tswift
8	Shamrock	Black	No	M	1,400	[NULL]

ORDER BY

- Used to change the order in which tuples are returned
  - SELECT * FROM horses ORDER BY name;

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	John	Grey	No	M	1,800	mgrimes
4	Trotty	Brown	Yes	M	1,300	mgrimes
5	Rio	Grey	No	F	1,700	tswift
6	Robin	Yellow	No	M	1,100	jisbell
7	Katy	Brown	No	F	1,200	jisbell
8	Pegasus	Brown	No	M	1,750	mgrimes
9	Sammy	Black	Yes	M	2,200	mgrimes
10	Pinky	Red	No	M	1,050	tswift
11	Hulk	Grey	No	M	2,050	mgrimes
12	Pat	White	No	F	1,400	mgrimes
13	Betty	White	Yes	F	1,250	tswift
14	Shamrock	Black	No	M	1,400	[NULL]



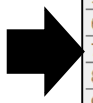
	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Betty	White	Yes	F	1,250	tswift
2	Erica	Yellow	Yes	F	920	canderson
3	Hulk	Grey	No	M	2,050	mgrimes
4	John	Grey	No	M	1,800	mgrimes
5	Katy	Brown	No	F	1,200	jisbell
6	Pat	White	No	F	1,400	mgrimes
7	Pegasus	Brown	No	M	1,750	mgrimes
8	Pinky	Red	No	M	1,050	tswift
9	Rio	Grey	No	F	1,700	tswift
10	Robin	Yellow	No	M	1,100	jisbell
11	Sam	Brown	No	F	1,500	mgrimes
12	Sammy	Black	Yes	M	2,200	mgrimes
13	Shamrock	Black	No	M	1,400	[NULL]
14	Trotty	Brown	Yes	M	1,300	mgrimes

- By default is ascending (ASC), use the keyword DESC to reverse the order

ORDER BY

- Used to change the order in which tuples are returned
  - SELECT * FROM horses ORDER BY weight DESC;

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	John	Grey	No	M	1,800	mgrimes
4	Trotty	Brown	Yes	M	1,300	mgrimes
5	Rio	Grey	No	F	1,700	tswift
6	Robin	Yellow	No	M	1,100	jisbell
7	Katy	Brown	No	F	1,200	jisbell
8	Pegasus	Brown	No	M	1,750	mgrimes
9	Sammy	Black	Yes	M	2,200	mgrimes
10	Pinky	Red	No	M	1,050	tswift
11	Hulk	Grey	No	M	2,050	mgrimes
12	Pat	White	No	F	1,400	mgrimes
13	Betty	White	Yes	F	1,250	tswift
14	Shamrock	Black	No	M	1,400	[NULL]



	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sammy	Black	Yes	M	2,200	mgrimes
2	Hulk	Grey	No	M	2,050	mgrimes
3	John	Grey	No	M	1,800	mgrimes
4	Pegasus	Brown	No	M	1,750	mgrimes
5	Rio	Grey	No	F	1,700	tswift
6	Sam	Brown	No	F	1,500	mgrimes
7	Pat	White	No	F	1,400	mgrimes
8	Shamrock	Black	No	M	1,400	[NULL]
9	Trotty	Brown	Yes	M	1,300	mgrimes
10	Betty	White	Yes	F	1,250	tswift
11	Katy	Brown	No	F	1,200	jisbell
12	Robin	Yellow	No	M	1,100	jisbell
13	Pinky	Red	No	M	1,050	tswift
14	Erica	Yellow	Yes	F	920	canderson

- By default is ascending (ASC), use the keyword DESC to reverse the order

Pattern Matching

- Use the LIKE operator works in conjunction with the two wildcard characters:
  - % - Any number of characters
  - _ - Exactly one character

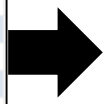
- Note: In access we use:
  - * instead of % for any number of characters
  - ? Instead of _ for exactly one character

The LIKE Operator

Select all attributes about a horse where horse's name starts with S:

```
SELECT * FROM Horses WHERE Name LIKE 'S%';
```

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	John	Grey	No	M	1,800	mgrimes
4	Trotty	Brown	Yes	M	1,300	mgrimes
5	Rio	Grey	No	F	1,700	tswift
6	Robin	Yellow	No	M	1,100	jisbell
7	Katy	Brown	No	F	1,200	jisbell
8	Pegasus	Brown	No	M	1,750	mgrimes
9	Sammy	Black	Yes	M	2,200	mgrimes
10	Pinky	Red	No	M	1,050	tswift
11	Hulk	Grey	No	M	2,050	mgrimes
12	Pat	White	No	F	1,400	mgrimes
13	Betty	White	Yes	F	1,250	tswift
14	Shamrock	Black	No	M	1,400	[NULL]



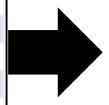
	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Sammy	Black	Yes	M	2,200	mgrimes
3	Shamrock	Black	No	M	1,400	[NULL]

The LIKE Operator

Select all attributes about a horse where the second character of the name of the horse is i:

```
SELECT * FROM Horses WHERE Name LIKE '_i%';
```

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	John	Grey	No	M	1,800	mgrimes
4	Trotty	Brown	Yes	M	1,300	mgrimes
5	Rio	Grey	No	F	1,700	tswift
6	Robin	Yellow	No	M	1,100	jisbell
7	Katy	Brown	No	F	1,200	jisbell
8	Pegasus	Brown	No	M	1,750	mgrimes
9	Sammy	Black	Yes	M	2,200	mgrimes
10	Pinky	Red	No	M	1,050	tswift
11	Hulk	Grey	No	M	2,050	mgrimes
12	Pat	White	No	F	1,400	mgrimes
13	Betty	White	Yes	F	1,250	tswift
14	Shamrock	Black	No	M	1,400	[NULL]



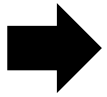
	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Rio	Grey	No	F	1,700	tswift
2	Pinky	Red	No	M	1,050	tswift

The LIKE Operator

Select all attributes about a course where the name of the horse contains “in”:

```
SELECT * FROM Horses WHERE Name LIKE '%in%';
```

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	John	Grey	No	M	1,800	mgrimes
4	Trotty	Brown	Yes	M	1,300	mgrimes
5	Rio	Grey	No	F	1,700	tswift
6	Robin	Yellow	No	M	1,100	jisbell
7	Katy	Brown	No	F	1,200	jisbell
8	Pegasus	Brown	No	M	1,750	mgrimes
9	Sammy	Black	Yes	M	2,200	mgrimes
10	Pinky	Red	No	M	1,050	tswift
11	Hulk	Grey	No	M	2,050	mgrimes
12	Pat	White	No	F	1,400	mgrimes
13	Betty	White	Yes	F	1,250	tswift
14	Shamrock	Black	No	M	1,400	[NULL]



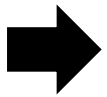
	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Robin	Yellow	No	M	1,100	jisbell
2	Pinky	Red	No	M	1,050	tswift

IS NULL and IS NOT NULL

- NULL is the absence of data, so nothing is EVER “equal” to NULL – therefore we have to use the “IS” operator

SELECT * FROM horses WHERE owner IS NULL

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	John	Grey	No	M	1,800	mgrimes
4	Trotty	Brown	Yes	M	1,300	mgrimes
5	Rio	Grey	No	F	1,700	tswift
6	Robin	Yellow	No	M	1,100	jisbell
7	Katy	Brown	No	F	1,200	jisbell
8	Pegasus	Brown	No	M	1,750	mgrimes
9	Sammy	Black	Yes	M	2,200	mgrimes
10	Pinky	Red	No	M	1,050	tswift
11	Hulk	Grey	No	M	2,050	mgrimes
12	Pat	White	No	F	1,400	mgrimes
13	Betty	White	Yes	F	1,250	tswift
14	Shamrock	Black	No	M	1,400	[NULL]

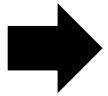


	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Shamrock	Black	No	M	1,400	[NULL]

IS NULL and IS NOT NULL

SELECT * FROM horses WHERE owner IS NOT NULL

	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	John	Grey	No	M	1,800	mgrimes
4	Trotty	Brown	Yes	M	1,300	mgrimes
5	Rio	Grey	No	F	1,700	tswift
6	Robin	Yellow	No	M	1,100	jisbell
7	Katy	Brown	No	F	1,200	jisbell
8	Pegasus	Brown	No	M	1,750	mgrimes
9	Sammy	Black	Yes	M	2,200	mgrimes
10	Pinky	Red	No	M	1,050	tswift
11	Hulk	Grey	No	M	2,050	mgrimes
12	Pat	White	No	F	1,400	mgrimes
13	Betty	White	Yes	F	1,250	tswift
14	Shamrock	Black	No	M	1,400	[NULL]



	NAME	COLOR	SPOTS	SEX	WEIGHT	OWNER
1	Sam	Brown	No	F	1,500	mgrimes
2	Erica	Yellow	Yes	F	920	canderson
3	John	Grey	No	M	1,800	mgrimes
4	Trotty	Brown	Yes	M	1,300	mgrimes
5	Rio	Grey	No	F	1,700	tswift
6	Robin	Yellow	No	M	1,100	jisbell
7	Katy	Brown	No	F	1,200	jisbell
8	Pegasus	Brown	No	M	1,750	mgrimes
9	Sammy	Black	Yes	M	2,200	mgrimes
10	Pinky	Red	No	M	1,050	tswift
11	Hulk	Grey	No	M	2,050	mgrimes
12	Pat	White	No	F	1,400	mgrimes
13	Betty	White	Yes	F	1,250	tswift

These will always return nothing – not good queries!

- SELECT * FROM horses WHERE owner <> NULL;
- SELECT * FROM horses WHERE owner = NULL;

SQL Cheat Sheet (Unary)

- Reading
  - SELECT
  - FROM
  - GROUP BY
  - ORDER BY
  - ASC / DESC
  - DISTINCT
- Writing
  - INSERT
  - INTO
  - VALUES
  - UPDATE
  - SET
- Conditional statements
  - WHERE
  - NOT
  - LIKE
  - IS
  - AND
  - OR
  - BETWEEN
  - HAVING

StudentID	Fname	Lname	Address	City	State	Phone
211	John	Davids	11 West Main	Tucson	AZ	555-123-2345
212	Dave	Smith	234 South street	Tucson	AZ	555-464-3454
213	Jill	West	55 north front	Oro Valley	AZ	555-156-4354
214	James	Franklin	999 Fake street	Salt Lake City	UT	444-643-4364
215	Lisa	Kilp	22 Dove cove	Marana	AZ	555-545-5643
216	Eric	Northman	7 Baxter Street	Oro Valley	AZ	555-234-2344
217	Fran	Davis	22 North Oracle	Tucson	AZ	555-546-3455
218	Mike	Smith	142 West Mountain	Provo	UT	444-234-1234
219	Edna	Davis	227 Baxter Street	Marana	AZ	555-235-5355

- Wildcards
  - * (Access) - many characters
  - ? (Access) - one character
  - % (Oracle, MSSQL, MySQL) - many characters
  - _ (Oracle, MSSQL, MySQL) - one character
- Summarizing
  - COUNT()
  - SUM()
  - AVG()
  - MIN()
  - MAX()

Progress Quiz Time!

- The Progress Quiz is available in Canvas
  - You MUST complete the quiz on Canvas by 5:00 on Friday – This in-class activity does not count for points!
  - Each week we will discuss the questions, so for those of you that are in class and keeping up with things, you'll have an extra easy time with it!
- Go to http://kahoot.it and we'll get started momentarily!

BZAN 6354

Lecture 9

Go forth and do great things!

Dr. Mark Grimes, Ph.D.
gmgrimes@bauer.uh.edu

UNIVERSITY of HOUSTON
C. T. BAUER COLLEGE of BUSINESS
Department of Decision & Information Sciences