Database

- collection of related data

Agenda

✓ Relational model

✓ ① Properties

✓ ② Keys

✓ SQL

- Schema Design

① K-V {

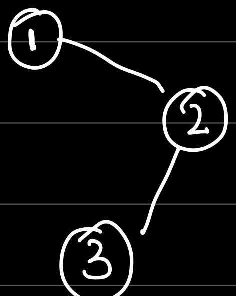
key : value

key1 : value1

}

Redis

②



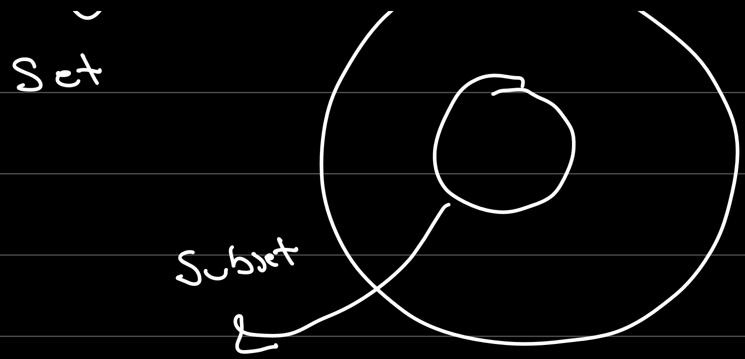
Neo4J

DATA MODEL

Relational Model

J,

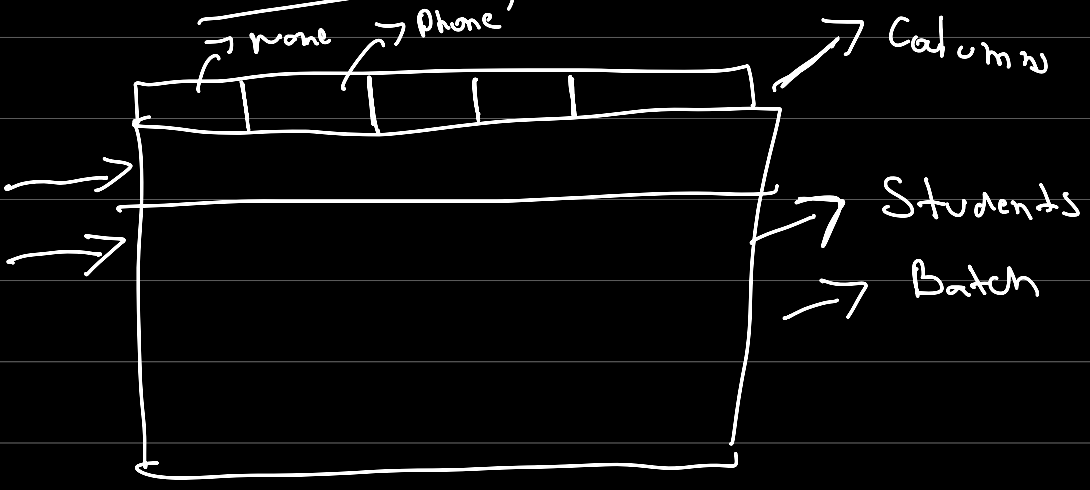
< 25



$\{1, 2, 3, 4\}$

$\downarrow$

$\boxed{\{1, 2\}}$ SUB - SET



Properties

① Uniqueness

→ Rows

→ Column

② $\boxed{\{1, 2, 3\}} = \boxed{\{1, 3, 2\}}$

Unordered

① Rows

② Columns

1

3

2

Age

③ 21 Uniform

Age data type

④ Atomic - no collection

- single column

id	Email
A	A @ B.com
→ B	[b @c.com, c@ D.com]

[1, ②, ②, 3, 4]

↓

{1, 2, 3, 4}

Keys

Each row should be uniqe

1	Tantia	25
2	Tantia	25

One value which uniquely identifies row

↓

KEY

id	Name	Age	Phone	Email
1	Tantia	25	1	a@a
2	Sherlock	26	2	b@c

Super Key

- A set of columns that are a key
  - id
  - phone
  - email
  - id, name
  - id, age

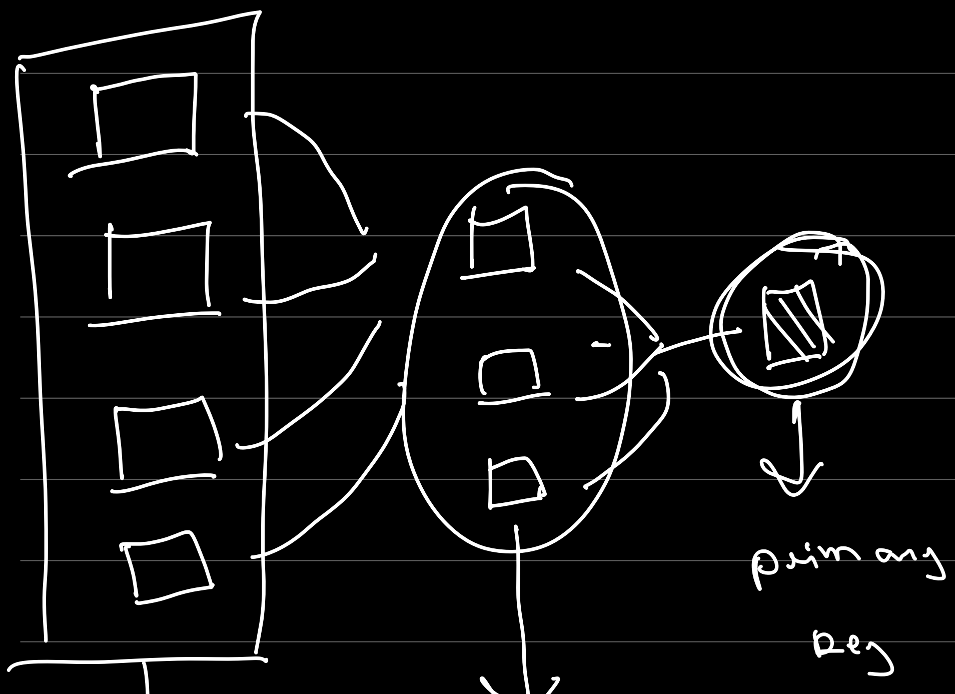
Candidate Key

- minimal set that can uniquely idⁿ a row

{id, name} X

{age, phone} X

{name} X



Super

Candidate (minimal)

{ phone, email }

SK ✓

CK ✗

id

email

phone

i

Key →

Super Key → id   (~~id~~, phone)

(id, name) ✗

(id, ~~name~~, ~~age~~) ✗

(id, phone, ~~name~~, ~~age~~) ✗

Candidates → id

(minimal)   phone

email

Primary Key

Super Key

{~~phone~~, email}

SK ✓

CK ✗

(ID, ~~name~~)

SK ✓

CK ✗

- id
- phone
- id, name

id	name	phone	age
1	Tantra	+91 12	
2	Sherlock	+91354	

Candidate ✗ {id, phone} ✗

- minimal

(id, name) ✗

(email, phone) ✗ ✗

Student

id | name | batch

Schema Design

↓

blueprint

Tables

└ Columns

└ PK }
└ FK }

Design Document

TD

└> Schema design

Requirements

Scale

Schema
└> Blue print

Requirements
└> Entities
→ Nouns

① There will be students with a roll no
↓

Class Assignmt

(Batch | student)

Assignment

(student | type)