



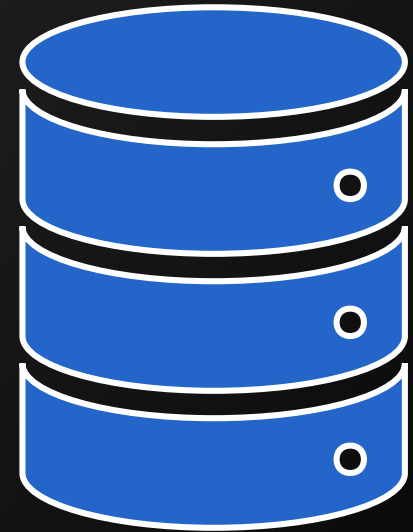
DATABASE DESIGN

Julie Allen 2023

DATABASE DESIGN

What we will cover this semester:

- What is a database
- The Different Database Models
- Relational Databases
- SQL – Structured Query Language
- Entity Relationship Diagrams
- Database Normalisation
- Data Integrity
- Database Transactions

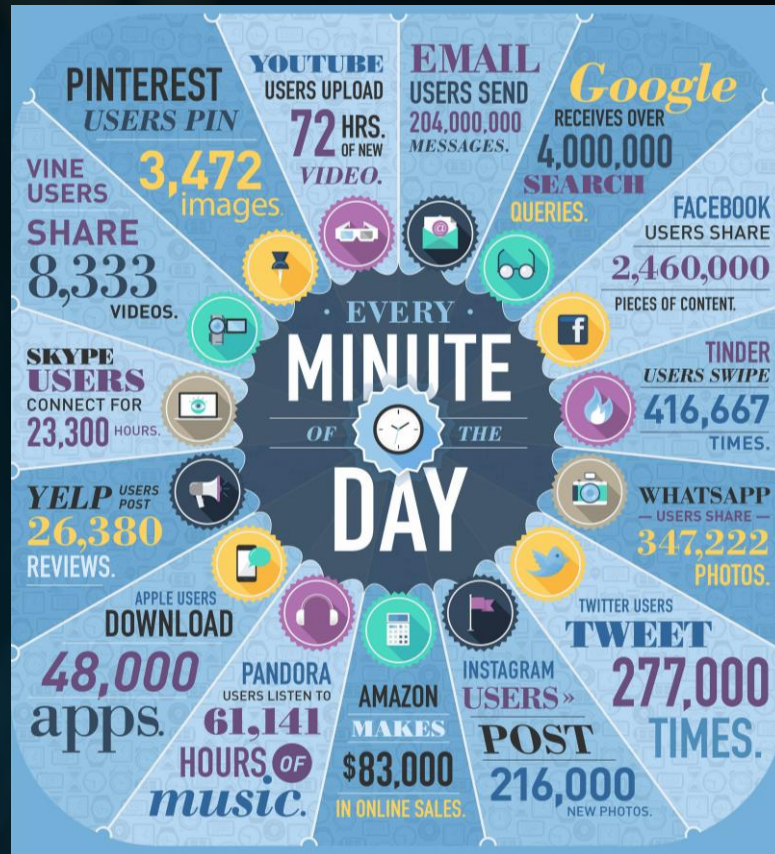


INTRODUCTION TO DATABASES

Today's Agenda:

- The Different Types of Database
- Introduction to Relational Databases
- Database Management Systems (DBMS)
- Introduction to SQL (Structured Query language)

DATA NEVER SLEEPS



2013



2023

DATA NEVER SLEEPS

The world's internet population continues to grow significantly year-over-year. As of November 2023, the internet represents 5.2 billion people—approximately 64.6% of the global population. According to Statista, the total amount of data predicted to be created, captured, copied, and consumed globally in 2023 is 120 zettabytes, a number projected to grow to 181 zettabytes by 2025.

Global Internet Population Growth (IN BILLIONS)



A database is
an organized
collection of
data, that
models some
aspect of the
real world



SIMPLE DATABASES

Birthdays

Name	Birthday
Grace Hopper	09/12/1906
Alan Turing	23/06/1912
Larry Page	26/03/1973
Charles Babbage	26/12/1791
Ada Lovelace	10/12/1815
Frances E. Allen	04/08/1932
Tim Berners-Lee	08/06/1955

Contacts

Name	Telephone No.
Grace Hopper	07441013245
Alan Turing	08675672347
Larry Page	07994501239
Charles Babbage	08662017845
Ada Lovelace	07552098107
Frances E. Allen	07449086098
Tim Berners-Lee	05541023971

AN EXAMPLE

- Let's say you open an online store selling socks.
- And you need to be able to keep track of all your sales and stock.



THE SOCK COMPANY

SOCKCOMPANY.CO

SOCK COMPANY SALES

Stored in one flat file

Name	Address	Date	Product	Qty	Price
Matthew Bates	17 Grasmere Street	12/02/21	Motorbike Socks	2	20.00
Jonathan Edwards	401 Ullswater Road	13/02/21	Unicorn Socks	1	12.50
Halima Kure	297 Tam Hows Street	14/02/21	Glittery Socks	1	15.99
John Hrycak	12 Windermere Cl	16/02/21	Superman Socks	1	8.50
Edwards, Jonathan	401 Ullswater Road	18/02/21	Fluffy Puppy Socks	2	50.00
Oliver Kerr	45 Rydal Rd	19/02/21	Superman Socks	3	25.50
Julie allen	96 Derwent Road	20/02/21	Ninja Turtles Socks	1	7.95
J. Edwards	401 Ullswater Road	27/02/21	Ninja Turtles Socks	1	7.95
C. Fasching	126 Ennerdale Avenue	01/03/21	Glittery Socks	1	15.99
John King	1 Palace Road	02/03/21	Tartan Socks	1	12.50
Jonathan Edwards	401 Ullswater Road	04/03/21	Fluffy Bunny Socks	3	75.00
Folayo Aina	123 Buttermere Street	05/03/21	Unicorn Socks	1	12.50
j. Edwards	401 Ullswater Road	08/03/21	Cute Cat Socks	1	15.99

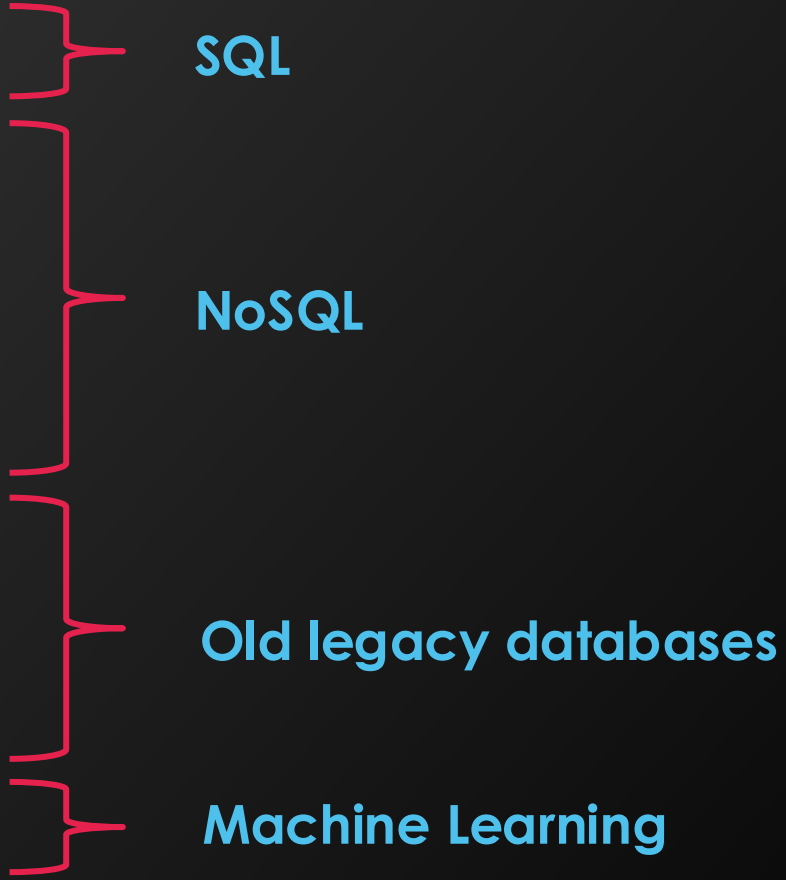
RELATIONAL VS NON-RELATIONAL DATABASES

There are basically two types of database: relational and non-relational (or SQL vs NoSQL)

Relational databases are the most popular, however, non-relational databases have become very popular over the last few years due to the growth in big data platforms, such as social media platforms.

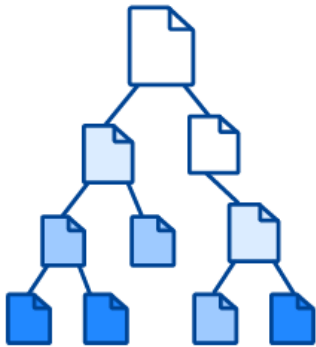
Many large companies now use a combination of both.

DATABASE MODELS

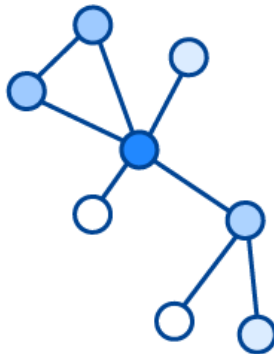
- Relational Databases
 - Document / Object
 - Key-Value
 - Wide-Column
 - Graph
 - Hierarchical
 - Network
 - Multi-value
 - Array / Matrix / Vectors
- 
- SQL**
- NoSQL**
- Old legacy databases**
- Machine Learning**

NON-RELATION (NoSQL)

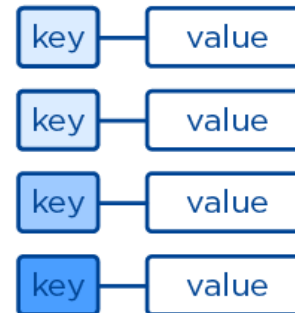
Document



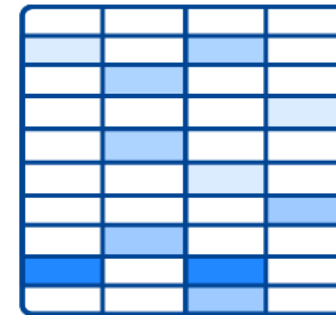
Graph



Key-Value



Wide-column

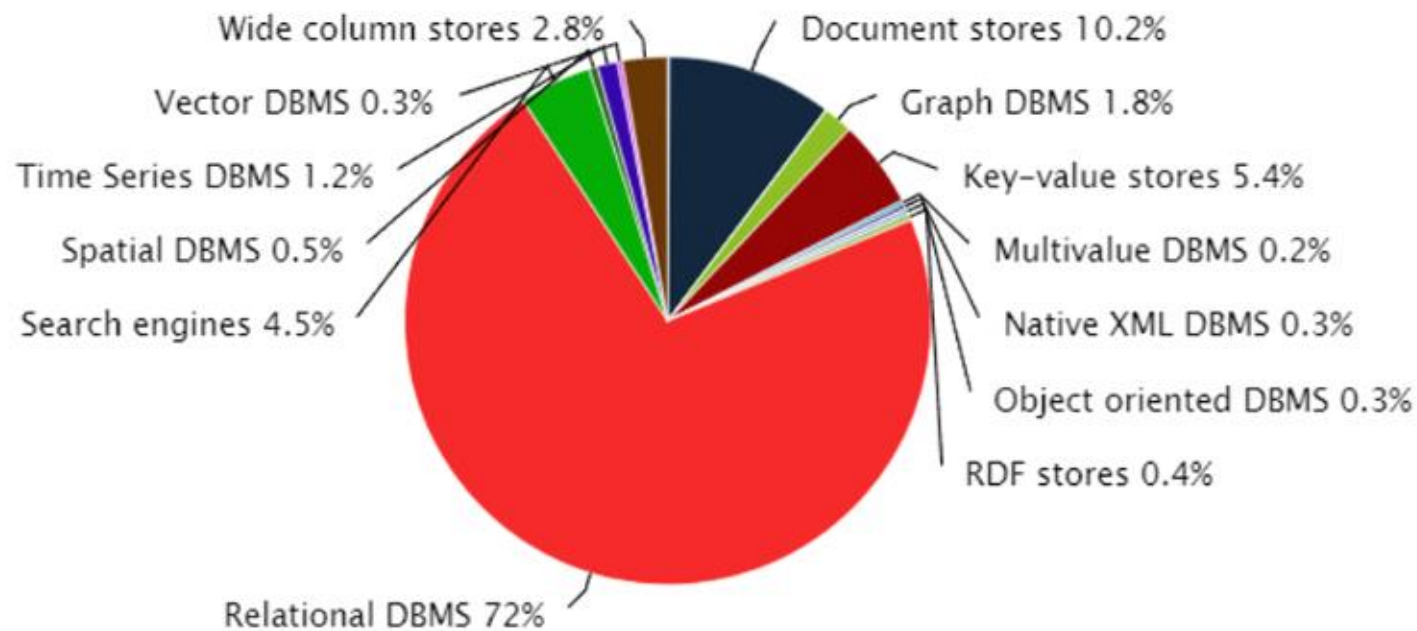


RELATIONAL DATABASES

- Relational databases organise data into tabular structures called relations.
 - Similar to a spread sheet in excel
 - Each field becomes a column in a table
 - Each record becomes a row
- Information is grouped into separate tables of related data
- Tables are linked together using keys

		A Column	
A row			

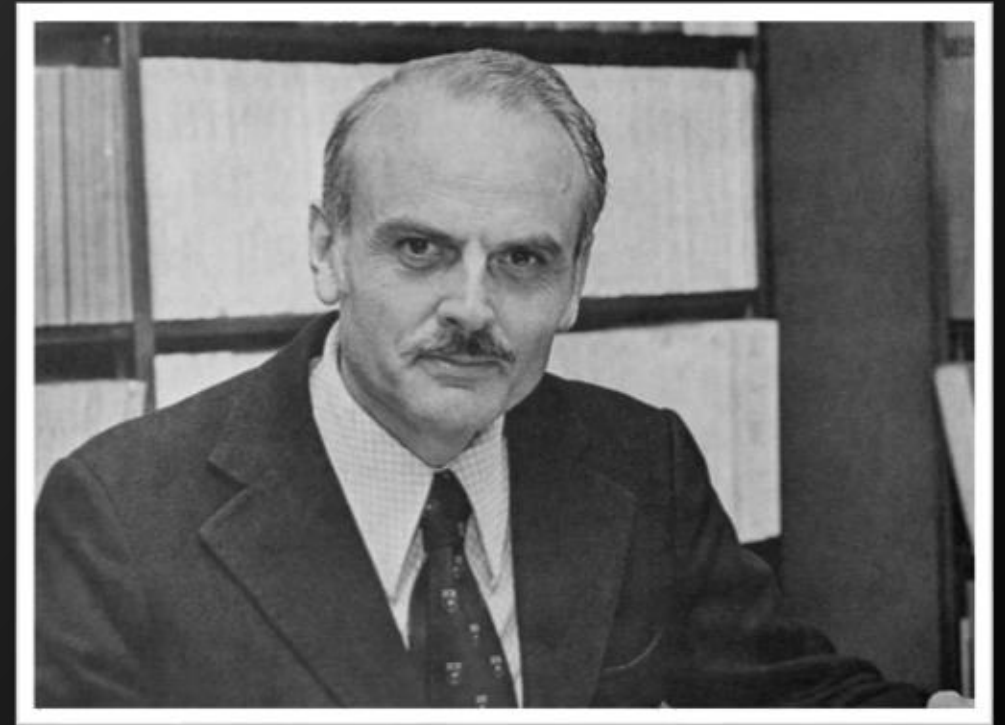
DATABASE MODELS: RANKING



The ranking shows the popularity of each model as of January 2024.

RELATIONAL DATA MODEL

- The term Relational Database was introduced in 1970 by Dr. Edgar Frank Codd from whilst he was working for IBM
- He developed **12 basic rules** which he said should be applied to database models
- The original relational database was known as '**System R**'



RELATIONAL DATABASES

Customer

CustomerID	FirstName	LastName	DOB	Address	TelNo
101	Charles	Babbage	26/12/1965	10 Long Road	07457237833
102	Grace	Hopper	09/12/1982	7 Short Road	09341908664
103	Tim	Berners-Lee	08/06/1979	9 High Street	07966234982
104	Ada	Lovelace	10/12/1991	3 East Avenue	09553012787

← Attributes

} Tuples (rows)

TERMINOLOGY

- **Table** and **Relation** mean the same thing.
- **Row**, **Record**, **Tuple** all mean the same thing.
- A **Tuple** (row) has a set of **Attributes** (columns) containing values

PRIMARY KEYS

- Each Table has a **Primary Key**.
- The primary key value for each tuple (record) is unique.
- Primary keys ensure that each row in a table is unique.
- When you create a table you usually define a primary key, e.g. customerID, studentNO, productId.
- Alternatively, the DBMS can usually create a primary key field for you.

CustomerID is
the Primary
Key for our
student table

CustomerID	FirstName	LastName	DOB	Address	TelNo
101	Charles	Babbage	26/12/1965	10 Long Road	07457237833
102	Grace	Hopper	09/12/1982	7 Short Road	09341908664
103	Tim	Berners-Lee	08/06/1979	9 High Street	07966234982
104	Ada	Lovelace	10/12/1991	3 East Avenue	09553012787

FOREIGN KEYS

A Foreign Key is an attribute (or set of attributes) in one table that links to the Primary Key of another table – this establishes the connection between the tables.

Customer

CustomerID	FirstName	LastName	DOB	Address	TelNo
101	Charles	Babbage	26/12/1965	10 Long Road	07457237833
102	Grace	Hopper	09/12/1982	7 Short Road	09341908664
103	Tim	Berners-Lee	08/06/1979	9 High Street	07966234982
104	Ada	Lovelace	10/12/1991	3 East Avenue	09553012787

Order

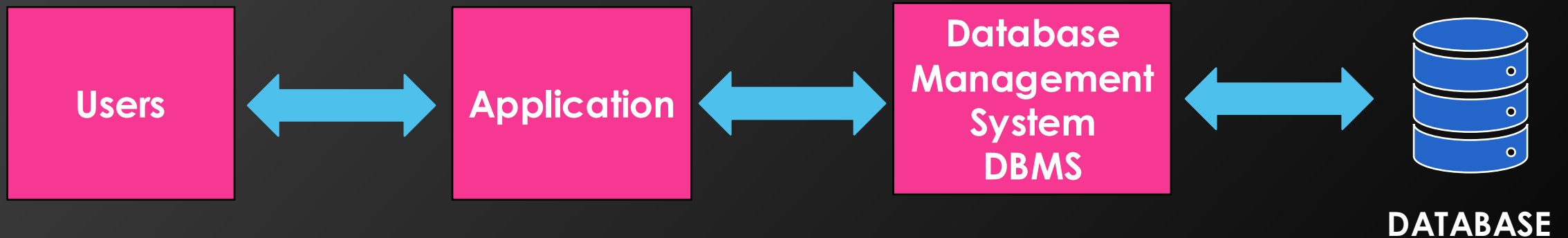
OrderID	CustomerID	OrderDate	Status
3001	104	19/12/2023	Shipped
3002	101	20/12/2023	Shipped
3003	101	20/12/2023	Packing
3004	103	21/12/2023	Pending

In the Customer table CustomerID is the Primary Key

This refers to CustomerID in the Orders table which is the Foreign Key

DATABASE MANAGEMENT SYSTEM (DBMS)

A software application which is used to create and manage databases.



FUNCTIONS OF THE DBMS

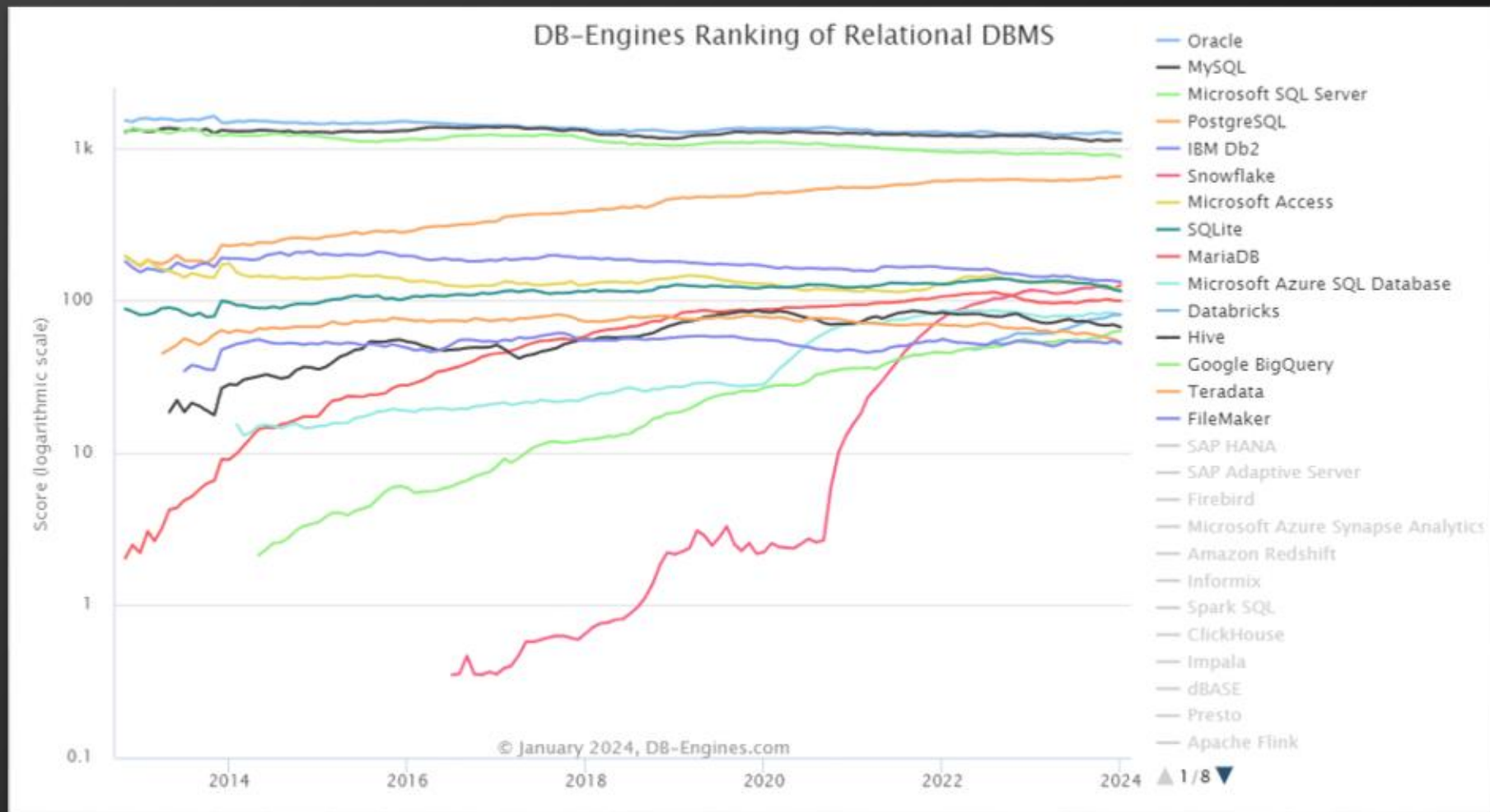
- DBMS serves as the intermediary between the applications and the database
- Manages and controls all database activities:
 - Creates databases and tables
 - Data storage, retrieval and updates
 - Enforcing rules: integrity constraints, user access rights, etc.
 - Concurrency control
 - Back-up and recovery
 - Providing security
 - Etc.

RELATIONAL DATABASE MANAGEMENT SYSTEMS



And many more

RELATIONAL DATABASE MANAGEMENT SYSTEMS



SQL

- Stands for *Structured Query Language*
- The standard language for interacting with databases
- There are many different Relational Database Management Systems. The most popular are: MySQL, MS SQL Server, Oracle, MS Access, PostgreSQL, etc.
- They all use SQL, however, there are some variations in syntax here and there.



<https://i.imgur.com/NtGaNA8.png>



<https://brothersincode.com>

SQL – CATEGORY OF LANGUAGE

- **DDL – Data Definition Language**
The part of SQL for processing the **structure** of a database
- **DML – Data Manipulation Language**
The part of SQL for processing the **data** in a database
- **DCL – Data Control Language**
Provides database security and manages user access rights
- **Querying**
Statements used to retrieve data based on specific criteria

DDL: DATA DEFINITION LANGUAGE

DDL statements change the structure of the database. There are three basic commands:

CREATE: Creates new tables in the database

ALTER: Changes the structure of an existing table

DROP: Removes a table from the database

CREATING TABLES

```
CREATE TABLE tablename  
    ( attributename      data type,  
      attributename      data type,  
    );
```

Attributename may be any **alphanumeric name** starting with an alphabetic character. The name may also contain '\$' '-' '_' '#' '@' (maximum 30 characters)

*Valid data types are listed at the end of the slides.

ALTER TABLES

- To **ADD** columns:

```
ALTER TABLE table_name  
ADD ( column_1 column-definition, ...  
      column_n column_definition );
```

To **MODIFY** columns

```
ALTER TABLE table_name  
      MODIFY ( column_1 column_type, ...  
              column_n column_type );
```

To **DROP** columns

```
ALTER TABLE table_name  
      DROP COLUMN column_name;
```

DROP TABLES

To delete(DROP) a whole table use the following command:

```
DROP TABLE table_name;
```

* obviously this should be done with caution

DML: DATA MANIPULATION LANGUAGE

This is the section of SQL which handles data manipulation...
i.e. inserting, updating and deleting rows in tables. It consists of three basic statements:

INSERT: Inserts new records into a table

DELETE: Deletes existing rows in a table

UPDATE: Updates existing rows in a table

INSERTING ROWS INTO A TABLE

INSERT INTO tablename VALUES (value1, value2,.....);

Inserts ONE complete row into the table. Values in the list **must** be in the same order as the columns in the table and there must be a value for each column:

```
INSERT INTO customer  
VALUES(104, 'ADA', 'LOVELACE', '10/12/1991', '3 SHORT AVENUE', '09553012787');
```

CustomerID	FirstName	LastName	DOB	Address	TelNo
101	Charles	Babbage	26/12/1965	10 Long Road	07457237833
102	Grace	Hopper	09/12/1982	7 Short Road	09341908664
103	Tim	Berners-Lee	08/06/1979	9 High Street	07966234982
104	Ada	Lovelace	10/12/1991	3 East Avenue	09553012787

Customer

INSERTING ROWS INTO A TABLE

If you do not have values for all the columns, a list of columns may be specified, and values provided in the same order as the specified columns:

```
INSERT INTO customer (customerID, firstname, lastname, telno)  
VALUES(104, 'ADA', 'LOVELACE', '09553012787');
```

The unspecified columns will be set to NULL.

CustomerID	FirstName	LastName	DOB	Address	TelNo
101	Charles	Babbage	26/12/1965	10 Long Road	07457237833
102	Grace	Hopper	09/12/1982	7 Short Road	09341908664
103	Tim	Berners-Lee	08/06/1979	9 High Street	07966234982
104	Ada	Lovelace	-	-	09553012787

Customer

DELETING ROWS FROM A TABLE

DELETE FROM tablename WHERE [conditional statement];

DELETE FROM CUSTOMER WHERE LASTNAME = 'Hopper'

...deletes the second row in the table below

DELETE FROM CUSTOMER

...deletes ALL rows from the customer table

CustomerID	FirstName	LastName	DOB	Address	TelNo
101	Charles	Babbage	26/12/1965	10 Long Road	07457237833
102	Grace	Hopper	09/12/1982	7 Short Road	09341908664
103	Tim	Berners-Lee	08/06/1979	9 High Street	07966234982
104	Ada	Lovelace	10/12/1991	3 East Avenue	09553012787

Customer

QUERYING

Basic statement construction:

SELECT [columns] FROM [table];

With search criteria (WHERE clause):

SELECT [columns] FROM [table] WHERE [search criteria];

SELECT firstname, lastname FROM [customer] WHERE customerID = 103

CustomerID	FirstName	LastName	DOB	Address	TelNo
101	Charles	Babbage	26/12/1965	10 Long Road	07457237833
102	Grace	Hopper	09/12/1982	7 Short Road	09341908664
103	Tim	Berners-Lee	08/06/1979	9 High Street	07966234982
104	Ada	Lovelace	10/12/1991	3 East Avenue	09553012787

Customer

SQL DATA TYPES

CHAR(size): consists of upper- and lower-case letters, numbers and special characters (+,-,%, \$,&, etc.) size the maximum length, in characters, of the column. May not be larger than 255.

VARCHAR(size): Variable length character string data type. Only stores the actual length of the data field, does not space fill to size specified. size must be specified. Can distinguish between NULL and an empty string.

VARCHAR2(size): Same as VARCHAR, except does not distinguish between NULL and an empty string.

LONG: Character data of variable length up to 65,535 characters. Only one long column per table.

NUMBER: Number values consisting of digits 0 -- 9, values may be 38 digits wide.

NUMBER(w,d): Number values with decimal places specified
w - specifies the total number of digits
d - specifies the number of decimal places.
(i.e. (4,2) will allow a max. number of 99.99 to be inserted.)

DECIMAL(w,d): Stores numbers with up to 22 decimal digits.

INTEGER: Stores numbers with 10 or fewer digits, digits to the right of the decimal point are truncated.

SMALLINT: Stores digits with 5 or fewer digits.

DATE: Date values - usual form dd-mmm-yy i.e. 30-OCT-19



ORACLE APEX DEMO

RESOURCES

- [W3 Schools - SQL Tutorials](#)
- LinkedIn Tutorials:
 - [Understanding relational databases \(linkedin.com\)](#)
 - [Learning to use MySQL](#) (although we are not using MySQL there is a lot of general database and SQL content here)
- Richie, C. (2008). *Database Principles and Design*. 3rd Ed. Thompson Learning