CS Fundamentals

- DBMS
- OS
- CN

LLD

HLD

Sumeet Malik

DBMS

OS

CN

Theory

DBMS

1. Intro to DBMS
2. Schema Design

SQL
3. CRUD
4. Joins
5. Aggregate & Builtin

6. Transactions &
   Indexing , ACID

SQL
7. UDFs & Stored Procs.
8. Triggers, Subquerying

OS → 6 lectures

CN → Socket Programming

Agenda
1. Data, Database, DBMS
2. Type of DBMS
3. Relational DBMS
   └ Terms

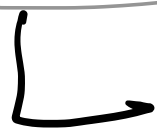$\llcorner$ Properties

4. Keys
5. SQL

DBMS

Data?   Database?

Collection of Data.

DBMS [A software system to
help store & retrieve
data]

Types of DBMS

$\llcorner\longrightarrow$ File System work as

a dbms?

```
┌─────────────────┐   ┌─────────────────┐
│ Scalar_Class    │   │ Scalar_Sales    │
├─────────────────┤   ├─────────────────┤
│ Students. txt   │   │ Leads. txt      │
│ Batches. txt    │   │ Sales Calls. txt│
├─────────────────┤   └─────────────────┘
│                 │
│ Instructors. txt│
└─────────────────┘
```

Cons of Using File System

1. Retreival is going to be a problem

```
         Students. txt
┌──────────────────────┐
│ 1, Suneet, sas,   ④  │
│                      │
│ 2. Sathly ,  — ,  5  │
│                      │
│ 3. Arun ,    — ,  6  │
└──────────────────────┘
```

```
        Batches. txt
       ┌──────────────┐
       │ 1   Batch 1  │
       │ 2   Batch 2  │
       │ 3   Batch 3  │
       └──────────────┘
```

2. Inconsistency

3. Duplicacy

4. Security

Dedicated DBMS

✓

Features.
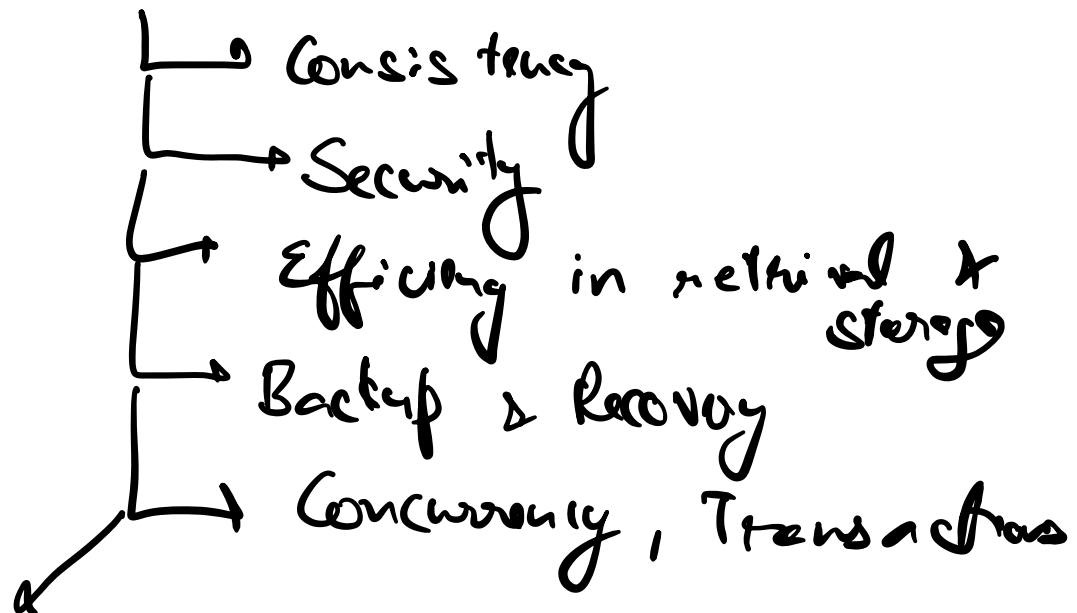
- Consistency
- Security
- Efficieny in retrival & storage
- Backup & Recovery
- Concurrency, Transactions
- Replications, Scaling

Relational DBMS | NoSQL Databases

Graph
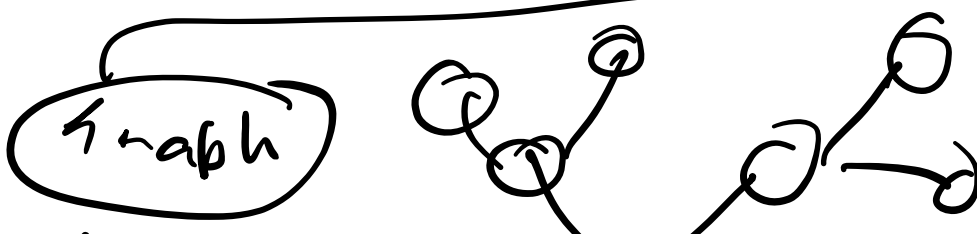
Neo4j
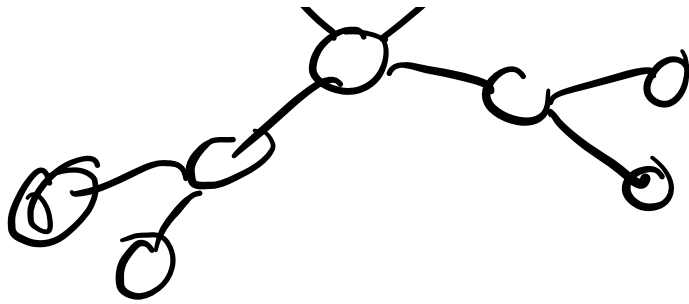
Key-value → Redis

Columns → Cassandra

Document → MongoDB
Oriented
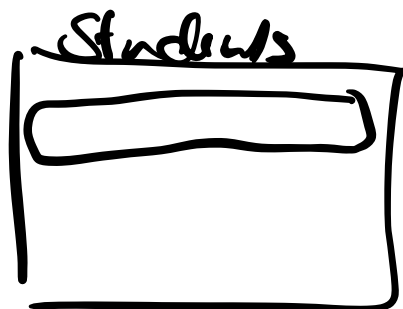
Relational DBMS
↓
Relational Algebra [Set-Theory]

Terms
1. Relation [Table]
   Students
2. Attribute
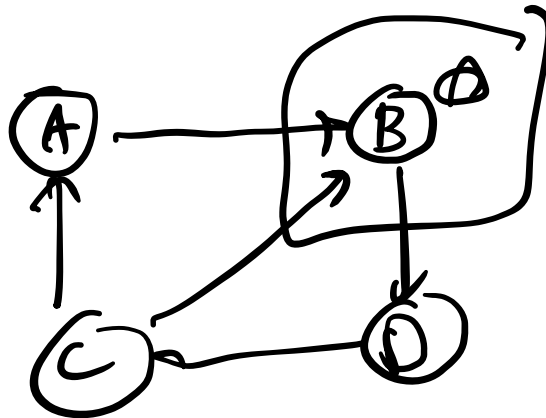   ⌐ Columns
3. Tuple / Entity / Row

| | (1, Sumed, sum.com, 4) |

4. Degree → Count of Attributes
5. Cardinality → Count of Tuples

Properties of Relational Databases

1. Rows are unique
2. Atomicity in values
3. Order of Columns is irrelevant
4. Order of rows don't matter

ID	Sumed	Malik
1		
2	Sumed	Malik

$$(A, A, B, C) = (C, B, A)$$

Students

St-id

first_name

last_name

email

phone_numbers →

T 1

Students 7

id	fn	ln	email	pnos
1	Sam	Malk	sam	TT99 OT

T 2

Students

id	fn	ln	e
1	S	M	sam

Phone

sfid	no
1.	xy2
1.	det
1.	abc

JS

CH

↳ Statically

Student

id	fn	ln	email
1	A	B	a2b
2	C	D	c2d
3	E	F	e2f

Select *
from
Student

email

id	email	ln	fn
2	c2d	D	C
3	c2f	F	E
1	a2b	B	A
0	1	2	3

2 - 0
2

tbl [0] ~~[email]~~   tbl [ ] [email]

a	b	c

c	a	b

Select * from Students

Keys

Relation
Tuple
Attributes

A collection of attributes that
uniquely identifies any tuple

Students

fn	ln	email	phone	ano

1. Super key → Any collection uniquely ident-
the tuple

2. Candidate key → A minimal collection of
attributes that uniquely
identifies a tuple.

3. Primary key

4. Composite key

5. Foreign key

Every reln. has
exactly 1 Primary key

	super key	candidate key
fn	✗	✗
fn, ln	✗	✗
fn, ln, email	✓	✗ [fn, ln]
fn, ln, email, pno	✓	✗ [fn, ln, email]
email, pno	✓	✗
email	✓	✓
pno	✓	✓
email, an	✓	✗

Students

st-id	fn	ln
1	a	b
2	c	d

Batches

b-id	bname
1	b1
2	b2
3	b3

student-batch

s-id	b-id	b-date
1	1	
	3	
2	1	

② srd or bid
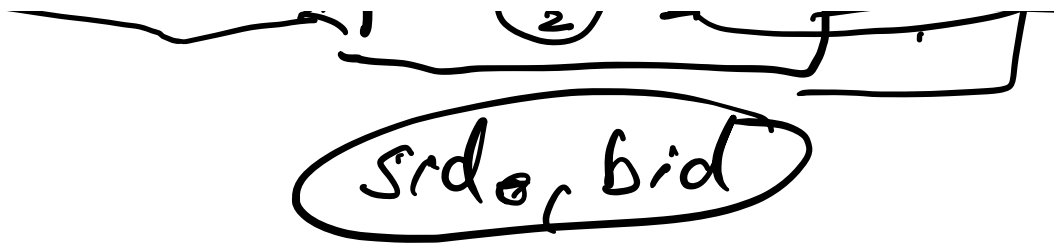
Primary
Composite
Foreign

10:39 - 10:49