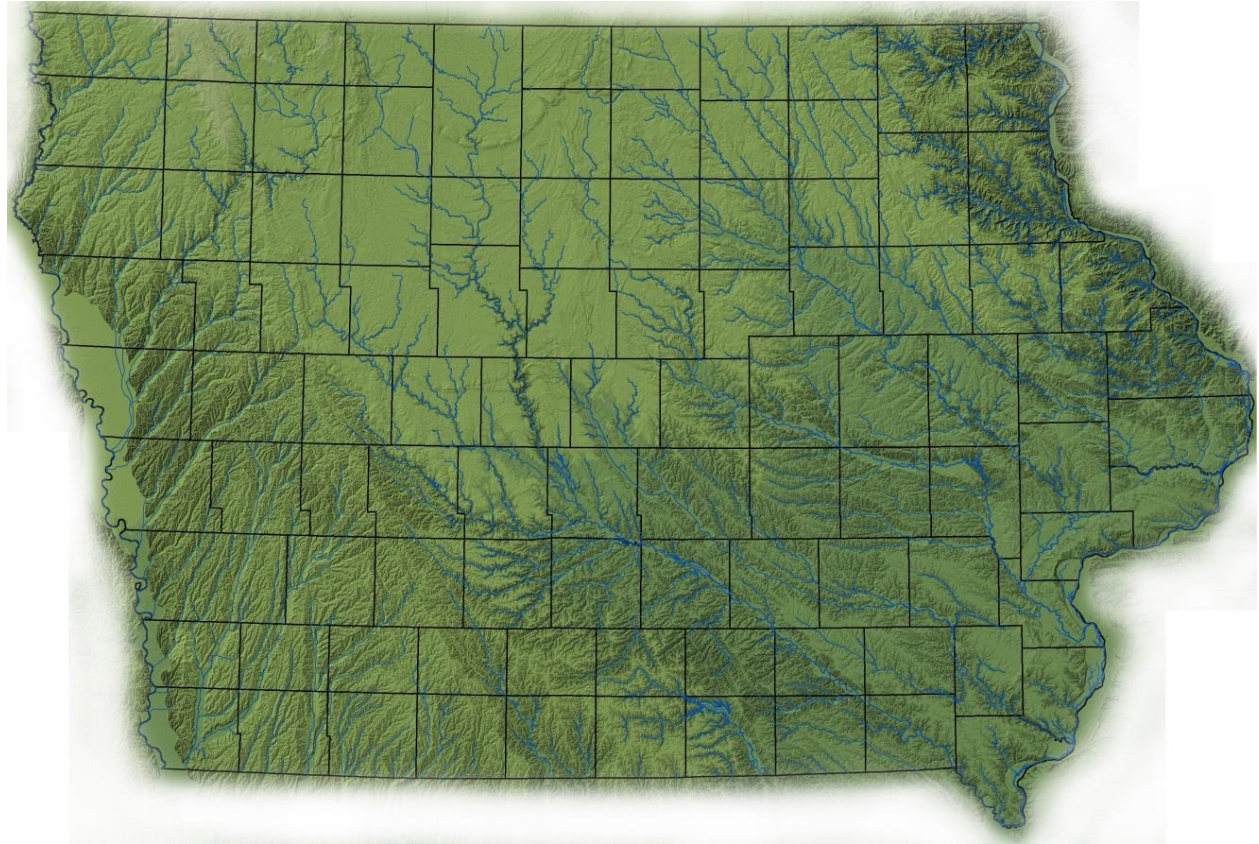




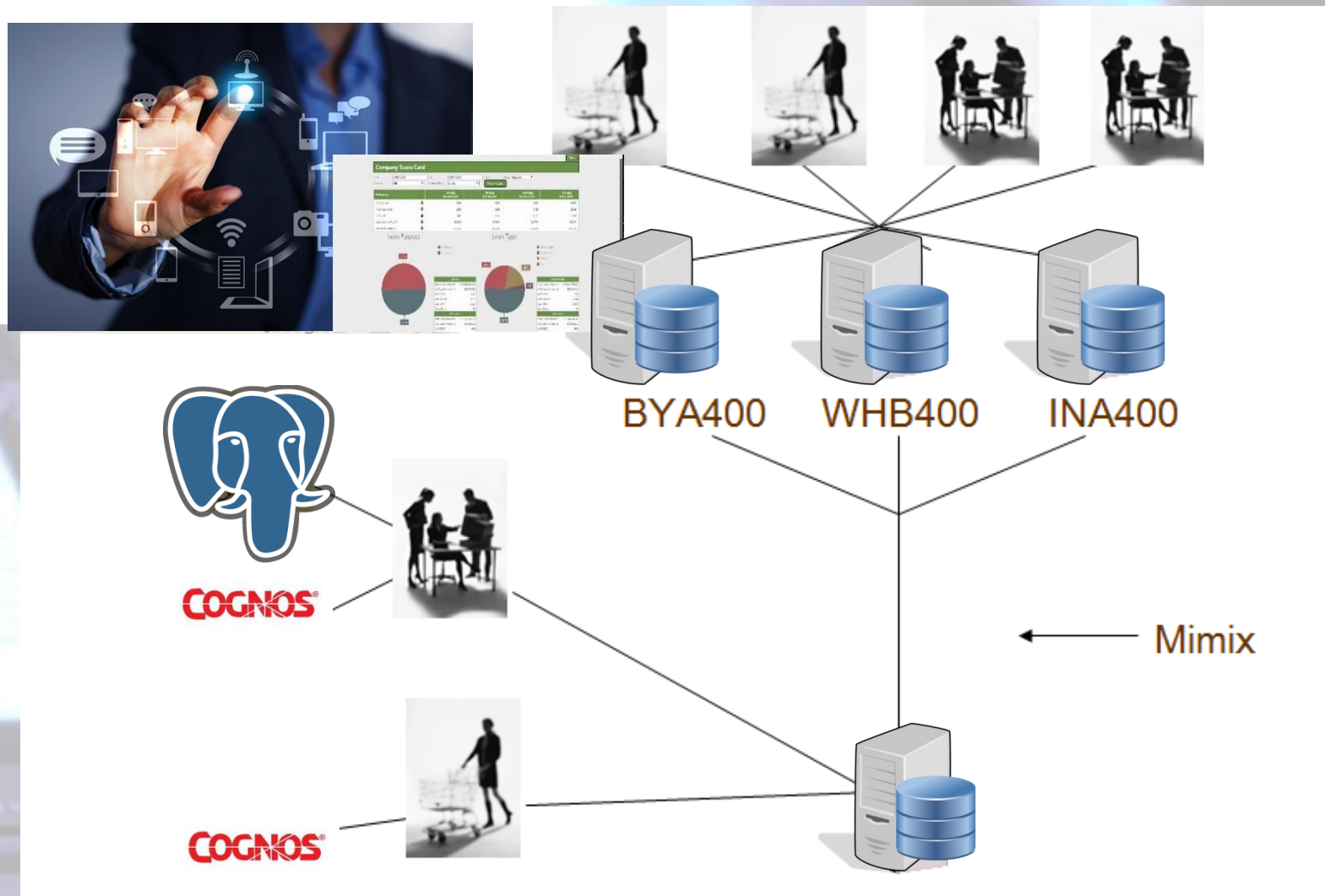
OBTAIN THE DATA

- ☐ Identify the “right” data set(s)
- ☐ Import data and set up local or remote data structure
- ☐ Determine most appropriate tools to work with data

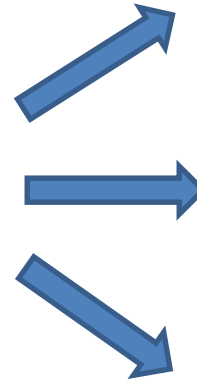
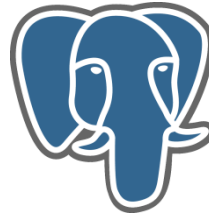
Value of Data



Value of Data



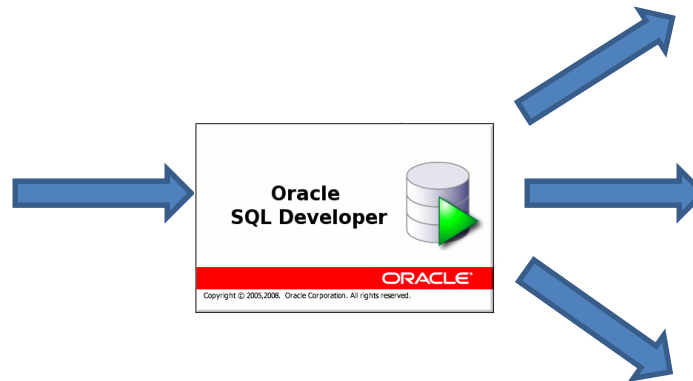
Value of Data



COGNOS



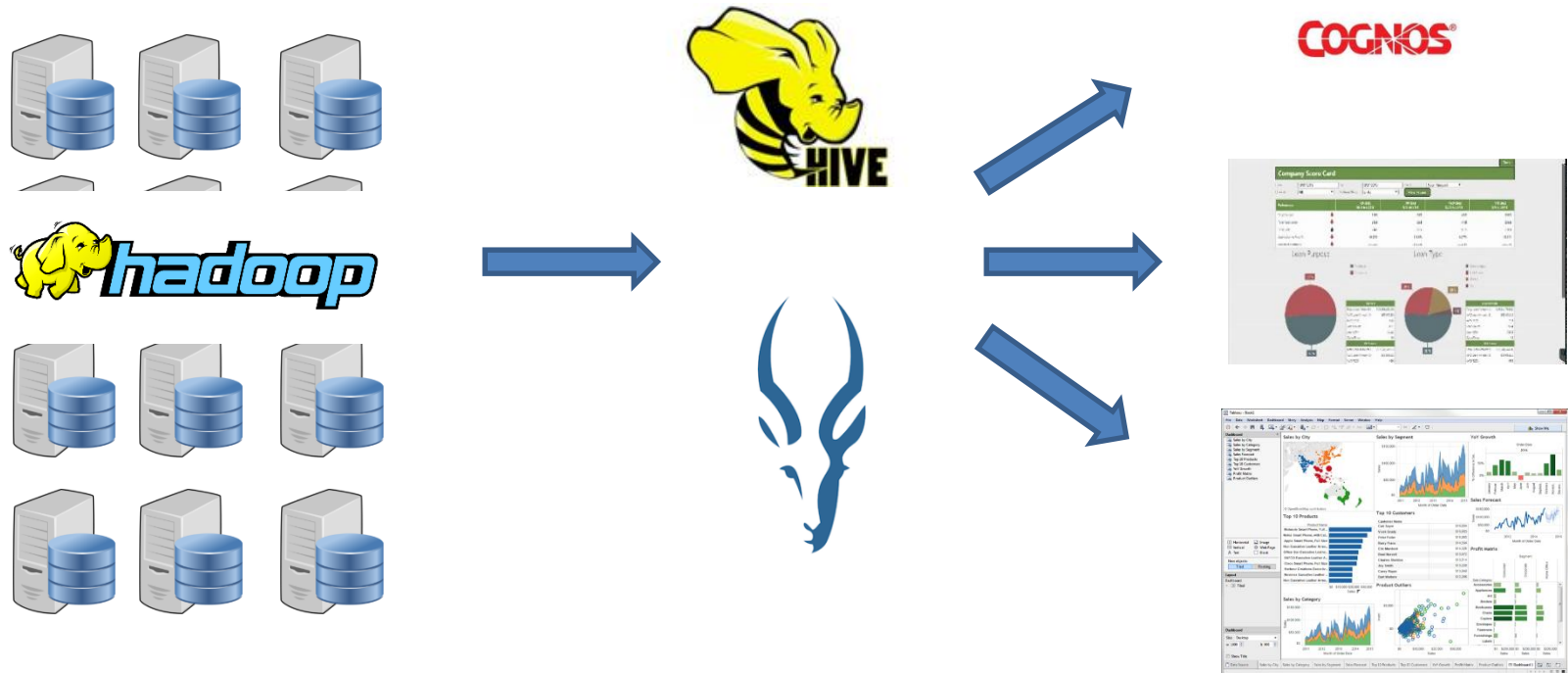
Value of Data



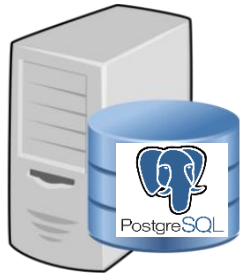
COGNOS



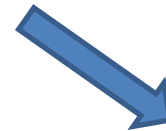
Value of Data



Value of Data



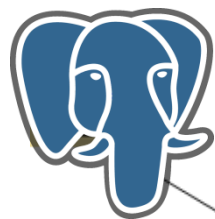
alteryx



COGNOS



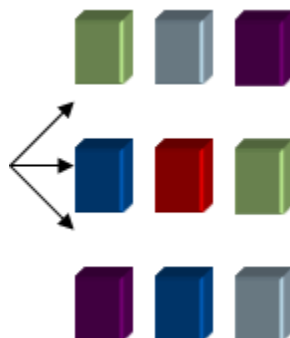
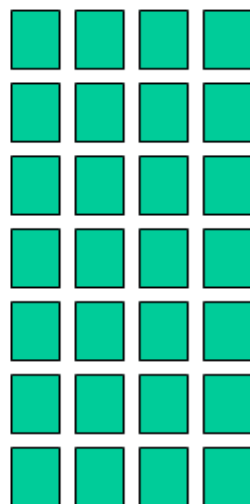
Value of Data

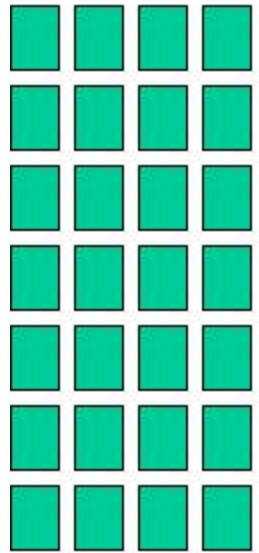


MIMIX



COGNOS®





Libraries/Collections

Schema

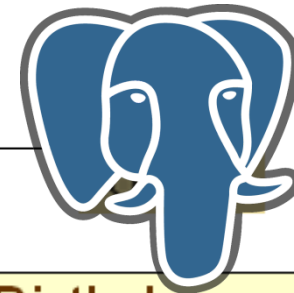
Tables/Files/Objects

Members/Partitions in files



NETWORK

<u>Contacts</u>	<u>Gift Ideas</u>	<u>Party</u>
Name	Name	Name
Friends	Gift ideas	Holidays
Family	Gifts received	Birthday
Business Name	Gifts given	Anniversaries
Address		
Phone number		
Occupation		



<u>Birthdays</u>
Name
Birthday
Gift Ideas
Gifts received
Gifts given
Family



SELECT
FROM
WHERE
ORDER BY
LIMIT

SELECT *

SELECT FIELD1, FIELD2 ...

SELECT (FIELD1+FIELD2), FIELD 3...

SELECT SUM(FIELD1), FIELD2

```
SELECT DISTINCT Location, NumberOfSales
```

Location	NumberOfSales
Seattle	101
Seattle	40
Tacoma	72

```
SELECT DISTINCT Location, NumberOfSales, Date
```

Location	NumberOfSales	Date
Seattle	101	10/28/17
Seattle	101	10/27/17
Seattle	40	10/26/17
Tacoma	72	10/28/17
Tacoma	72	10/27/17

WHERE COUNTRY = US

WHERE COUNTRY = US
AND STATE = WA

WHERE COUNTRY = US
AND STATE = WA
AND SALES > 100

ORDER BY 1

ORDER BY 1,2 DESC

LIMIT 1000

ROWNUM \leq 1000

1. Select various fields from the SALES table that interest you.
*BE sure to use LIMIT 1000
2. Practice using filters.
3. Use AND to apply multiple filters
Change the sort.
4. Save your Query



SELECT
FROM
WHERE
GROUP BY
HAVING
ORDER BY
LIMIT

GROUP BY store, item

GROUP BY 1,2 DESC

HAVING AVG(sales)>100
AND COUNT(customers)>20

SELECT

- Fields you want to see in your results

FROM

- Table where fields come from

WHERE

- Filters for your results

GROUP BY

- Groups dimensions when using an aggregate

HAVING

- Filters aggregations

ORDER BY

- How you can sort your results

LIMIT

- Limits number of records returned

Aggregate functions

- MIN
- MAX
- SUM
- COUNT



String Math Date functions

- COUNT: Returns the number of rows that matches some specified criteria.

- SYNTAX

- Count(field1)

- EXAMPLE

- SELECT category_name, count(item_no)

- FROM products

- GROUP BY category_name

- LIMIT 100

- MIN: The MIN() function returns the smallest value of the selected column

- SYNTAX

- MIN(field1)

- EXAMPLE

- SELECT store, MIN(total)

- FROM sales

- GROUP BY store

- LIMIT 100

- **MAX:** The MAX() function returns the largest value of the selected column.

- SYNTAX

- MAX(field1)

- EXAMPLE

- SELECT store, MAX(total)

- FROM sales

- GROUP BY county

- LIMIT 100

- SUM this function is used to find the sum of a field in various records.

- SYNTAX

- SUM(field1)

- EXAMPLE

- SELECT store, SUM(total)

- FROM sales

- GROUP BY store

- LIMIT 100

String Math Date functions

- **AVERAGE:** The AVG() function returns the average value of a numeric column.

- **SYNTAX**

- AVG(field)

- **EXAMPLE**

- SELECT store, AVG(total)
 - FROM sales
 - GROUP BY store
 - LIMIT 100

GROUP BY

Dimensions

Measures

Having



```
SELECT Store, (cost –sell price), SUM(sales),  
FROM sales  
WHERE Category = 'Tequila'  
AND units purchased >2  
GROUP BY Store  
HAVING SUM(sales) > 30.00  
ORDER BY 3
```

How many unique products have less than 12 in a pack?

What is the total of number of bottles sold?

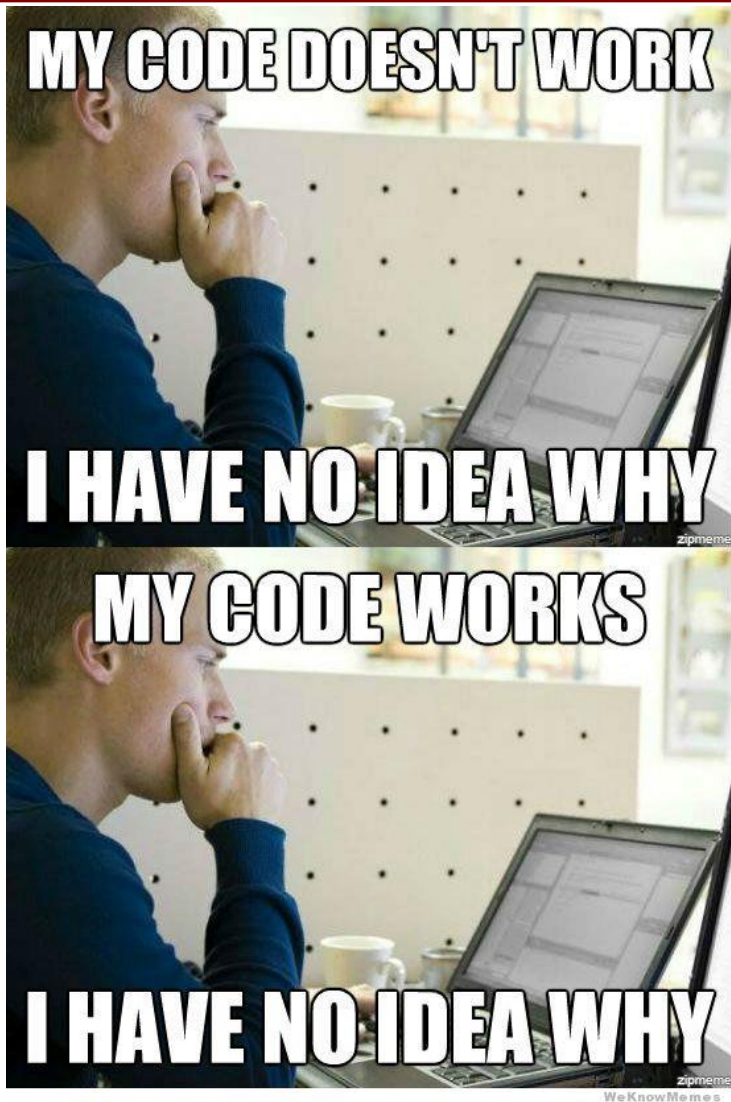
How many stores are active (use store_status)?

How many are inactive?

What is the sum of case_cost, *per item_description* for all scotch whiskies?

Bonus (try exporting to Excel and creating a pivot table to answer
What is the average bottle price per vendor of Canadian whiskies?)

Commenting



-- Basic commenting

/* Multiple line
comment

inside a line comment

End of Multiple line
comment*/

Filters and Aggregations

Which products are not from vendor 'Jim Beam Brands'?

Which products are over 90 proof?

Which products have a case cost of less than \$60?

Which products are either Single Malt Scotches or Canadian Whiskies (based on category name)?

Which products have 'Whiskies' in the category name?

Which products don't have 'Whiskies' in the category name?

Which products are have a shelf_price of between \$4 and \$10?

WHERE

- =, !=, >, <
- IS NULL, IS NOT NULL
- IN, NOT IN
- BETWEEN
- LIKE
- OR



Which products have a case cost of more than \$100?

Which tequilas have a case cost of more than \$100?

Which tequilas or scotch whiskies have a case cost of more than \$100?

Which tequilas or scotch whiskies have a case cost between \$100 and \$120?

Which whiskies of any kind cost more than \$100?

Which whiskies of any kind cost between \$100 and \$150?

Which products except tequilas cost between \$100 and \$120?

Filter and Aggregations

*Sometimes questions are more
important than answers.*

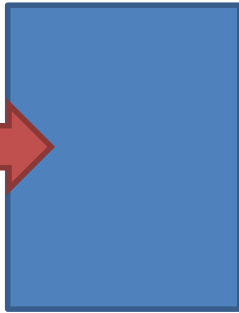
– Nancy Willard





SELECT
FROM
JOIN
ON
WHERE
GROUP BY
HAVING
UNION
ORDER BY
LIMIT

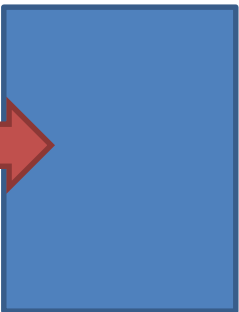
FY17



```
SELECT fy, pd, store_name, week1,  
week2, week3 week4  
FROM FY17
```

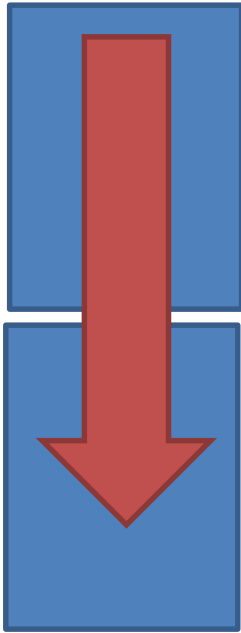
UNION

FY18



```
SELECT fy, pd, store_name, week1,  
week2, week3 week4  
FROM FY18
```

FY17



FY18

```
SELECT fy, pd, store_name, week1,  
week2, week3 week4  
FROM FY17  
UNION  
SELECT fy, pd, store_name, week1,  
week2, week3 week4  
FROM FY18
```

COLUMNS
CONDITIONS
UNION and UNION ALL
ORDER BY

Joins



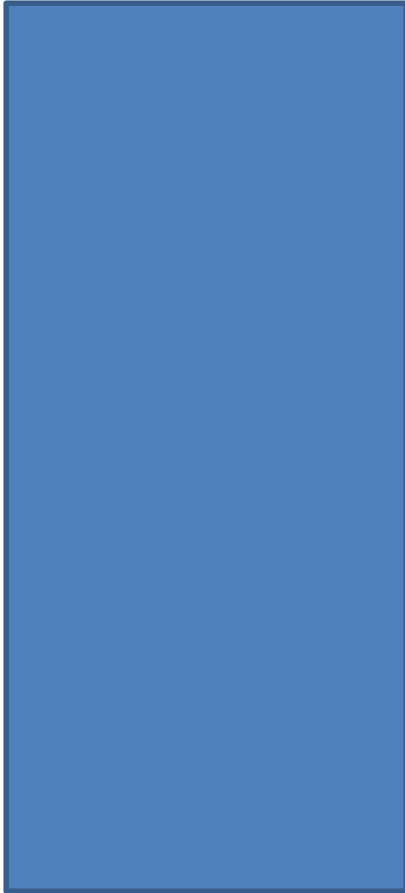
Dynamic Referencing SQL

Joins



Dynamic Referencing SQL

LEFT/PRIMARY



RIGHT/SECONDARY



What table is the transaction table?

If you wanted to link on the lowest level of detail to the other tables what fields would you use?

```
SELECT a.item, b.description, a.sales
```

```
FROM sales a
```

```
JOIN products b
```

```
ON a.item = b.item
```

1. Create separate queries to join each table to Sales
2. Products to Sales
3. County to Sales
4. Stores to Sales
5. Use this as an opportunity to bring fields in from both tables.
6. Try out some aggregations or Wild card searches.

```
SELECT b.field1, a.field2, a.field3, c.field4  
FROM table1 a  
JOIN table2 b  
ON a.field1 = b.field1  
JOIN table3 c  
ON a.field1 = c.field1  
LIMIT 100
```

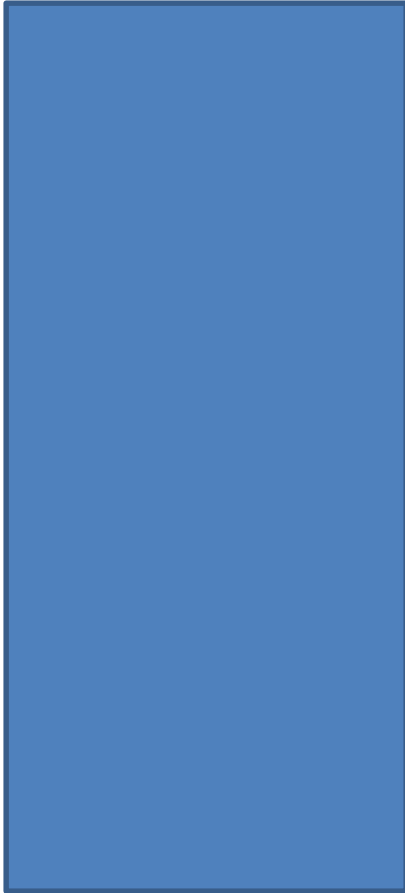
Joins

Using Sales as the primary table, create links to the all of the other tables in the Iowa liquor database. Result should be 1 query with several JOINS.

Bring back county from the county table, store from the stores table, name from the stores table, case_cost from the products table and total from the sales table. Limit to 1000.

```
SELECT b.field1, a.field2, a.field3, c.field4  
FROM table1 a  
INNER JOIN table2 b  
ON a.field1 = b.field1  
INNER JOIN table3 c  
ON a.field1 = c.field1  
LIMIT 100
```


LEFT/PRIMARY



RIGHT/SECONDARY



Types of Joins

Inner Join	Match in both tables
Left-Outer Join	Includes data from the primary table that may not have matches
Right-Outer Join	Includes data from the secondary table that may not have matches
Exception Join	Returns Primary table data that does not match with the secondary table
Right-Exception Join	Returns Secondary table data that does not match with the Primary table
Cross Join	Returns all data whether a match exists or not

Lesson Material

Employees

id	first_name	last_name
2	Gabe	Moore
3	Doreen	Mandeville
5	Simone	MacDonald
7	Madisen	Flateman
11	Ian	Paasche
13	Mimi	St. Felix

Salaries

id	current_salary
2	50000
3	60000
7	55000
11	75000
13	120000
17	70000

Inner Join

Employees

id	first_name	last_name
2	Gabe	Moore
3	Doreen	Mandeville
5	Simone	MacDonald
7	Madisen	Flateman
11	Ian	Paasche
13	Mimi	St. Felix

Salaries

id	current_salary
2	50000
3	60000
7	55000
11	75000
13	120000
17	70000

id	first_name	last_name	id	current_salary
2	Gabe	Moore	2	50000
3	Doreen	Mandeville	3	60000
7	Madisen	Flateman	7	55000
11	Ian	Paasche	11	75000
13	Mimi	St. Felix	13	7000

Left Outer Join

Employees

id	first_name	last_name
2	Gabe	Moore
3	Doreen	Mandeville
5	Simone	MacDonald
7	Madisen	Flateman
11	Ian	Paasche
13	Mimi	St. Felix

Salaries

id	current_salary
2	50000
3	60000
7	55000
11	75000
13	120000
17	70000

id	first_name	last_name	id	current_salary
2	Gabe	Moore	2	50000
3	Doreen	Mandeville	3	60000
5	Simone	MacDonald	NULL	NULL
7	Madisen	Flateman	7	55000
11	Ian	Paasche	11	75000
13	Mimi	St. Felix	13	120000

Right-Outer Join

Employees

id	first_name	last_name
2	Gabe	Moore
3	Doreen	Mandeville
5	Simone	MacDonald
7	Madisen	Flateman
11	Ian	Paasche
13	Mimi	St. Felix

Salaries

id	current_salary
2	50000
3	60000
7	55000
11	75000
13	120000
17	70000

id	first_name	last_name	id	current_salary
2	Gabe	Moore	2	50000
3	Doreen	Mandeville	3	60000
7	Madisen	Flateman	7	55000
11	Ian	Paasche	11	75000
13	Mimi	St. Felix	13	120000
NULL	NULL	NULL	17	70000

Exception Joins

Employees

id	first_name	last_name
2	Gabe	Moore
3	Doreen	Mandeville
5	Simone	MacDonald
7	Madisen	Flateman
11	Ian	Paasche
13	Mimi	St. Felix

Salaries

id	current_salary
2	50000
3	60000
7	55000
11	75000
13	120000
17	70000

id	first_name	last_name	id	current_salary
5	Simone	MacDonald	NULL	NULL

id	first_name	last_name	id	current_salary
5	Simone	MacDonald	NULL	NULL

Lesson Material

Employees

id	first_name	last_name
2	Gabe	Moore
3	Doreen	Mandeville
5	Simone	MacDonald
7	Madisen	Flateman
11	Ian	Paasche
13	Mimi	St. Felix

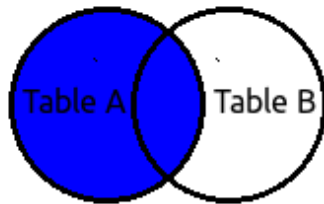
Salaries

id	current_salary
2	50000
3	60000
7	55000
11	75000
13	120000
17	70000

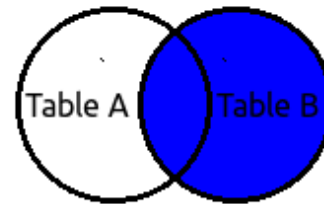
id	first_name	last_name	id	current_salary
2	Gabe	Moore	2	50000
3	Doreen	Mandeville	3	60000
5	Simone	MacDonald	NULL	NULL
7	Madisen	Flateman	7	55000
11	Ian	Paasche	11	75000
13	Mimi	St. Felix	13	120000
NULL	NULL	NULL	17	70000

Projects

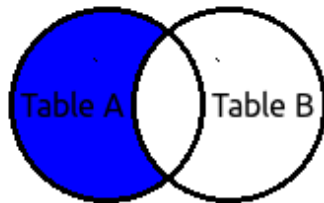
Joins



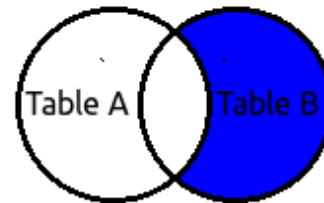
```
SELECT [list] FROM  
[Table A] A  
LEFT JOIN  
[Table B] B  
ON A.Value = B.Value
```



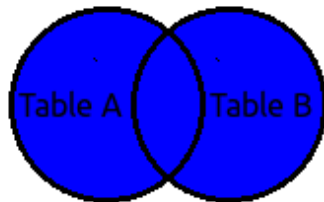
```
SELECT [list] FROM  
[Table A] A  
RIGHT JOIN  
[Table B] B  
ON A.Value = B.Value
```



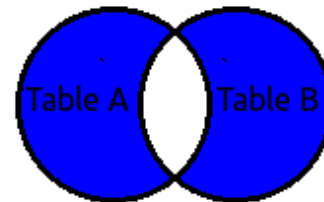
```
SELECT [list] FROM  
[Table A] A  
LEFT JOIN  
[Table B] B  
ON A.Value = B.Value  
WHERE B.Value IS NULL
```



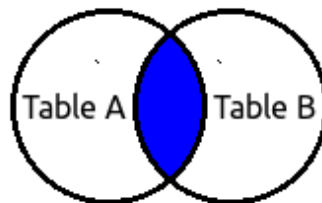
```
SELECT [list] FROM  
[Table A] A  
RIGHT JOIN  
[Table B] B  
ON A.Value = B.Value  
WHERE A.Value IS NULL
```



```
SELECT [list] FROM  
[Table A] A  
FULL OUTER JOIN  
[Table B] B  
ON A.Value = B.Value
```



```
SELECT [list] FROM  
[Table A] A  
FULL OUTER JOIN  
[Table B] B  
ON A.Value = B.Value  
WHERE A.Value IS NULL  
OR B.Value IS NULL
```



```
SELECT [list] FROM  
[Table A] A  
INNER JOIN  
[Table B] B  
ON A.Value = B.Value
```

Dynamic Referencing SQL

RIGHT JOIN

```
SELECT b.location, b.address, b.status,  
a.location, a.sales  
FROM sales a  
RIGHT JOIN stores b  
ON a.location = b.location
```

LEFT JOIN

```
SELECT a.location, a.sales, b.location,  
b.address, b.status  
FROM sales a  
LEFT JOIN stores b  
ON a.location = b.location
```

Request:
What Tequila products are not selling?