SQL DATABASE FUNDAMENTALS

LEARNING OBJECTIVES

In today's lesson, we will:

1. Review the market trend toward self-serve data access (using SQL).
2. Learn about database structures and the role of structured query language (SQL).
3. Introduce SQL's `SELECT` statement with `WHERE` clauses.
4. Explore practice data, using the Iowa Liquor Sales Database.
5. Practice a selection of query command tools, including: `DISTINCT`, `COUNT`, `AND`, `OR`, and `CAST`.

INTRODUCTION

DATA ANALYTICS: AGILE WRANGLING

Faster Data Preparation Unlocks Agility And Insights

By gaining access to more data and reducing the amount of time it takes to prepare data, data professionals can increase their productivity and value to the organization.

Source: A commissioned study conducted by Forrester Consulting on behalf of Datawatch, February 2016
*Source: Forrester's Global Business Technographics® Data And Analytics Survey, 2015

BASIC SQL

SQL (pronounced: “si-kwel”) is short for structured query language. It was developed at IBM by Donald D. Chamberlin and Raymond Boyce in the early 1970s.

There are a number of different types of SQL database implementations and platforms, all of which support standard (ANSI) SQL, including:

- PostgreSQL
- MySQL
- MS SQL
- Oracle

BASIC SQL

Sources: https://goo.gl/ANp296

EXCEL VS. SQL

Why do we need SQL when we have Excel?

- Excel is limited by your computer's available memory and system resources.
- Excel has a fixed upper **limit** of **1,048,576 rows** and **16,384 columns**.

In other words, Excel is a local tool that is not able to capably manage or interact with very large datasets. *This is when SQL steps in!*

EXCEL VS. SQL

What can SQL do that Excel can’t?

- SQL can rapidly navigate databases and query, retrieve, and aggregate **millions** of records.
- SQL is also more adept than Excel at creating data flows for cleaning and preparing data at high volumes.
- SQL is the industry standard for data query and retrieval.

EXCEL VS. SQL

However…

- SQL is **not** a data visualization tool.
- SQL can query and organize data, but is not typically used to analyze it.

In other words, SQL is **not** a substitute for Excel. Instead, SQL is normally used in conjunction **with** Excel and other data visualization tools when working with large data repositories.

EXCEL VS. SQL

Ok, so let's summarize:

- Excel and SQL use similar functions. For example, both tools offer ways for you to apply methods like: `COUNT`, `SUM`, `AVG`, `IF THEN ELSE`, etc.
- But compared to spreadsheets with tons of linked tabs, SQL databases provide users with a much more efficient way to connect related data.
- SQL is also much more scalable than Excel, allowing users to remotely interact with large datasets in production environments.

BASIC SQL: POSTGRES

The SQL database tool we'll use in this class is PostgreSQL. We'll use this because:

- It's an object-relational database management system (DBMS).
- It's powerful and standardized.
- It's free and open source!

Note:

Many of the SQL skills we'll learn are applicable to other SQL database implementations. Many of the same principles and syntax rules apply to other versions, like MS SQL, etc.

PRACTICE DATA SET: IOWA LIQUOR SALES DATABASE

ACTIVITY: NAVIGATING THE IOWA LIQUOR SALES DATABASE

DIRECTIONS

For this unit, we will be working as new analysts at the consulting firm Deloitte. We have been added to an account for the State of Iowa.

Iowa is looking to understand statewide alcohol sales and consumption. We will have access to a database of information provided by the state.

DELIVERABLES

- Connect to the SQL database that we'll be using for this unit.
- Explore the functions of the client software (execute, stop, save, new query).
- Look at the first and last 100 rows of the data from the tables using the menus.
- Review how the column properties are defined using the menus.

Iowa Liquor Sales

Economy

This dataset contains the spirits purchase information of Iowa Class "E" liquor licensees by product and date of purchase from January 1, 2012 to current. The dataset can be used to analyze total spirits sales in Iowa of individual products at the store level.

Updated
April 1, 2017

data.iowa.gov
EXPLORE • DISCOVER • SHARE

Unsaved View Save As... Revert

Based on Iowa Liquor Sales

This dataset contains the spirits purchase information of Iowa Class "E" liquor licensees by product and date of purchase from January 1, 2012 to current. The dataset can be used to analyze

Find in this Dataset

Manage More Views Filter Visualize Export Discuss Embed About

	Invoice/Item Number	Date	Store Number	Store Name	Address	City	Zip Code	Store Location	County Number	County
1	INV-04039700001	03/31/2017	5378	Casey's General Store # 2543	201 S Main St.	Albion	50005	(42.109312°, -92.9	64	MARSH
2	INV-04111400036	03/31/2017	4073	Uptown Liquor, Llc	306 Hwy 69 South	Forest City	50436	(43.261538°, -93.6	95	WINNE
3	INV-04021400001	03/31/2017	3728	Brewski's Beverage	726 Creek Top	Council Bluffs	51503	(41.262316°, -95.8	78	POTTA
4	INV-04100700038	03/31/2017	4823	Casey's General Store #3223	903 N Sumner Ave	Creston	50801	(41.066824°, -94.3	88	UNION
5	INV-04100600043	03/31/2017	2553	Hy-Vee Food Store / Creston	600 Sheldon	Creston	50801	(41.051313°, -94.3	88	UNION
6	INV-04100600011	03/31/2017	2553	Hy-Vee Food Store / Creston	600 Sheldon	Creston	50801	(41.051313°, -94.3	88	UNION
7	INV-04100800026	03/31/2017	4291	Fareway Stores #597 / Cresto	105 E Adams St	Creston	50801	(40.71572°, -94.23	88	UNION
8	INV-04100800017	03/31/2017	4291	Fareway Stores #597 / Cresto	105 E Adams St	Creston	50801	(40.71572°, -94.23	88	UNION
9	INV-04111000040	03/31/2017	2585	Hy-Vee Wine and Spirits / Alg	1516 Highway 169 North	Algona	50511	(43.081288°, -94.2	55	KOSSU
10	INV-04111000056	03/31/2017	2585	Hy-Vee Wine and Spirits / Alg	1516 Highway 169 North	Algona	50511	(43.081288°, -94.2	55	KOSSU
11	INV-04110700007	03/31/2017	3682	Jumbo's	110 E Main St	Wesley	50483	(43.088185°, -93.9	55	KOSSU
12	INV-04111000009	03/31/2017	2585	Hy-Vee Wine and Spirits / Alg	1516 Highway 169 North	Algona	50511	(43.081288°, -94.2	55	KOSSU
13	INV-04110300013	03/31/2017	5145	South Side Food Mart	1101 Army Post Rd	Des Moines	50315	(41.526511°, -93.6	77	POLK
14	INV-04110100012	03/31/2017	5131	Oasis	1401, Buchanan St	Des Moines	50316	(41.598791°, -93.5	77	POLK
15	INV-04110000017	03/31/2017	3698	Ingersoll Wine Merchants	1300 50th St	West Des Moines	50266	(41.590889°, -93.7	77	POLK
16	INV-04110000009	03/31/2017	3698	Ingersoll Wine Merchants	1300 50th St	West Des Moines	50266	(41.590889°, -93.7	77	POLK
17	INV-04109900001	03/31/2017	4669	Vom Fass / Des Moines	833, 42nd St	Des Moines	50312	(41.593431°, -93.6	77	POLK
18	INV-04098800023	03/31/2017	2607	Hy-Vee Wine and Spirits / She	520 So Fremont	Shenandoah	51601	(40.761736°, -95.3	73	PAGE
19	INV-04109600037	03/31/2017	4597	Quik Trip #523 / Army Post Dr	850 Army Post Rd	Des Moines	50315	(41.526428°, -93.6	77	POLK
20	INV-04109600029	03/31/2017	4597	Quik Trip #523 / Army Post Dr	850 Army Post Rd	Des Moines	50315	(41.526428°, -93.6	77	POLK
21	INV-04109500020	03/31/2017	4478	U S Gas	3000, SW 9th St	Des Moines	50315	(41.557862°, -93.6	77	POLK
22	INV-04109500005	03/31/2017	4478	U S Gas	3000, SW 9th St	Des Moines	50315	(41.557862°, -93.6	77	POLK
23	INV-04109300037	03/31/2017	5245	Phillips 66	1516, SE 1st St	Des Moines	50315	(41.575862°, -93.6	77	POLK
24	INV-04109300029	03/31/2017	5245	Phillips 66	1516, SE 1st St	Des Moines	50315	(41.575862°, -93.6	77	POLK
25	INV-04109100066	03/31/2017	3443	Super Saver Iv	1141 N Broadway	Council Bluffs	51503	(41.270881°, -95.8	78	POTTA
26	INV-04109100059	03/31/2017	3443	Super Saver Iv	1141 N Broadway	Council Bluffs	51503	(41.270881°, -95.8	78	POTTA

Visualize

Calendar

Map

Chart

Visualization Type: Tree Map

Column, Stacked Column, Bar, Stacked Bar, Pie, Donut, Line, Area, Timeline, Bubble, Tree Map

Data Selection

Configuration

* Choose label data: No column selected

* Choose value data: No column selected

Data Presentation

Colors

Chart Details

GUIDED PRACTICE: NAVIGATION

Connect to the server by navigating to it in the object browser. We'll explore the object browser to see how the database is organized.

Here is the directory tree we'll need to access the data tables:

- Servers (*pgAdmin can be configured for more than one server configuration.*)
  - Databases
    - Iowa_Liquor_Database (*our database for today*)
      - Schemas
        - Public
          - Tables
            - Columns

GUIDED PRACTICE: NAVIGATION

If we want to view the contents of a table, we should:

1. Navigate to the tables of the Iowa database's public schema.
2. There you should see four tables: Products, Sales, Stores, and Counties.
3. To view the data, right click on the table, go to View Data, and select the top 100 rows.

GUIDED PRACTICE: NAVIGATION

To understand the data, we need know what the **stored data** means. We will refer to this as telling “the story of one row.”

Examining 100 rows of data may help us better understand if this is the data set we need to answer our research questions.

This subset allows us to preview the data.

INDEPENDENT PRACTICE: IOWA LIQUOR SALES DATABASE

UNDERSTANDING THE IOWA LIQUOR SALES DATABASE

DIRECTIONS

1. Go through the tables in the database and explore the data they contain.
2. Characterize each table as either transactional or reference.
3. Take five minutes and make notes about the database.

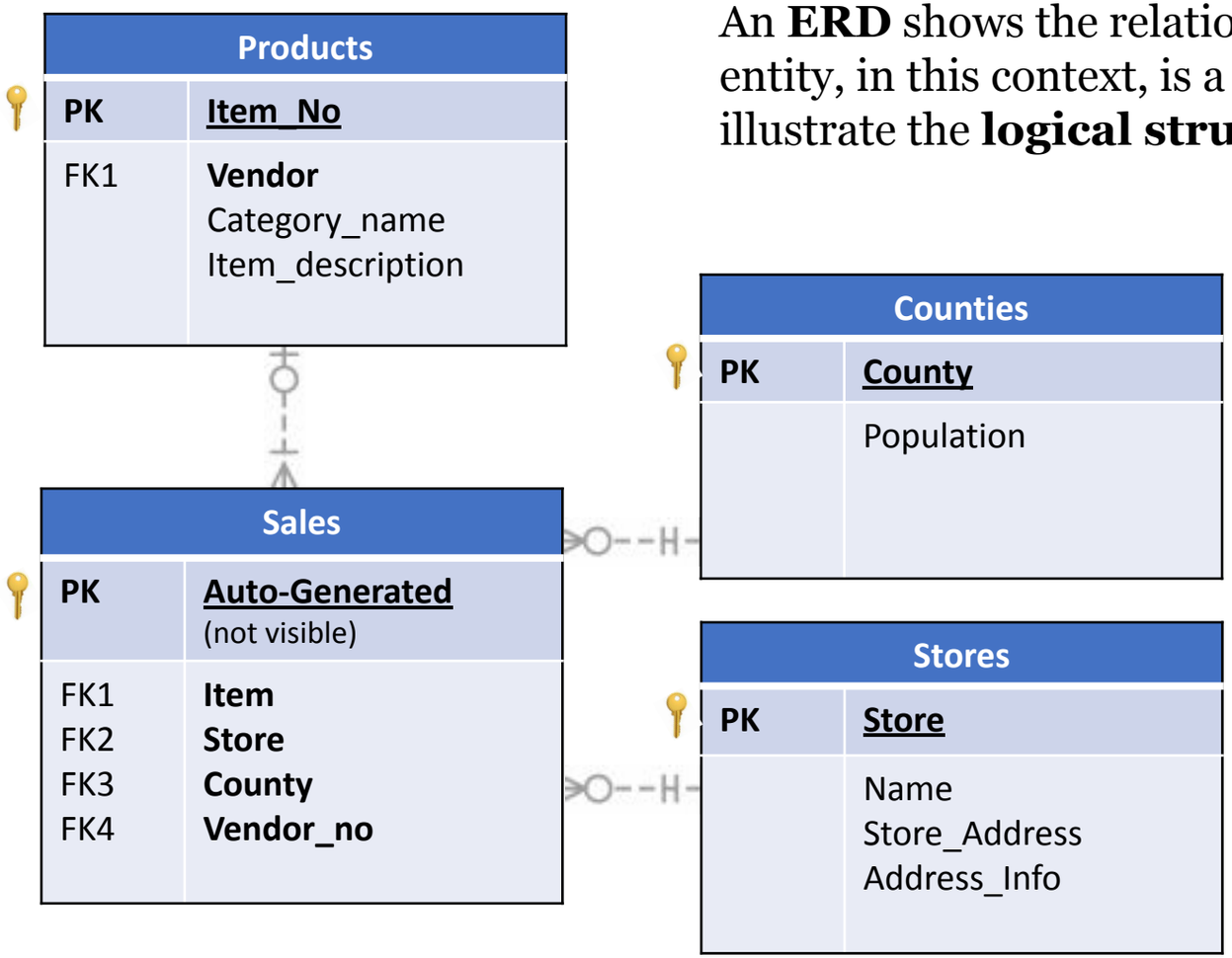
DELIVERABLES

1. Write a few sentences describing the data stored in each table.
   Note the data types assigned to each column.
2. What columns could serve as links between tables later in our data exploration?

DATA DESCRIPTION

Download and Open: Data Description - Iowa Liquor Sales DB.pdf

DATA DESCRIPTION: ENTITY RELATIONSHIP DIAGRAM (ERD)



An **ERD** shows the relationships of **entity sets** stored in a database. An entity, in this context, is a component of data. In other words, ERDs illustrate the **logical structure** of databases.

PK: Primary key
FK: Foreign key

data.iowa.gov
EXPLORE • DISCOVER • SHARE

GUIDED PRACTICE: BASIC SQL

BASIC SQL: SELECT STATEMENT PREVIEW

- ✓ **`SELECT`**: *Selects* the columns.
- ✓ **`FROM`**: *Points* to the table.
- ✓ **`WHERE`**: *Filters* on rows.
- ✓ **`GROUP BY`**: *Aggregates* across values of a variable.
- ✓ **`HAVING`**: *Filters* groups.
- ✓ **`ORDER BY`**: *Sorts or arranges* the results.
- ✓ **`LIMIT`**: *Limits* result to the first `n` rows.

BASIC SQL

SQL is used to communicate questions to the database. The three main clauses are `SELECT`, `FROM`, and `WHERE`.

1. `SELECT`
    - Allows you to select certain columns from a table.
    - Determines which columns of information are downloaded.
2. `FROM`
    - Specifies the tables from which the query extracts data.
    - Defines the relationships between the tables (JOIN conditions).
3. `WHERE`
    - Filters which rows are selected from the tables.

BASIC SQL

When we selected the top 100 rows of a table using the View Data menu or Query Tool menu selections, we actually ran a SQL statement in the background:

```
SELECT * FROM products LIMIT 100;
```

Let's think through these questions:

- What does `*` mean?
- What does "`FROM products`" mean?
- What does the `LIMIT` do?

BASIC SQL

All queries can be run in the pgAdmin SQL window. For the remainder of the lesson, we'll be modifying queries.

We can tell **SELECT** which columns or variables we want:

```
SELECT item_no, item_description FROM products LIMIT 100;

SELECT store, store_address FROM stores LIMIT 100;
```

We can add **DISTINCT** to the query statement to eliminate duplicates:

```
SELECT DISTINCT category_name, vendor_name FROM products;
```

BASIC SQL: FILTER WITH WHERE

We may query all columns with **SELECT** * or query for specific ones. Columns are presented in the order of the **SELECT** query line.

DISTINCT further defines the results by eliminating exact duplicates. The **WHERE** clause filters rows by setting a criteria:

```
SELECT * FROM products WHERE category_name = 'SCOTCH WHISKIES';
SELECT DISTINCT vendor_name FROM products WHERE category_name =
'SCOTCH WHISKIES';
SELECT * FROM products WHERE case_cost >= 100;
```

BASIC SQL: ORDER BY

`ORDER BY` sorts results in an ascending or descending order. After the **`ORDER BY`** is a number that indicates the column by which you're sorting.

The default sort order is ascending, but you can specify ascending (**`ASC`**) or descending (**`DESC`**) to determine the sort order.

```
SELECT * FROM products WHERE case_cost >= 100 ORDER BY 1;
```

BASIC SQL: DATA VALIDATION

COUNT returns the number of rows that matches some specified criteria. If the criteria includes only a column name, this returns the number of non-**NULL** values in that column.

Syntax: `COUNT(field1)`

Example:

```
SELECT category_name, COUNT(item_no)
FROM products
GROUP BY category_name
ORDER BY category_name;
```

More on this later….

BASIC SQL: AND, OR, & CAST( )

Some additional common query methods that we'll be using include:

- **AND**: Returns TRUE if both conditions are true.
- **OR**: Returns FALSE if neither condition is true (TRUE if either is true).
- **CAST**(field AS type): Converts the "field" content to the specified data type to enable comparisons.

INDEPENDENT PRACTICE: BASIC SQL

INDEPENDENT PRACTICE: BASIC SQL

DIRECTIONS

Let's answer the following questions by writing and executing SQL queries:

1. Which products come in packs larger than 12? How many unique products have less than 12 in a pack?
2. Which products have a case price of less than $70?
3. Which products come in packs larger than 12 AND have a case_cost of less than $70?
4. Which types of products have a proof of 85 or higher?
5. Which products are scotch whiskies OR are higher than 85 proof?
6. How many stores are active (use store_status)? Inactive?
7. Work in small groups and share your results with the class.

CONCLUSION

RESOURCES

REVIEW: FUNDAMENTALS OF DATABASES AND SQL

In today's lesson, we learned how to :

1. Define database structures and the role of SQL.
2. Connect to our database for this unit.
3. Define basic SQL grammar, syntax, and punctuation.
4. Explain how to use SQL's `SELECT` statement with `WHERE` clauses.
5. Apply various SQL statements to sample data.
6. Identify a range of additional SQL commands, including `DISTINCT`, `COUNT`, `AND`, `OR`, and `CAST`.

FUNDAMENTALS OF DATABASES AND SQL

Q&A

RESOURCES

- **AND**/**OR**: https://www.techonthenet.com/sql/and_or.php

- **CAST** Function: https://msdn.microsoft.com/en-us/library/ms187928.aspx