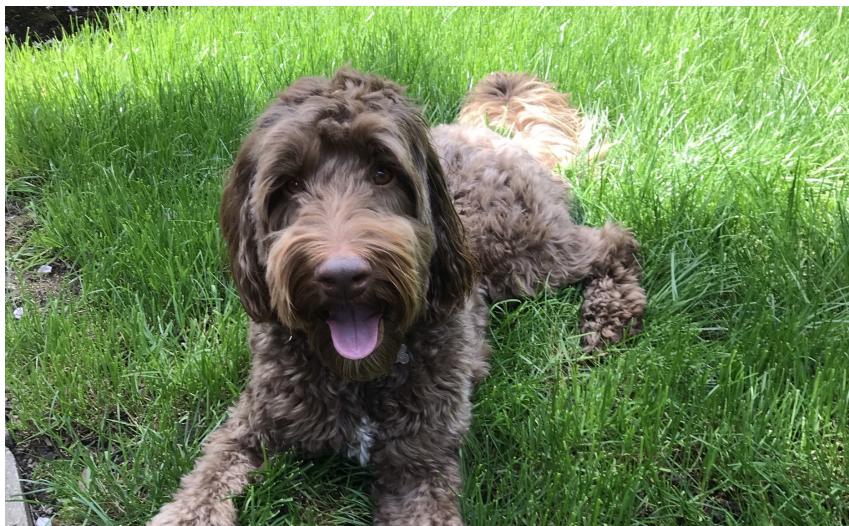


# Intro to Databases

---

JUNE 30, 2021



# About Me

---

- Name: Daman Oberoi
- BSEE 2004, MSCS 2006
- 12 years at Intel
  - SW Engineer
  - Business development
  - People manager
  - Project/Program manager
- CS Instructor, Univ of Portland
- Interests: Sports (esp basketball), video games, board games, eating, TV
- In addition to CS, I can offer guidance on:
  - Writing resumes & cover letters
  - Interview prep
  - Workplace etiquette
  - Other career tips

# What is a Database?

---

- A database (DB) is a systematic collection of data.
- There are two kinds of databases: relational and non-relational.
- You can think about a relational database as an Excel file.
  - Each tab in Excel is a **table** in the database.
  - Each row is a **record**.
  - Each column is a **field**.

CustomerID	CustomerName	ContactName
1	Alfreds Futterkiste	Maria Anders
2	Ana Trujillo Emparedados y helados	Ana Trujillo
3	Antonio Moreno Taquería	Antonio Moreno
4	Around the Horn	Thomas Hardy
5	Berglunds snabbköp	Christina Berglund
6	Blauer See Delikatessen	Hanna Moos
7	Blondel père et fils	Frédérique Citeaux
8	Bólido Comidas preparadas	Martín Sommer
9	Bon app'	Laurence Lebihans
10	Bottom-Dollar Marketse	Elizabeth Lincoln

Sources:

<https://www.guru99.com/introduction-to-database-sql.html>

[https://www.w3schools.com/sql/sql\\_intro.asp](https://www.w3schools.com/sql/sql_intro.asp)

# Non-Relational Database

---

- Non-relational databases are often called NoSQL databases.
- They store data in a non-tabular format and instead use data structures like documents.
- A document can be highly detailed while containing a range of different types of information in different formats.
- Non-relational databases may perform faster because a query doesn't have to view several tables in order to deliver an answer, as relational datasets often do.
- Non-relational can support rapidly developing applications requiring a dynamic database able to change quickly and to accommodate large amounts of complex, unstructured data.
- Non-relational databases are less common, at least in part because they don't use SQL, so we will focus on relational databases in this course.

Sources:

<https://www.mongodb.com/non-relational-database>

# What is SQL?

---

- SQL (Structured Query Language) is a standard language for accessing and manipulating databases.
- Although SQL is a standard, there are different versions of the SQL language. But all versions must support at least the major commands (such as SELECT, UPDATE, DELETE, INSERT, WHERE) in a similar manner.

Sources:

[https://www.w3schools.com/sql/sql\\_intro.asp](https://www.w3schools.com/sql/sql_intro.asp)

# Commands

---

Command	Description
<b>BASIC NAVIGATION &amp; QUERYING</b>	
mysql	Start an interactive SQL shell in Codio
SHOW DATABASES;	View list of databases
USE <i>database_name</i> ;	Select a database to use
SHOW TABLES;	View list of tables (must select a database first)
SELECT * FROM <i>table_name</i> ;	View records in table
<b>CREATING A DATABASE AND ADDING DATA</b>	
CREATE DATABASE <i>database_name</i> ;	Create a database
CREATE TABLE <i>table_name</i> ( <i>field types</i> )	Create a table in the selected database. See following slides for more info.
INSERT INTO <i>table_name</i> ( <i>fields</i> ) VALUES ( <i>values</i> );	Insert a record into the specified table. See following slides for more info.

# CREATE TABLE

```
CREATE TABLE Cars (
    id INT(8) UNSIGNED NOT NULL auto_increment,
    model VARCHAR(255) default NULL,
    manufacture_year YEAR(4) default NULL,
    PRIMARY KEY (id)
)
AUTO_INCREMENT=1;
```

- This command identifies the (1) table name, (2) the column names, (3) the type of data to be stored in each column.
- What is the table name?
  - Cars
- How many fields are specified?
  - 3

\*\* Make sure you have called USE *database\_name*; before you try to create a table \*\*

- What are the types of each field?
  - id → INT(8)
  - model → VARCHAR(255)
  - manufacture\_year → YEAR(4)
- What does UNSIGNED NOT NULL auto\_increment mean?
  - Only positive integers. NULL value is not allowed. No need to enter this data, instead it gets incremented automatically
- What does default NULL mean?
  - If no value is specified, it will be set to NULL
- What does PRIMARY KEY (id) mean?
  - The primary key value must be unique for each record in the table. This table will use the id as the unique identifier.

# Text Data Types (sample)

---

## Text types

Data type	Description
VARCHAR(size)	<p>The <b>variable character field</b> can contain letters, numbers, and special characters.</p> <p>Max. size is specified in parenthesis.</p> <p><b>Can store up to 255 characters</b>, just enough for a Twitter post.</p> <p>Note: If you put a greater value than 255 it will be converted to a TEXT type</p>
TEXT	Holds a string with a maximum length of 65,535 characters
LONGTEXT	Holds a string with a maximum length of 4,294,967,295 characters

# Numeric Data Types (sample)

---

## Number types:

Data type	Description
INT(size)	-2147483648 to 2147483647 normal. 0 to 4294967295 UNSIGNED. The <b>maximum number of digits may be specified in parenthesis</b>

The integer (INT) type has an extra option called UNSIGNED.

Normally, the integer ranges from a negative to positive value. Adding the UNSIGNED attribute will move that range up so it starts at zero instead of a negative number (more on this later).

# Date Data Types (sample)

---

## Date types:

Data type	Description
<code>DATE()</code>	A date. Format: YYYY-MM-DD <b>Note:</b> The supported range is from '1000-01-01' to '9999-12-31'
<code>YEAR()</code>	A year in two-digit or four-digit format. <b>Note:</b> Values allowed in four-digit format: 1901 to 2155. Values allowed in two-digit format: 70 to 69, representing years from 1970 to 2069

# INSERT TABLE

---

```
INSERT INTO Cars (make, manufacture_year)  
VALUES ("Ford", 1960);
```

- You can specify the fields in any order (though left to right is conventional), but the order of the values must match the order of the specified fields.
- If you wanted to use a default value, you could do this:

```
INSERT INTO Cars (make, manufacture_year)  
VALUES ("Ford", DEFAULT);
```

\*\* Make sure you have called `USE database_name;` before you try to insert \*\*

# Today's Activity

---

1. You will work with an interactive SQL shell to practice all of the commands we covered today.
  2. You will create a database and load your API data into it.
- 
- You will go into your TA's breakout room and work on the activity.
  - Feel free to work with others in your room and ask questions of the group.
  - The TA is there to assist you. If you want to ask me a question, come back to the main zoom room.

# To Convert Dictionary into DataFrame

---

- The Codio guide provides the following link. This may or may not prove useful to you, depending on how you implement your program.

[https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.from\\_dict.html?highlight=dataframe%20from\\_dict#pandas.DataFrame.from\\_dict](https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.from_dict.html?highlight=dataframe%20from_dict#pandas.DataFrame.from_dict)

- Another way to create a DataFrame from your data is as follows. Note the parts in *italics* that you must update for your program.

```
# creating data frame to add data to
col_names = ['col1', 'col2', 'col3']
df = pd.DataFrame(columns = col_names)
df.loc[len(df.index)] = [value1, value2, value3]
```

# To Load Your API into the Database

---

You will need these commands, which are given to you in the Codio guide. Note the parts in *italics* that you must update for your program.

```
engine = create_engine('mysql://root:codio@localhost/database_name')
```

```
dataframe_name.to_sql('table_name', con=engine, if_exists='replace', index=False)
```

when your lecturer asks if you have  
any questions



# Questions?

---