

# Other Kinds of Joins

More ways to put the pieces together

# Recall Our Old Friend the Inner Join

- We have several tables, we want to put them together in a meaningful way

bsg_cert_people	
cid	pid
3	1
1	1
3	2

bsg_cert	
Id	title
0	Raptor
1	Viper
2	Mechanic
3	Command

bsg_people				
Id	Fname	Lname	Homeworld	Age
0	William	Adama	2	61
1	Lee	Adama	2	30
2	Laura	Roslin	2	NULL
3	Kara	Thrace	2	NULL

# Inner Join Would Give Us This

- Stringing together a few Inner Joins might give us a table like this

Fname(bsg_people)	Title(bsg_cert)
Lee	Viper
Lee	Command
Laura	Command

- We put together the information in a meaningful way

# But What About Everyone Else?

- By doing inner joins we only keep matching information
- Makes sure we only have meaningful data
- But other meaningful data has been lost
- What are the names of people who have no certifications?
- What are the certifications no one holds?

# Recall How the Inner Join Worked

- We looked at the cross product of joined tables and looked for rows that shared common ids
- If common ids existed, we kept the row
- If not it was ignored
- Other kinds of joins find rows that were never included and include them

# The Left Join

- The left join keeps everything in the first table
- Will keep a null result in the right table if no matches are found

Foo	Bar_Ref
A	1
B	2
C	4

Bar	Color
1	Red
2	Blue
3	Green

- `select * from Foo left join Bar on Bar_Ref = Bar;`

FOO	BAR_REF	BAR	COLOR
A	1	1	Red
B	2	2	Blue
C	4	(null)	(null)

# Left Join Continued

- When to use the left join?
- When you want all of one table
- When you want to check if something is not in a table
- `select * from Foo left join Bar on Bar_Ref = Bar where Bar is null;`

FOO	BAR_REF	BAR	COLOR
C	4	(null)	(null)

# Right Join

- Unsurprisingly is the opposite of Left Join
- Uses are similar
- `select * from Foo right join Bar on Bar_Ref = Bar;`

Foo	Bar_Ref
A	1
B	2
C	4

Bar	Color
1	Red
2	Blue
3	Green

FOO	BAR_REF	BAR	COLOR
A	1	1	Red
B	2	2	Blue
(null)	(null)	3	Green



# Full Outer Join

- Does not exist in MySQL
- Keep all rows from left and right sides adding nulls as needed

Foo	Bar_Ref
A	1
B	2
C	4

Bar	Color
1	Red
2	Blue
3	Green

FOO	BAR_REF	BAR	COLOR
A	1	1	Red
B	2	2	Blue
(null)	(null)	3	Green
C	4	(null)	(null)

# Review

- Left, Right and Outer Joins let us keep rows that don't match any rows from the joining table
- Used when you want to display all the data from one side
- Also can be used to search for things which do not exist in a table