# 8/25/17

* Bit, byte, kb, mb, gb, tb
* SQL = Structured Query Language

# 9/6/17

* Sql compiles to relational algebra
  + If it is written incorrectly, then the code is wrong.
* Relational Model of the DataBaseManagementSystem
  + Very simple
  + Attribute/column/field – also degree
  + Tuple/row/record – also cardinality
* Entity Relationship Model
  + UML: unified model language
  + Object Oriented programming
* What is SQL?
  + Declarative
    - Say what you want without specifying how
  + Has many standards
    - ANSI SQL
    - SQL-92/SQL2
    - SQL-99/SQL3

# 9/14/17

* Includes multiple operators
  + Let R be a relation and t be a tuple with the same set of attributes
    - EXISTS R is true if and only if R contains at least one tuple
    - t IN R is true if and only if t equals a tuple R
    - t > ALL R is true if and only if R is unary (has one attribute) and t is greater than every value in R
      * can use any comparison operators
      * if we use <>, r need not be unary
    - t > ANY R (which is unary) is true if and only if t is greater than at least one value in R
    - We can use NOT to negate EXISTS, IN, ALL, and ANY.

# 9/22/17

# 10/4/17

* SQL is based off bag because removing duplicates is expensive.
* Features of transactions:
  + Concurrent access
    - Multiple users can access it at the same time.
* ACID
  + *Atomic*: Whole transaction or none is done.
  + *Consistent*: Database constraints preserved.
  + *Isolated*: It appears to the user as if only one process executes at a time.
  + *Durable*: Effects of a process survive a crash.
* Views
  + Virtual
  + Materialized
* SQL Uses B-Tree
* How things are paired
  + 1-1, many-1, many-many

# 10/25/17

|  |  |  |
| --- | --- | --- |
| **A** | **B** | **C** |
| 1 | 5 | 3 |
| 2 | 4 | 4 |
| 1 | 5 | 3 |
| 2 | 5 | 3 |

A 🡪 C🚫

C 🡪 A🚫

B 🡪 C✅ C 🡪 B ✅

B 🡪 A🚫

Inputs are locked to outputs but outputs aren’t locked to inputs

# 11/3/17

* Normalization (ch 10ish)
  + Motivation: remove redundancy & anomalies
  + FD (Functional Dependency)
  + Key:
    - Unique
    - Minimal
  + Tivial FD:
    - AB 🡪 B
    - AB 🡪 BC
  + Inference Rules
    - Armstrong’s Axiom:
      * Reflexivary
        + AB 🡪 A, A 🡪 A, B 🡪 B….
      * Augmentation:
      * Transitionary:
        + A 🡪 B, B 🡪 C, A 🡪 C
  + FD Closure:
    - Every FD holds in the table oFD
* Closure of attributes
  + 11.7

# 11/8/17

* trivial dependencies are always true.
* Prove vs. transitivity