RDBMS

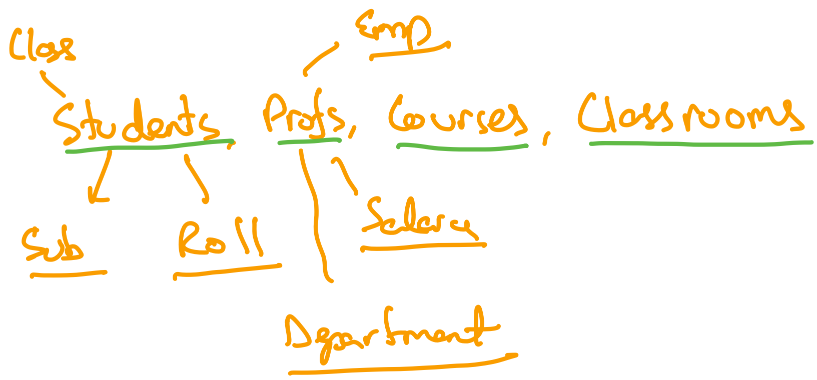
- What is 'relational', intro to RDBMS
- ER Diagram → Entity, Relations, Cardinality, modon Reps
- Tables, dB → Attributes, index, keys
- Example
- DCC → noSql, How to choose b/w noSql & RDBMS

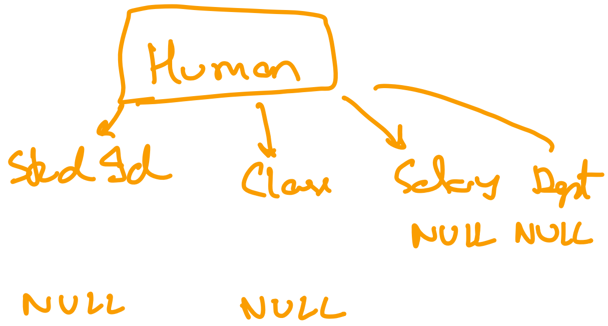
II Normalization

III Indexing

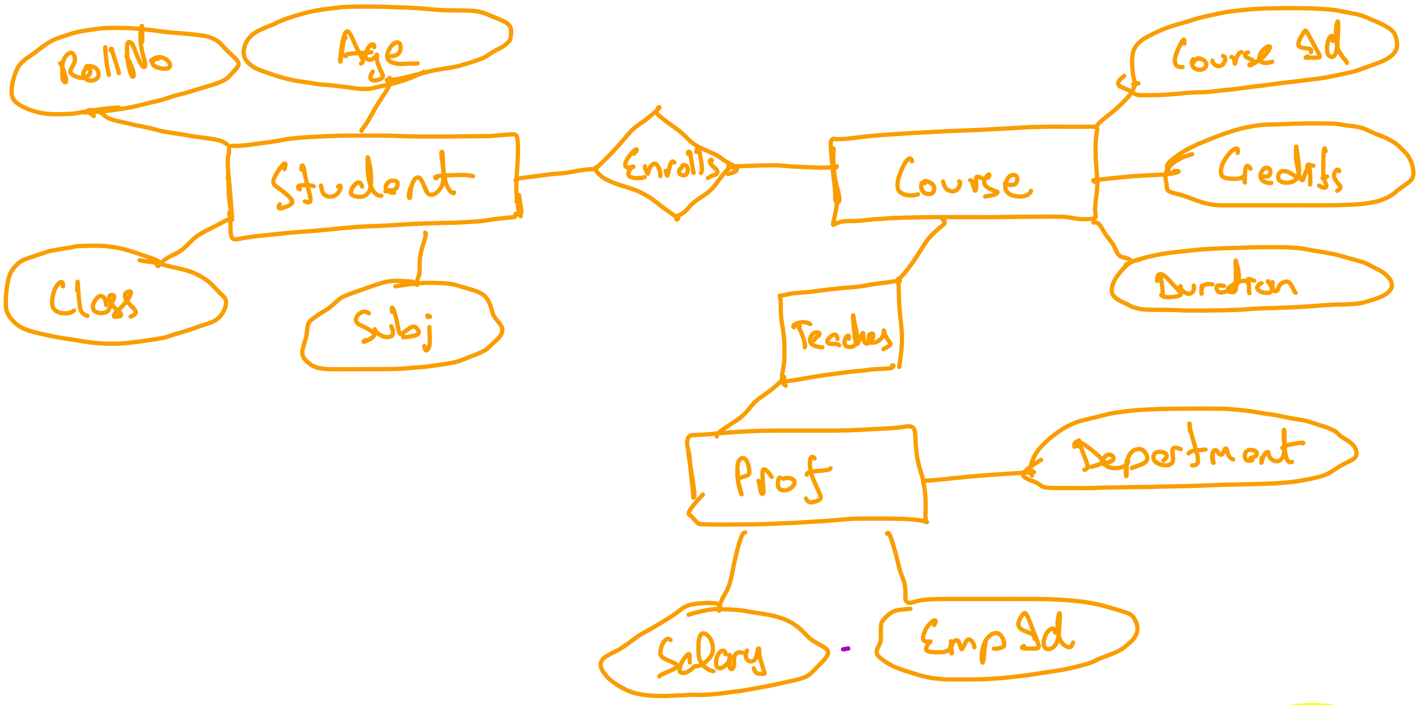
IV Sql Query

Data Model of a school





Entity → logical group of data s.t all members of that entity have the same properties



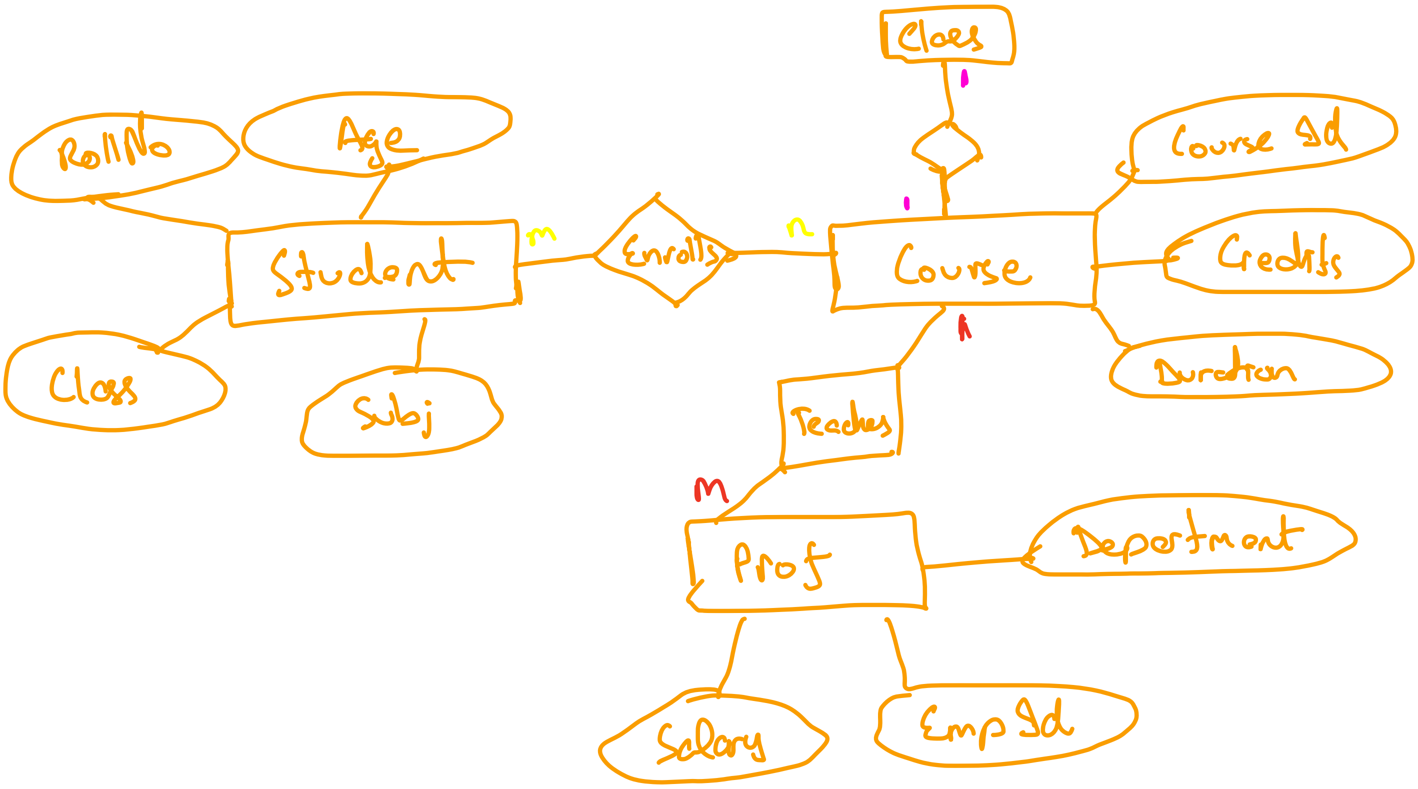
Bevesh
DSA
CN
Ab
Ajith

Q : Students enroll in many courses

→ Profs teaches only 1 course

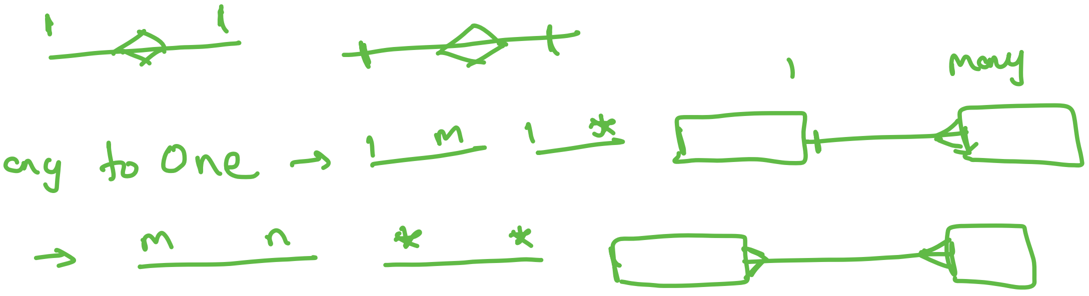
→ Courses can be taken by many students & many profs

Course takes place in only 1 class & vice verse

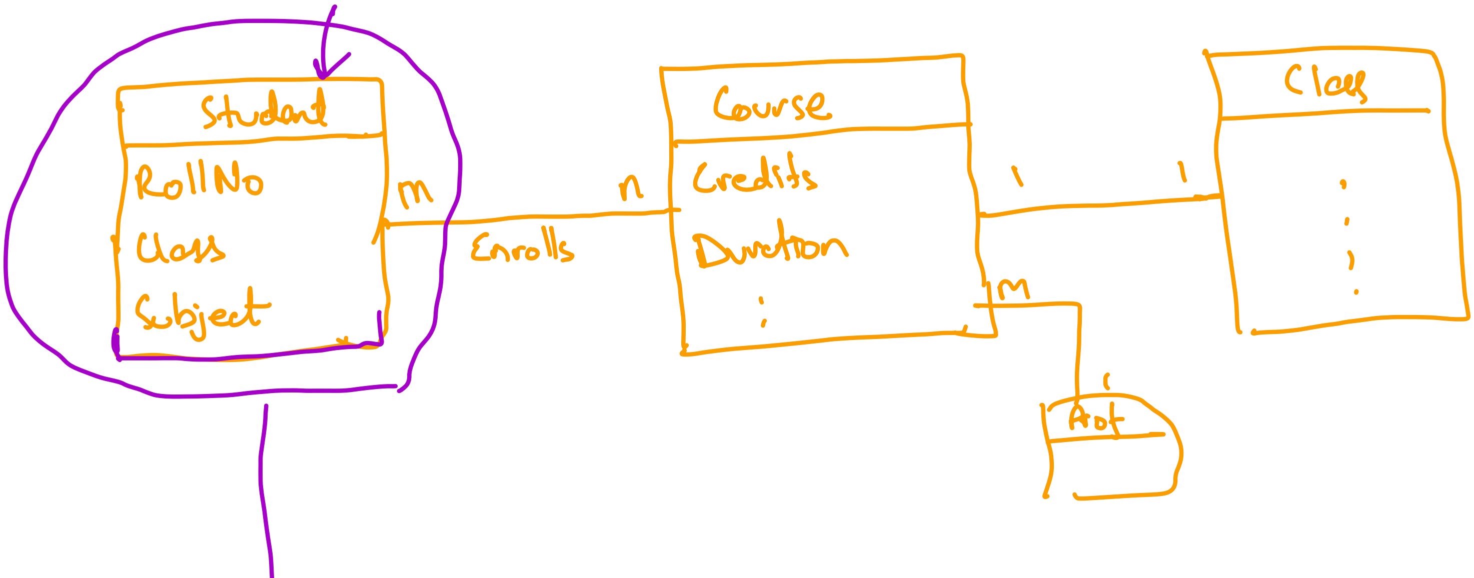


Cardinality

- One to One →
- One to Many / Many to One →
- Many to Many →



Modon



Student :

Head	RollNo	Class	Name	Age	Subject
	10	X	Devesh Singhal	25	DSA | CN | DBMS
	11	XII	Sankalp Soni	26	CN | OS
	12	–	–	–	–

Prof

PId	PName	sAge	Department	Salary	
2621	Andrew Ng	52	AI	10 mill	→ row / Tuple
5134	Geoff Hinton	60	ML	9 mill	
6228	Picaso	210	Art	50 mill	

Courses

CId	CSubject	CCredits	CDuration	CIsThesis
0	Psychology	4	0.5	0
1	Electronics	4	1	1
2	MPAE	4	2	0

Table → Logical grouping of data, which has fixed attributes & a structured fixed schema.

Schema → layout / structure of something
→ Tables → attributes + Info
→ db → Tables + Tables schema

Types of Attributes

Student :

	RollNo	Class	Name	Age	Subject	doB
data types	number	text	text	number	text	Date
	10	X	Devesh Singhal	(currD - doB)	DSA | CN | DBMS	:
	11	XII	Sankalp Soni	26	CN | OS	!
	12	-	-	-	-	

- Simple -> Atomic, cannot be broken down, generally queried ...
- Composite -> Can be broken down, generally queried as a whole
- MultiValued -> Can be broken, not generally queried as a whole
- Derived -> Columns derived from other columns

	RollNo	Class	Name	Age
->	10	X	Devesh Singhal	25

Subjects

SRollNo	Subject
10	DSA
10	CN
10	OS

Normalization

Key → sets of columns used to id rows in dB

Student

RollNo	F Name	Age	Address	Balance	LName
→ 15	Devesh	25	H1	100	
→ 16	Devesh	30	H2	200	
17	Harsh	25	H1	50	
→ null	Rachit	30	...	...	

RollNo ? CK ?

Super Key

set of columns which can uniquely id a row

<RollNo> ✓

<RollNo, Name, Age> ✓ →

< Name, Age> ✓ →

Candidate Key (allows 1 null)

minimal super key

→ <Name, Age> ✓ → composite CK

<Roll, Name, Age> ✗

<Roll> ✓ → simple CK

Alternate Key

Candidate which didn't get picked to by PK

Primary Key (not null)

↳ Candidate key which you select as PK
↓
(generally) index

<FName, LName, Age, Add> → PK, index
↳ Composite PK

Foreign Key

↳ key in a table that is the PK of some other table

Student

PK RollNo	Name	Age	Address
15	Devesh	25	H1
16	Devesh	30	H2
17	Harsh	25	H1

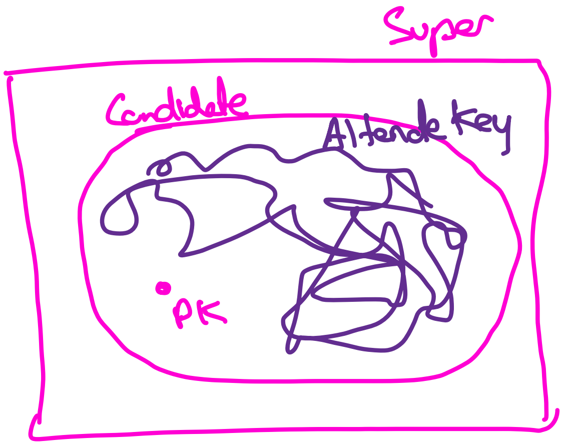
Student - Course

FK RollNo	FK CourseId
15	DSA
15	CN
15	OS
16	DSA
16	DBMS

PK → <RollNo, CourseId>

Course

PK CourseId	Credits	Duration
DSA	-	-
CN	-	-
OS	-	-



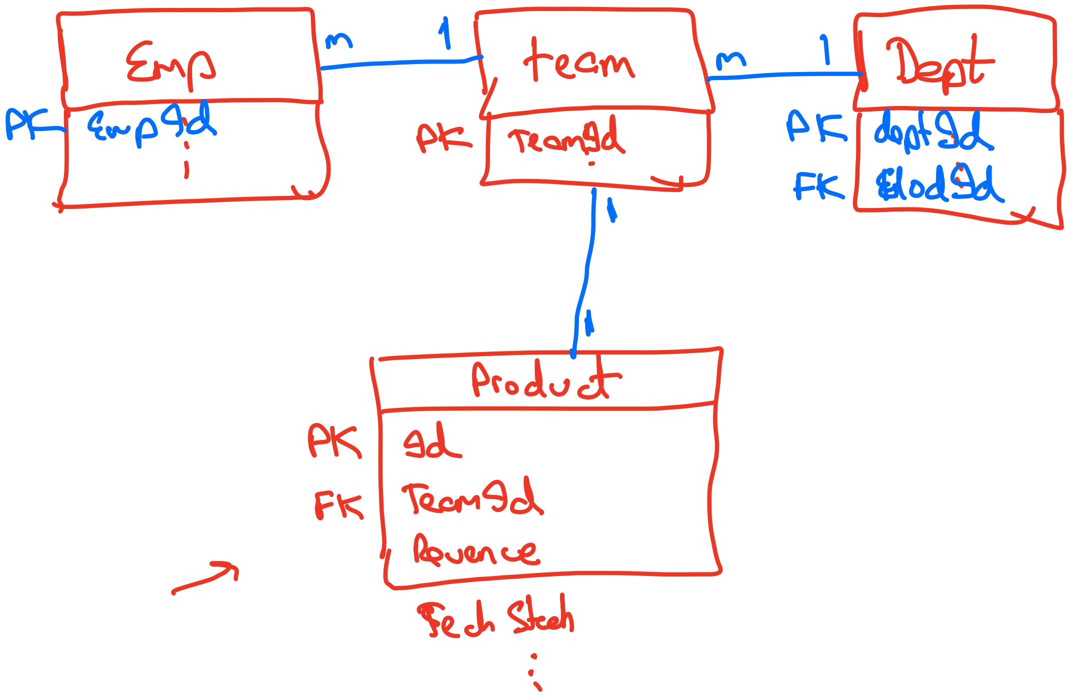
Startup

↳ Emp work in teams

↳ Teams belong to depts

↳ Each dept has HOD

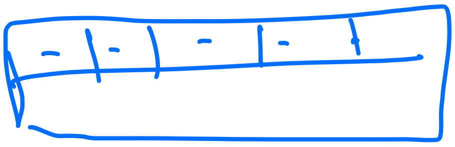
↳ Each team owns 1 product



DCC

Amazon → Products → chair , GPU , Sugar

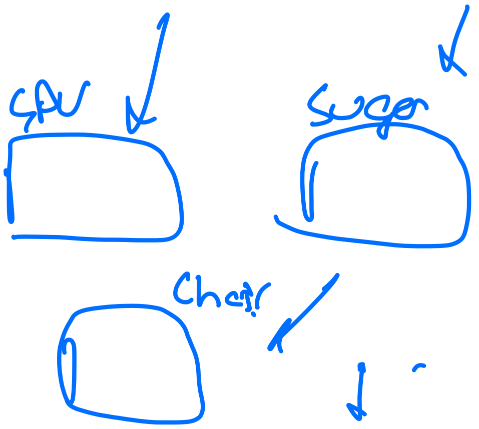
Product



Product

PId	Price	Stock	PJSON

→ { weight
Height
}



{ series → 1080
vram
clock
}

Multivalued Attribute

{ productId : ______
Price : ______
Stock : ______
series : ______
vram : ______
clock : ______
},

{ productId : ______
weight : ______
mat : ______
}

IRCTC

train → RDBMS
→ nosql

RDBMS → Data Integrity
→ ACID guarantees

Atomicity
Consistency
Isolation
Durability → RAM vs HDD

noSql → Flexibility

→ BASE

↳ Basically Available

Soft State

Eventual Consistency

→ CAP →

→ Sharding → Efficiency