DATABASE

23 April 2017

Database System Concept

CSE 3103

Last Sem ↝ Rupol sir-এ class নিয়েছেন।
(Nazmus Sakib)

Text books:

1. Database System Concepts, Siberschatz-Korth (6th Edition)
2. Database System: The complete Book, Molina-Ulman (3rd Edition).

Quiz — 4 (Best 3 will be counted)
always Sunday তে Quiz হবে (রোজার আগে 2টি + রোজার পরে 2টি)

Nazmus Sakib
Asst. Professor
Room: 7A01/M, Block-A, AUST.
Websites: sites.google.com/site/nazmussakib009/
Email: nazmussakib009@gmail.com
cell: 01939900271

শনি ও মঙ্গল বিকালে 4টা-5টায় পাওয়া যাবে।

What is a Database?

Database Management System (DBMS)

1টি software যা total data কে নিয়ন্ত্রণ করে। Backup করার ব্যবস্থা থাকে।

Drawbacks of using file systems to store data

24 April 2017

University Database Example

Levels of Abstraction

* physical level (how a record is stored) — describes
* Logical level (কী কী ধরনের Data এবং Relation etc)
* view level (details of data types, information can be hidden)

Instances and Schemas

- Logical scheme
- Physical scheme
- Scheme
- Sub scheme
- Instance
- Physical Data Independence
- Logical Data Independence

Schema

— চিত্র প্রকার ।

— total structure কে schema বলে ।

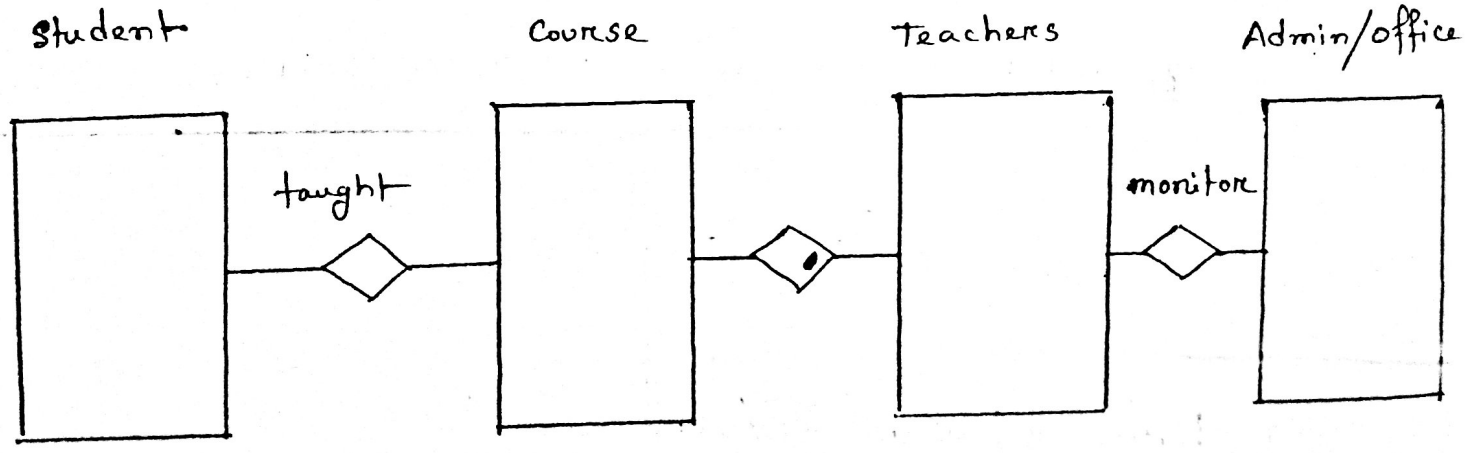


fig: Schema

* Database এর "instance" create করা খুব গুরুত্বপূর্ণ।

* Instance → নির্দিষ্ট moment এ Database এর snapshot.

* Industry level এ "Instance" কে "Snap Shot" বলা হয়।

* Physical level এ কিছু করলে যদি logical level এ কোনো impact না পড়ে, তাকে Physical data Independence বলে।

* logical data independence

** "Level of Database" - short Question আসতে পারে।

Database Users & DBA Activities

(self study)

DBA ⇒ Database Administrator

Data Models (Just জেনে রাখব, xm ও আসবে না)

E-R Model ⇒ Entity-Relationship Model

* 850-1050 MHz frequency এর মধ্যে- যেও এবং primary phone call dedicated path ও।

* ITU ⇒ International Telecommunication Union

* 6s এর মধ্যে- (Max. 7s) Connect হওয়া phone calls.

Data Definition Language (DDL)

Database ও Define করার কাজে ব্যবহৃত হয়- through checking constraints.

Data Manipulation Language (DML)

Database ও বিভিন্ন ইচ্ছামতো search/query এবং মাধ্যমে Data manipulate/control করা যায়, তাকে Data Manipulation Language (DML) বলে।

30 April 2017

Database Engine

3rd part मुख्य 1

Storage Management

Student

ID	Name	Address	D.O.B.
1	A	abc	15 APR
2	B	xyz	12 Sep
3	C	mno	01 JAN
4	D	uvw	19 APR
5	E	stx	18 DEC

View/Show

```
Select Name
FROM Student
where DOB = " ** APR"
```

} Query

A 15 APR
D 19 APR

} Showing Results

in Relational Algebra (Database এর language কোনো)

σ_{name} ^ student

(DOB = ** APR)

* "Optimization" এর main idea হচ্ছে "কম সময়ে বেশি কাজ করা"।

Query Processing

* দুই way তে Query করা যায় ⇒ (i) Top-Down
(ii) Bottom-Top

Transaction Management :

Database Users and Administrators :

Database Architecture

4 প্রকার।

Centralized Database

Client Server Database

ROLL 103466
BOARD DHK
Year 2016

Select *
From Result
where ROLL = "103466"
"DHK" Year = "2016"

* Faster Query

Select *
From Result
where Year = "2016"
Board = "DHK"
Roll = "103466"

Parallel Database

Q. Central ও Parallel Database এর পার্থক্য ?

⇒ Central Database ও direct database ও hit হয়, where Parallel Database ও direct database ও hit হয়ে আবার আলাদা আলাদা layer ও hit হয় এবং পরে central এ এসে sync হয়।

Distributed Database

"RAID"

04 May 2017

Chapter-6

ER Modeling

Steps:

(1) Requirement Analysis: Firstly you need to generate an idea about a project, from there you need to develop a High Level Description which is also called ER Modeling. The symbols which are used

in ER Modeling that are representing the Requirements.

So,

Idea → High Level Description → Relationship schema → (Tables) (DDL/DML)
(ER Modeling)

RDBMS ⇒ Implementation

ER Diagram :

The diagram which represents ER Mode that is ER Diagram.

ERD → Entity Relationship Diagram

There are 3 main parts in ERD.

(i) Entity
(ii) Relationship
(iii) Attribute

There are also many small parts in a ERD.

Entity :

→ It's a noun altime

→ After Requirement Analysis we have to findout the Entity and Relationships.

→ Entity will be a real object from Problem Domain and Data will be stored in DB on Entity.

→ Distinguishable from othere objects.

→ An entity is described using a set of attributes.

Example:

People	Places
Things	Organizations
Concepts	Events

Example:

There are many students in AUST. students can enroll in different courses and receive Grades.

AUST → Specific info

Grades → Values; no Attributes

Student →

Courses →

→ All entities in a entity set have some set of attributes.

→ Each entity set has a key (Underlined)

→ Each Attributes has a Domain.

07 May 2017

Entity set: Collection of Similar Entities.

Entity Type: Collection of Entities that have same attributes.

Example: Student { ID, name, DOB, Phone }

Student

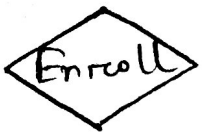
ID	Name	DOB	Phone
1	A	23 June	019
2	B	2 July	017

Relationship:

The interaction between Entity set.

→ Relationship is always a verb
→ It can have own attributes
→ Association among two or more attributes

Example:

There are many [Students] (??) in AUST. [Student] can 

in different [Courses] and receive (??) grades.

Student?

A	B
1	α
2	β

Courses?

C	D
3	x
4	y

Enroll

A	B	C	D	E
1	α	3	x	Yes
1	α	4	y	Yes
2	β	3	x	Yes
2	β	4	x	No

Communication
Interaction
Multiplication
Join

Attributes:

The properties of Entity or Relationship

→ Properties are remaining same for entity but value can change

→ If properties / attributes are described in story, use the given ones. Other than guess the attributes and use the meaningfull properties.

→ student

ID, Name, Parent Name, Address, DOB, Blood Group, Phone, E-mail, Eye color??, Height??, Hair color??, NID, Birth Certificate etc.

Primary Key:

The attribute which can distinguish one Entity from another can be a primary key.

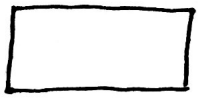
→ Primary key should be underlined in ERD.

→ More than one attribute can form Primary key.

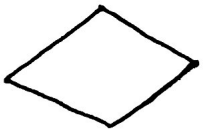
→ Primary key will be minimum number attribute.

→ Primary key can be the attribute which is most frequently use and monthly use to form.

Symbols :

Entity Set :  Rectangle

Relationship :

 Diamond

Attribute :

 Ellipse

Model : Crows Feet
OO ER

* 1টি entity কোনো entity এর সাথে connected হতে হলে তাদের মধ্যে অবশ্যই 1টি Relationship থাকা লাগবে।

* Entity ⇒ noun

Relationship ⇒ Verb

* কোনো 1টি Relationship create হলে ⇒ Table create হবে।

⊛ Entity : Student Courses

Relationship : Enroll

Attributes :

Student

Student ID, Student Name, DOB, Phone, Blood Group

Course

Course ID, Course Name, Credit

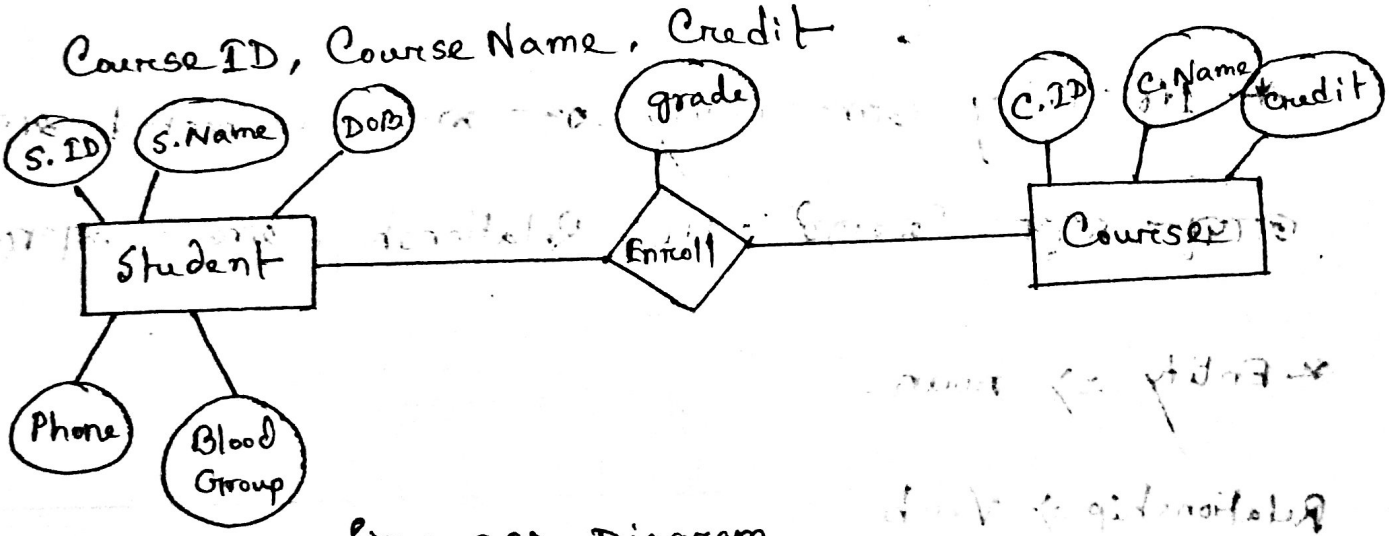


fig : ERD Diagram

S. ID	S. Name	DOB	Phone	Blood G.	C. ID	C. Name	Credit	Grade
1	A	"	"	"	CSE 3109	DB	3.0	A+
2	A	"	017	"	CSE 3109	DB	3.0	A-

08 May 2017

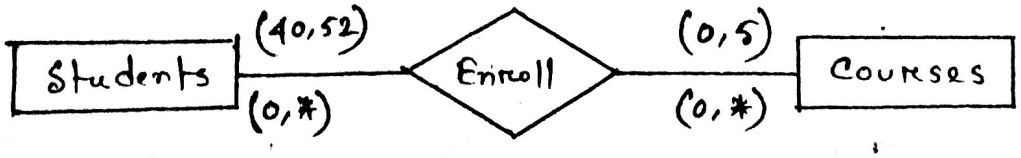
Cardinality Constraint of Relationship

Cardinality Ratio

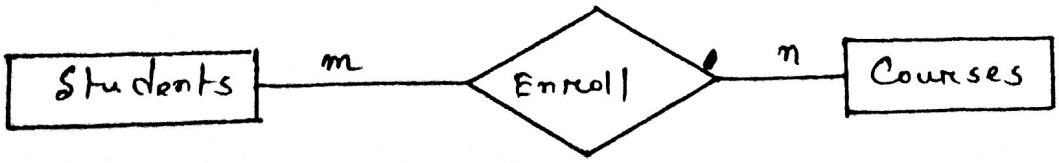
- Express the maximum number of Entity that can be associated with another Entity via a relationship.

4 types of Cardinality Constraints.

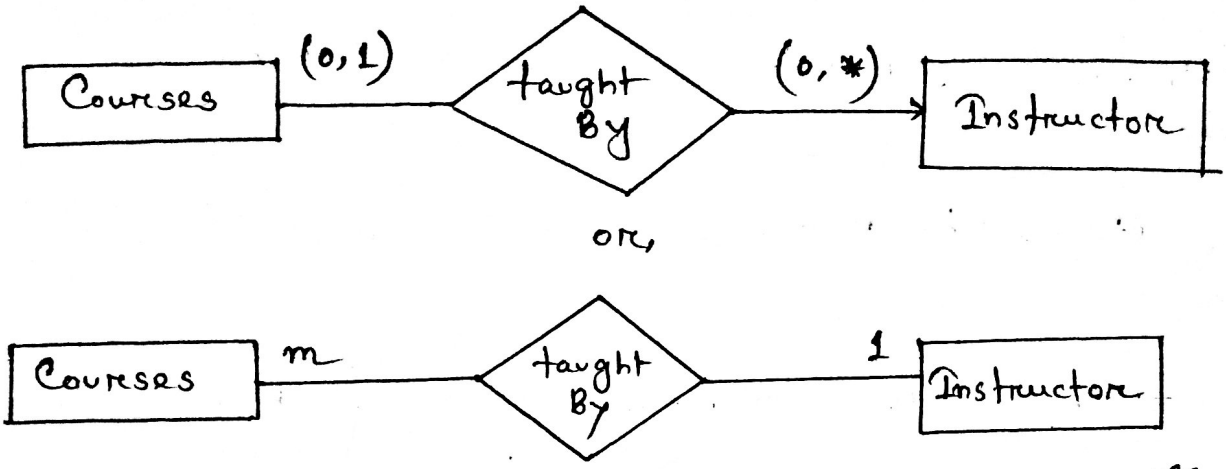
1. Many - To - Many



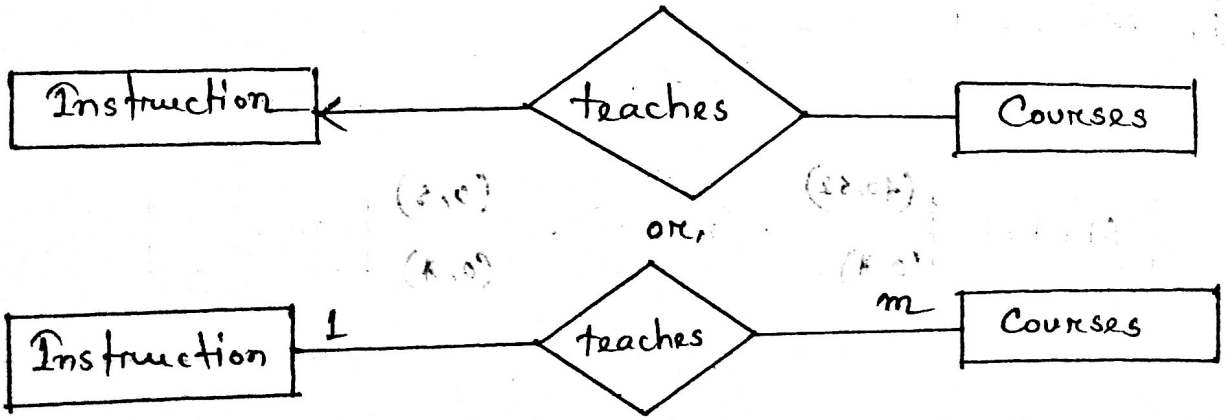
or,



2. Many - to - One

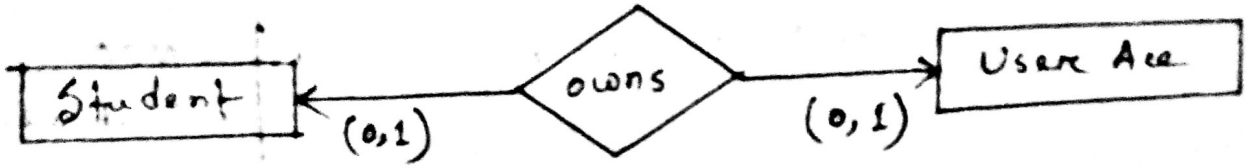


3. One - To - many

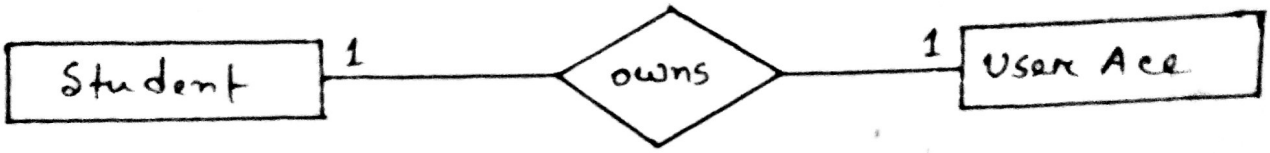


* Arrow বাদ representation avoid করার চেষ্টা করব।
* 2nd Representation টি follow করব।
* বই এর example গুলো practise করব।

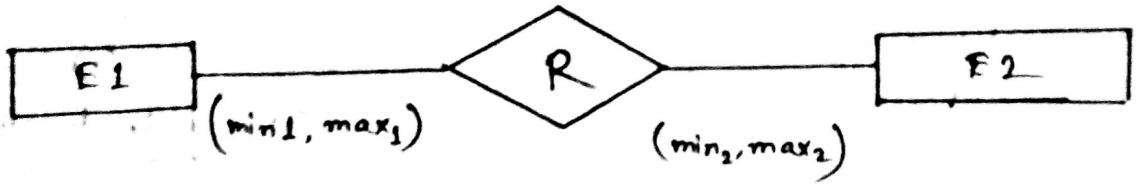
4. One-to-One



or,



Cardinality Limit / Degree of Constraints



Relationship	(min_1, max_1)	(min_2, max_2)
Many-to-Many	(0, *)	(0, *)
Many-to-One	(0, 1)	(0, *)
One-To-Many	(0, *)	(0, 1)
One-To-One	(0, 1)	(0, 1)

Quiz - 01 [14.05.2017]

Syllabus → slide 1, 2, 3 [uploaded]

[slide is not good to get 20/20]

→ MCQ
→ Short Question
→ 3 or 2 Marks এর Ques থাকতে পারে

এই এই Chapter - 1, 2

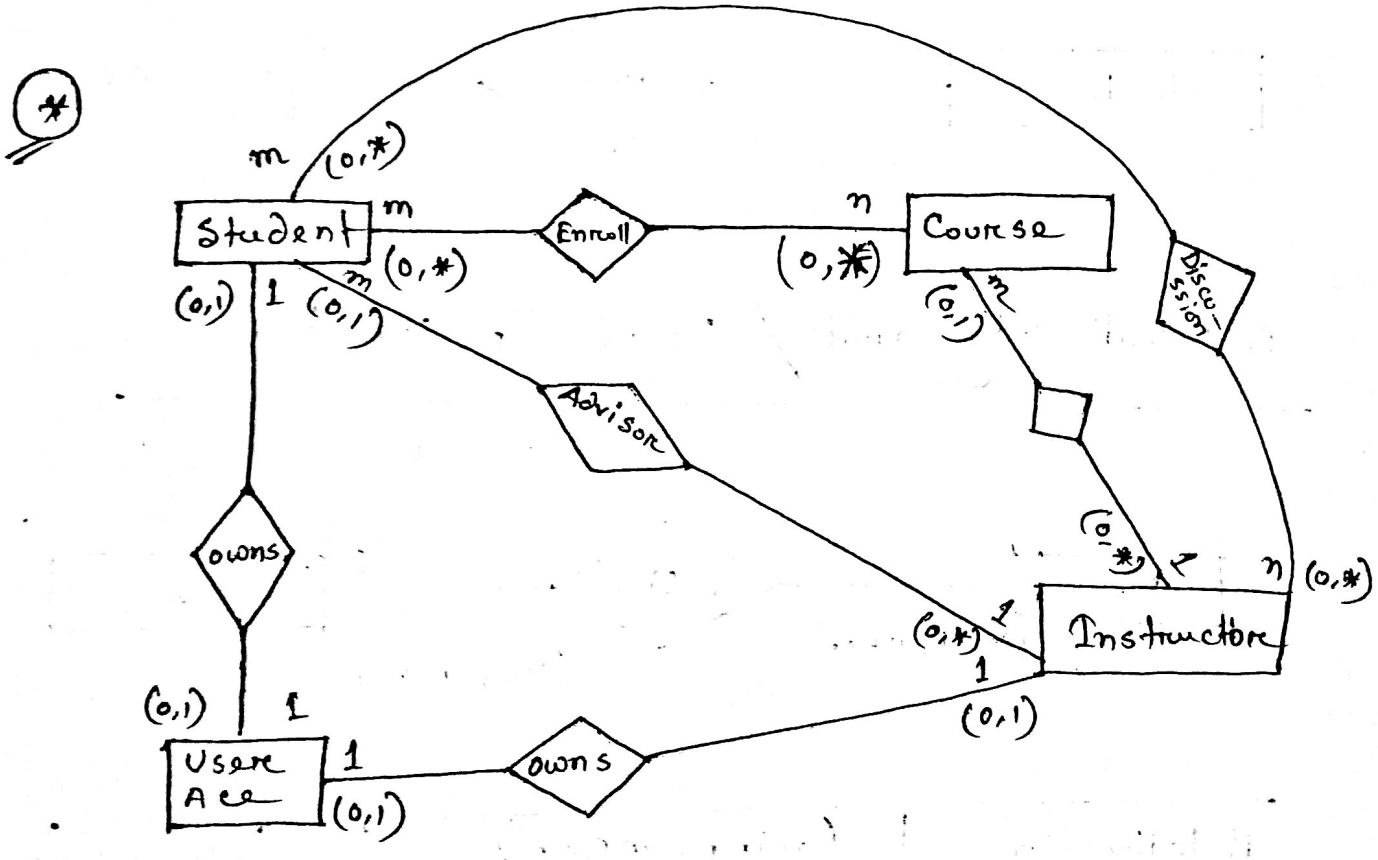


fig: Generic Representation

'*' দ্বারা কোনো Numeric Value বুঝায় না।

11 May 2017

Example 01: Student take courses and each student belongs to a particular Department. Students' grade in different courses are stored. Each department has multiple students and a Department offers multiple courses. A course can be offered by a single Department or multiple departments.

Entity:

Student, course, department

Relationship:

take, belongs, offers

Attributes:

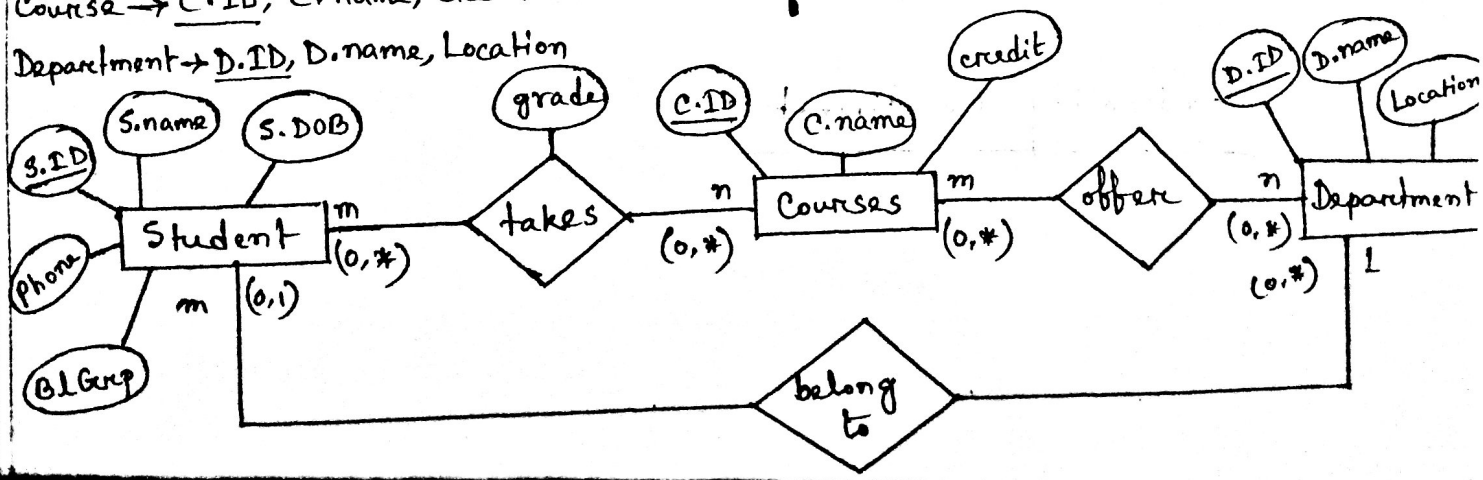
Student → S.ID, S.Name, S.DOB, Phone, Blgrp.

Course → C.ID, C.name, credit

Department → D.ID, D.name, Location

Cardinality Constraints:

ratio



Example-02: Patients are treated in a single ward by the doctors assigned to them. Usually each patient will be assigned by a single doctor but in rare cases they will have two. Assistants also attend to the patients and a number of these assistants are associated with each patient.

Entity: Patient, ward, doctor, Assistant

Relationship: treated, assign to, attend

Attributes:

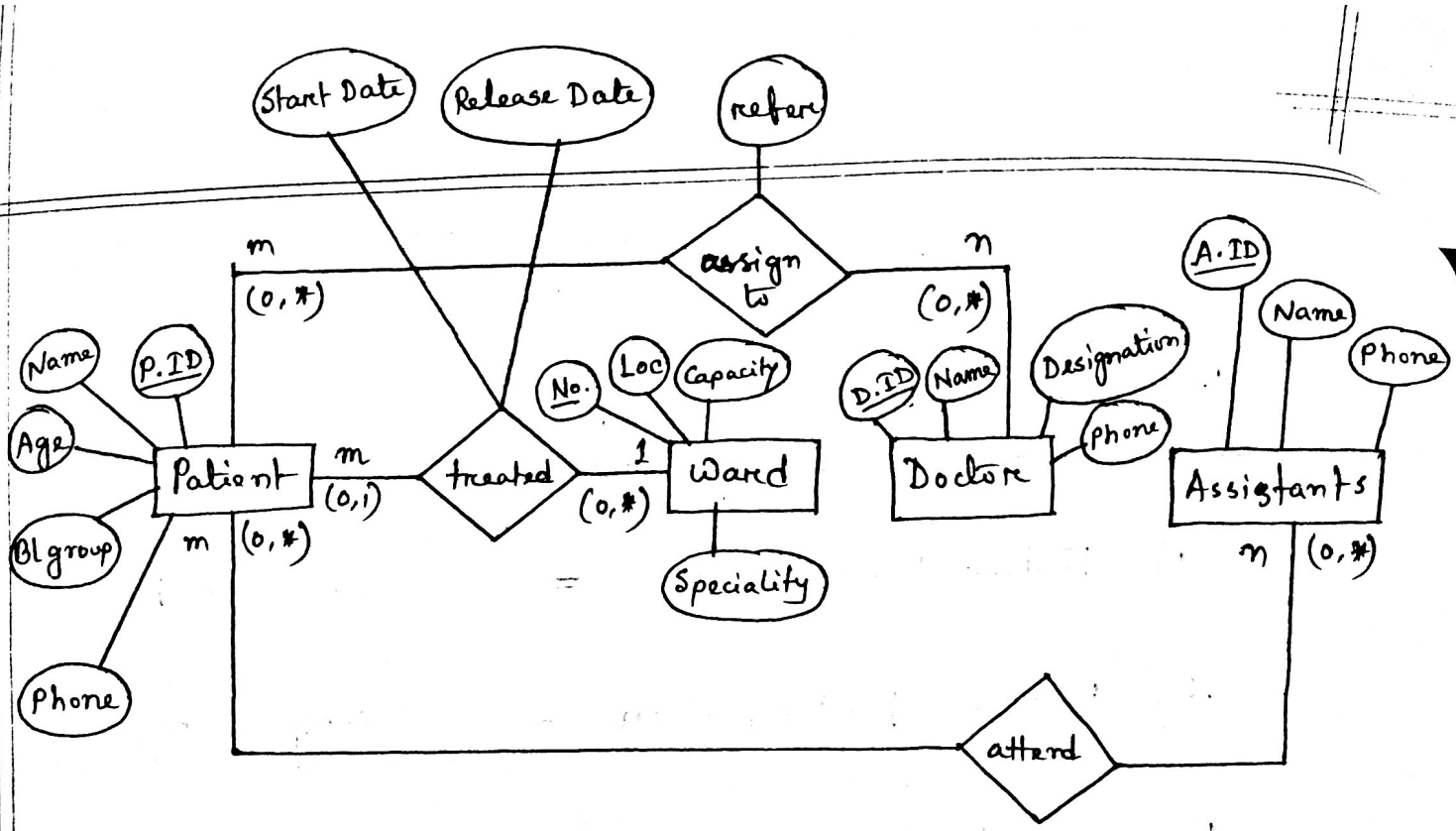
Patients → P. ID, Name, Age, Bl group, Phone

Ward → No., LOC, Capacity, Specialty

Doctor → D. ID, Name, Designation, Phone

Assistant → A. ID, Name, Phone

Cardinality Constraints:



* দুটি entity এর মধ্যে multiple relationship থাকতে পারে।

* সবসময়ই noun গুলোকে sentence থেকে বের করবে (Underline করে)।

* exam এ "5/6" টি Attribute তুলে ধরলেই হবে।

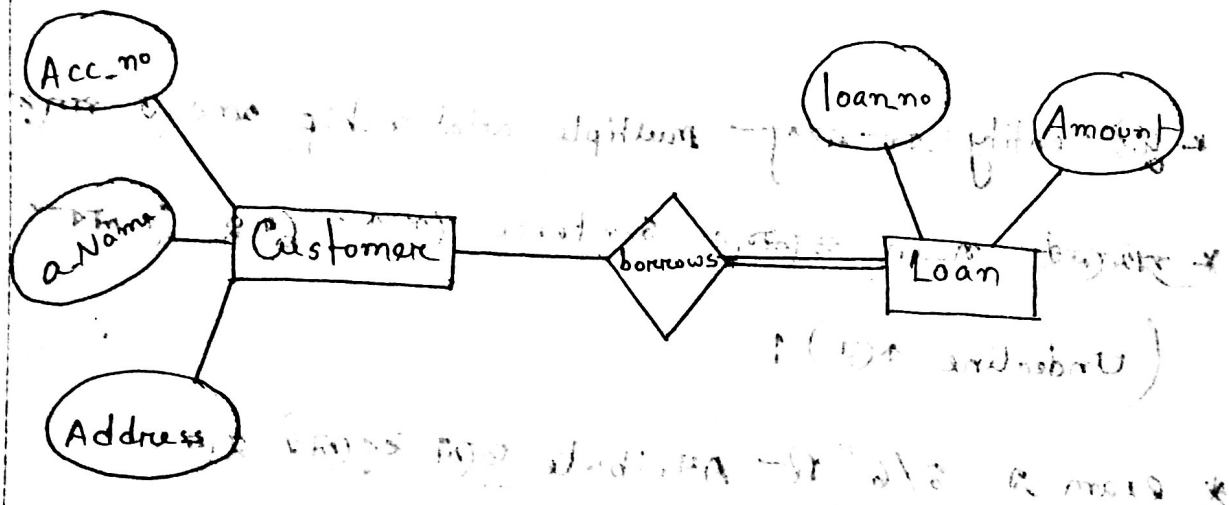
15 May 2017

Participation Constraints

1) Total Participation ═══ (double line)

2) Partial Participation ——— (single line)

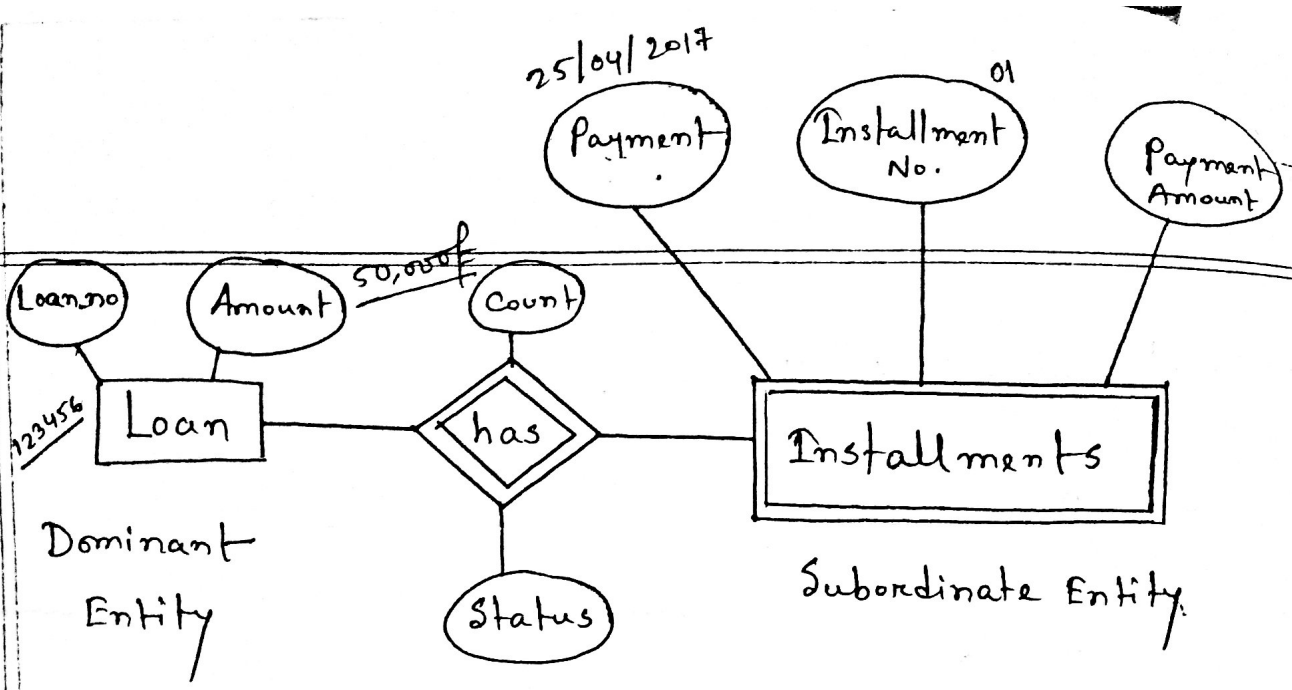
Example: Bank



Weak Entity Set: One-to-many relationship

Dominant Subordinate

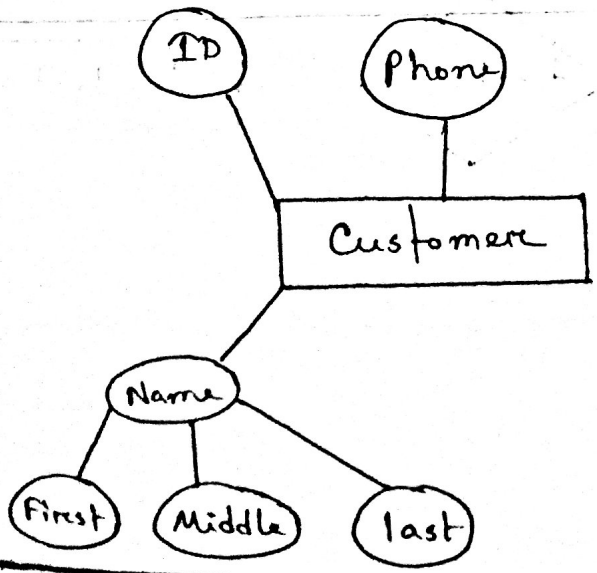
— Weak, entity, indentifying entity, discriminator, indentify relationship.



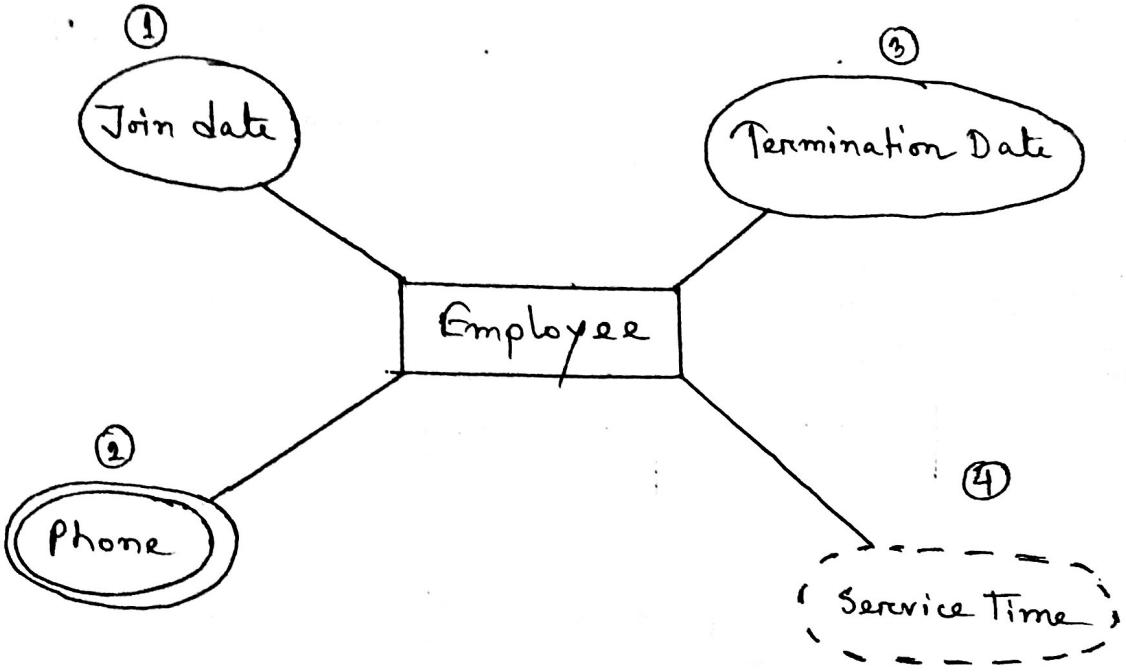
loan	Amount	Account
123456	50,000	ABC1200
123789	60,000	XYZ-987

Attribute Type

Simple & Composite



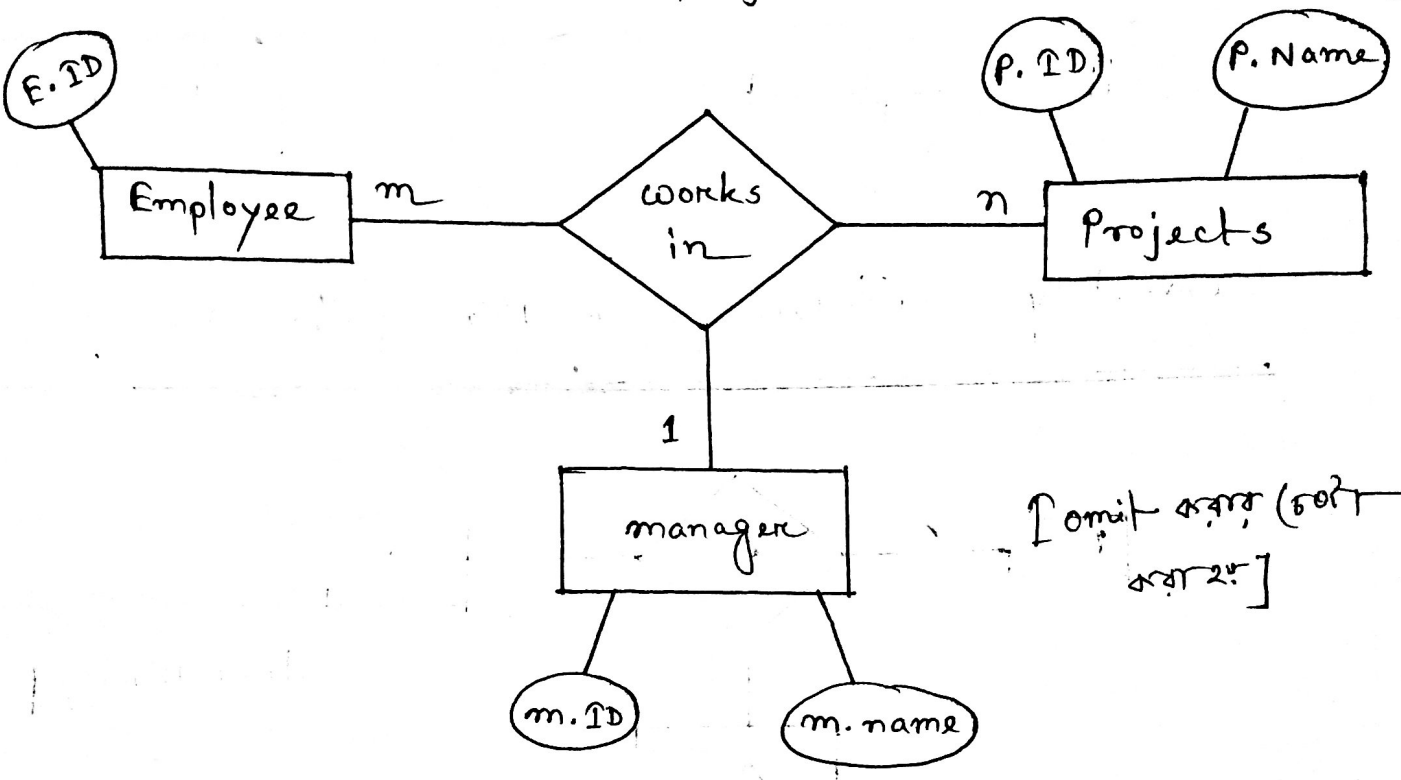
① Single Value, ② Multi Value, ③ Null and ④ derived



Degree of Relationship

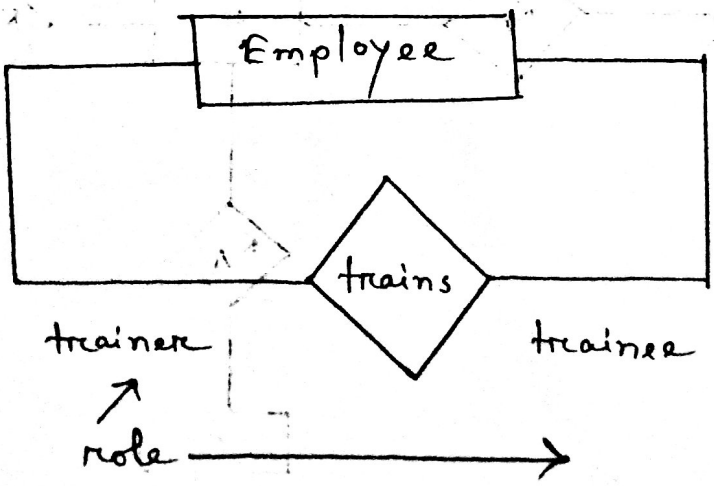
Binary

Ternary: A company has several projects, employees can work on several projects under one manager.



[omit করার (তবে করা যায়]

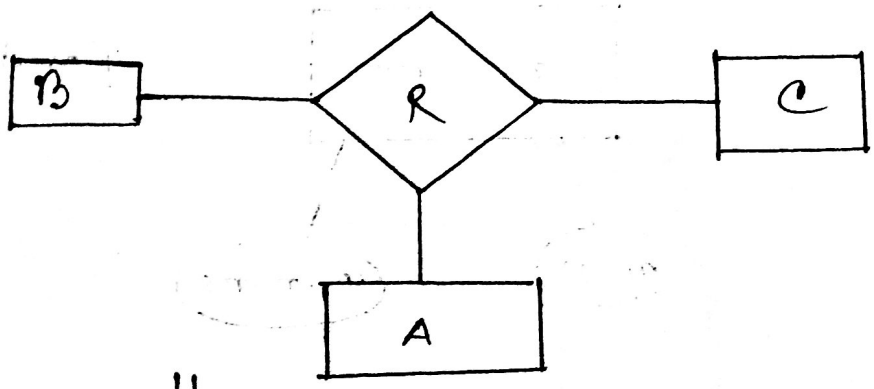
Unary / Recursive



N-ary : More than 3.

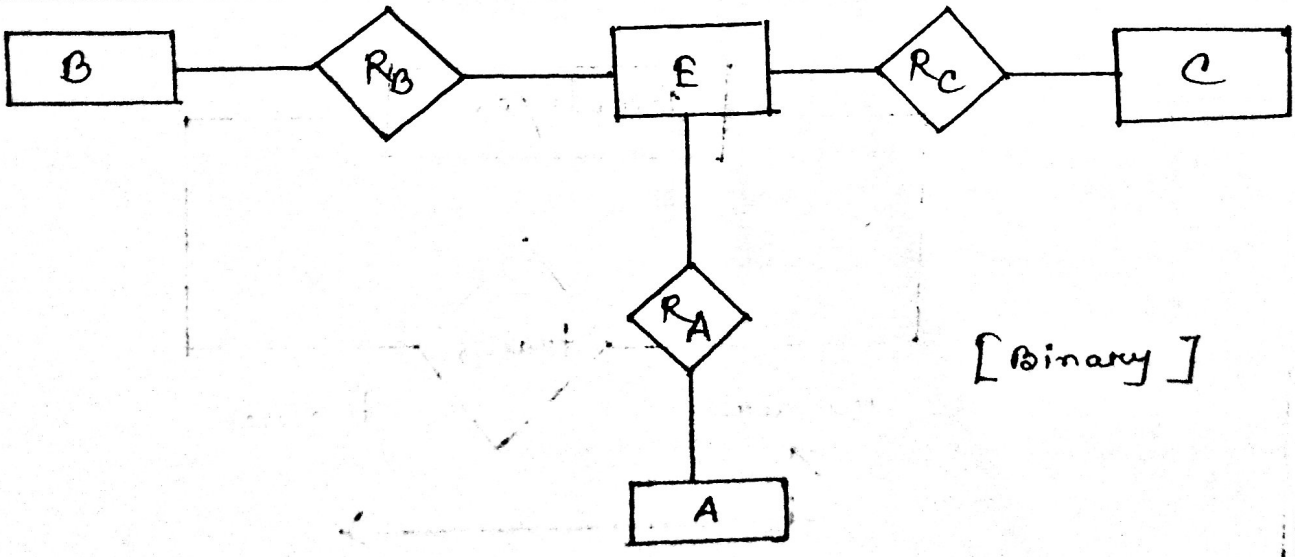
⇒ Try to avoid Unary and Ternary relationships as their implementation is not easy all time.

Converting non-Binary to Binary Form



[non-Binary]

⇓ conversion



[Binary]

Explanation

Simple

Composite

Joined date → single value

Teremination date → Null value

Phone → Multi value

Service Time → Derives [Calculation করা যায়]

Quiz-02 [04 JUNE 2017] (Sunday)

Syllabus → ERD

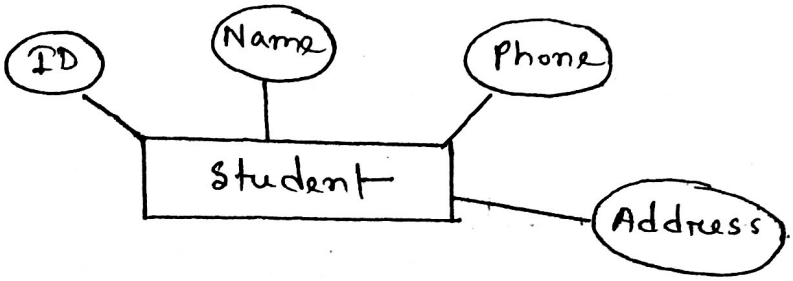
18 May 2017

Keys :

Suppose, (A) = {1, 2, 3, 4} (A → Entity set, 1, 2, 3, 4 → Attributes)

$2^4 = 16$ (2^n) P(A) = ? {1} {2} {3} {4} {1, 2}, ...

1) Super Keys : (যেকোন Entity set এর সকল Attribute এর সিট সঠিক



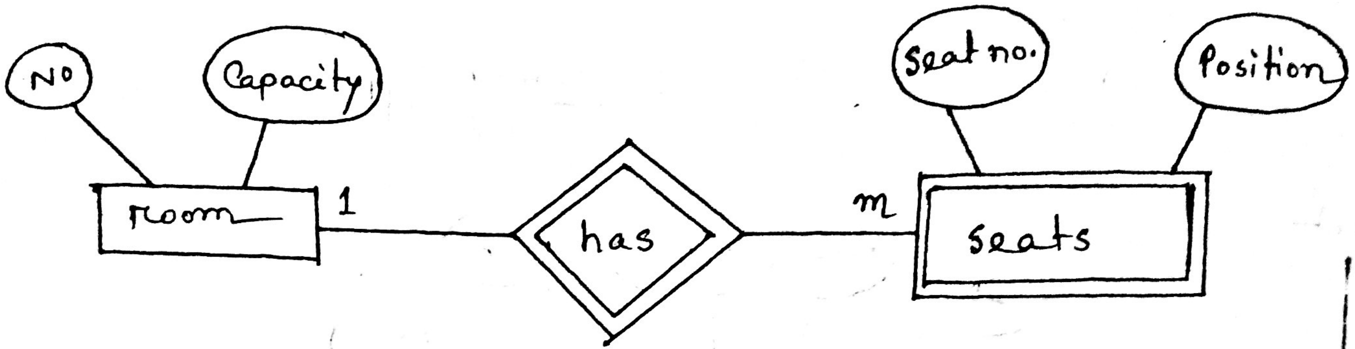
ii) Candidate Keys (যেসব super key, যার value minimal)
{ID}, {Phone}, {ID-Name}, {ID-Phone}

iii) Primary Key (যেকোন Candidate Key যার value Unique)

iv) Alternative Key (Primary Key select করার পর rest of the candidate keys are Alternative Keys)

v) Composite Key (একাধিক Attribute মিলে 1 টি Primary key declare করলে, তা Composite key)

vi) Foreign key (এমন একটি key যা অন্য table থেকে আসে তবে: সে table হতে আসা এবং Primary key)



* Foreign key is mostly used in weak Entity set.

Enhanced ER Modeling

→ In some cases, an Entity type has a number of subgroups.

→ Relationships and attributes of superclass are inherited to sub-class.

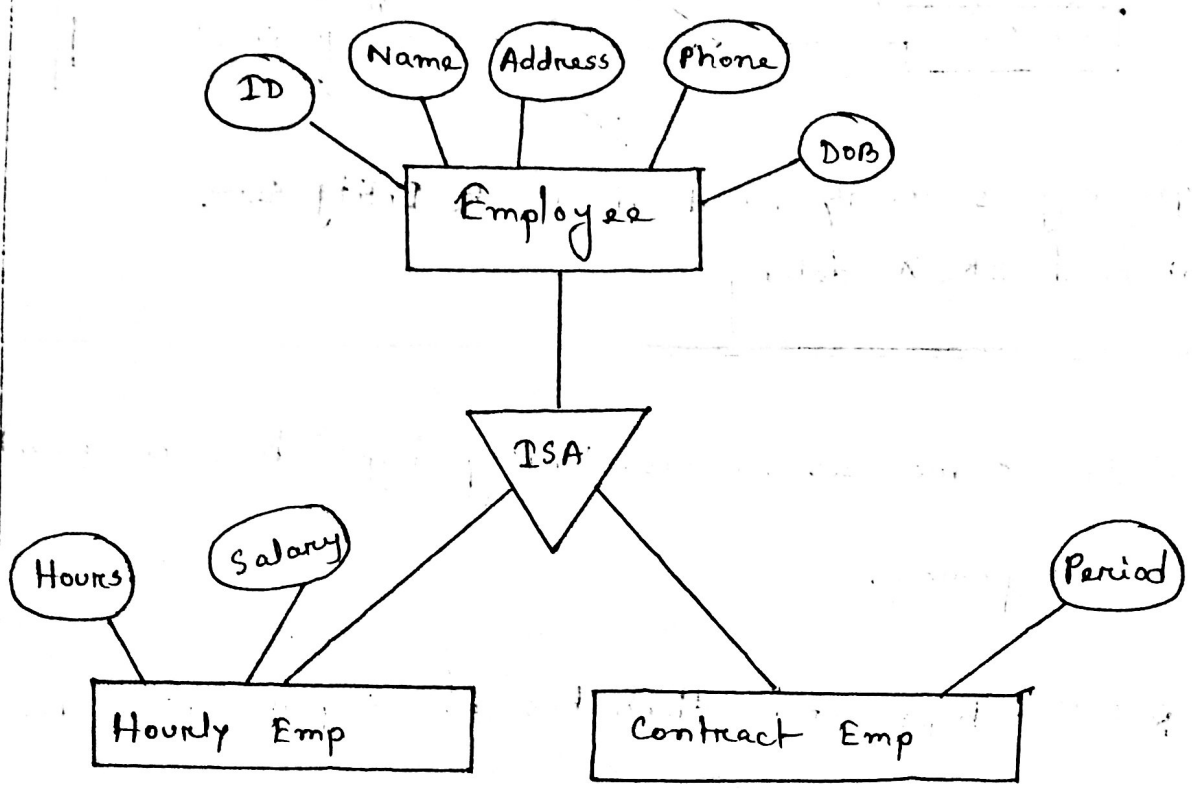
→ Sub class can have additional attributes and relationship.

Two types : i) Specialization

ii) Generalization

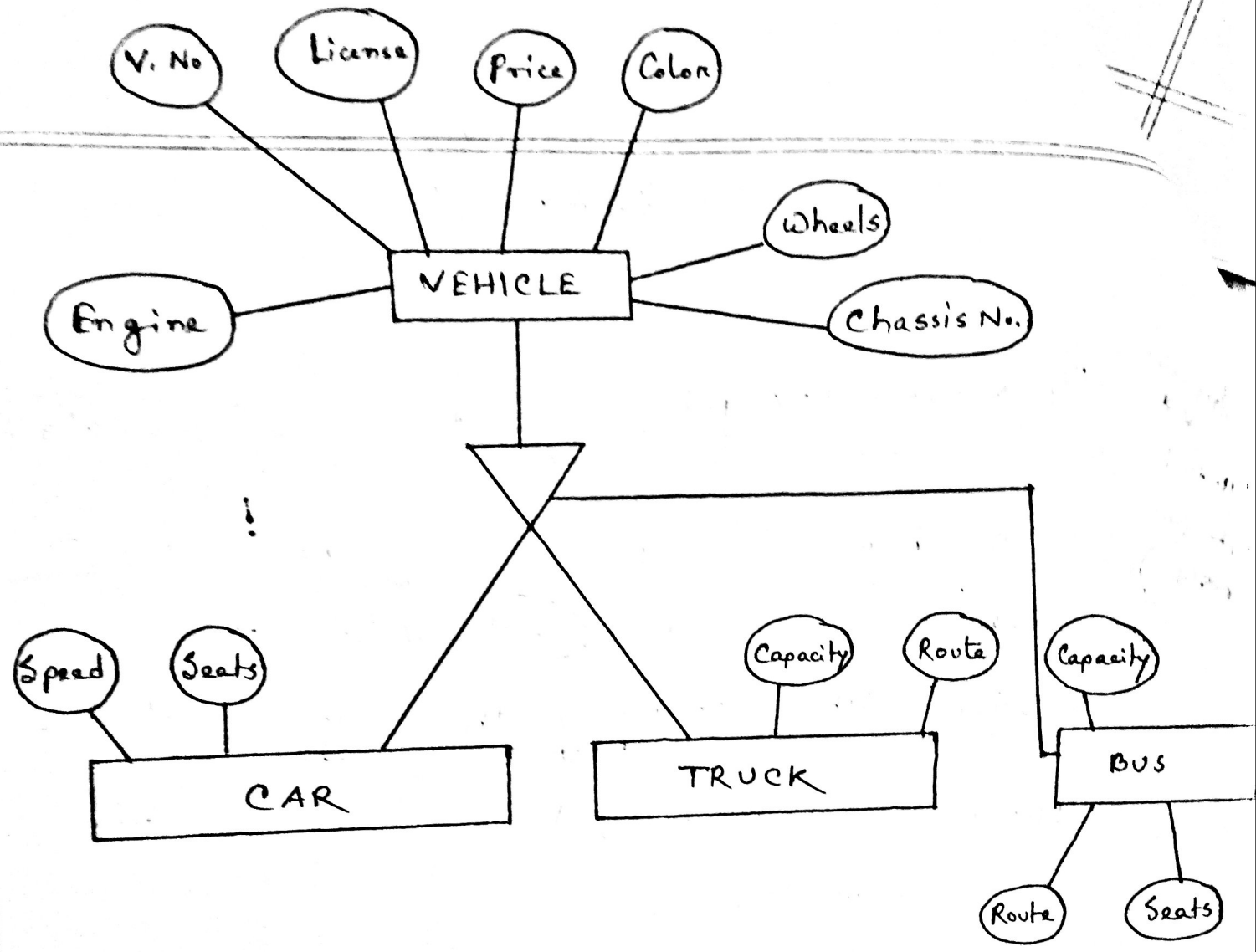
Specialization : Process of defining a set of sub classes of an Entity type (top down ↓)

Parent



ild

Generalization : Reverse process of Specialization (↑ bottom up). Identify common features of Entity types and generalize them into a single super class (Including Primary Key)


V. No
License
Price
Color
Wheels
VEHICLE
Engine
Chassis No.
Speed
Seats
Capacity
Route
Capacity
CAR
TRUCK
BUS
Route
Seats

21 May 2017

Important ** (Full at Ques. rate)

Relational Algebra:

The relational algebra is a theoretical language with operations that work on one or more relations (table) to define another relation without changing the original relation. It is used to manipulate relations to obtain a desired result.

Relational Algebra Operations:

Union, intersection, difference, division, renaming Selection, projection, cartesian product, join etc.

* **Unary Operation:** Operate in one relation (1 table).

Example: Selection, projection etc.

* **Binary Operation:** Operate on two relations.

Example: Union, Intersection, join etc.

* N-ary Operation:

(দুই বা তার অধিক table নিয়ে Operation)

Selection Operation: (Condition নিয়ে কাজ করে)

(Specific কোন কোন Row কে select করাই)

Syntax:

$$\sigma_{predicate}(R)$$

here,

σ = Selection (sigma)

predicate = Condition

R = relation

Symbols:

$<$ $>$ or $\neq$

$>$, $<$, $\geq$, $\leq$, $=$

$\vee$, $\wedge$, $\neg$

↓ ↓ ↓

OR AND NOT

Example:

Student

Roll	Name	GPA
01	Rupa	4.5
02	Rimi	3.5
03	Shuvo	2.5
04	Sami	3.2
05	Poly	2.7
06	Kuva	2.9

Find student whose GPA is > 3.00

$\sigma_{GPA > 3.0}$ (Student) ✓ (Just এই line টা Ans করার জন্য)

ROLL	Name	GPA
01	Rupa	4.5
02	Rimi	3.5
04	Sami	3.2

[এমন Table আঁকা লাগবে না]

Employee

Name	Age	Salary
Marly	25	9000
Luky	40	3000
Mark	36	4500
John	42	3900

1. Find the Employee whose age is less than 30 years.

   ⇒ $\sigma_{Age<30}$ (Employee)

2. Find the Employee whose salary is more than 4000

   ⇒ $\sigma_{salary > 4000}$ (Employee)

3. Find the Employee whose age is less than 30 and get 4000 salary?

   ⇒ $\sigma_{Age<30 \wedge salary = 4000}$ (Employee)

4. Find the Employee whose get 4000 or age below 30?

   ⇒ $\sigma_{salary = 4000 \vee age<30}$ (Employee)

5. Find the Employee whose name is 'Luky'.

=> $\sigma_{name = 'Luky'}(Employee)$

Projection Operation:

Syntax:

$$\Pi_{a_1, a_2, \ldots, a_n}(R)$$

Here,

Π = projection (pie)

$a_1, \ldots, a_n$ = attributes

staff

Name	Gender	Date of Birth	Salary
Rony	M	01/05/86	20,000
Tony	M	05/07/80	30,000
Jony	M	07/02/95	40,000
Rupa	F	03/09/96	35,000

* Show the names of staff Relation.

=> $\Pi_{name}(staff)$

Name
Rony
Tony
Jony
Rupa

* Show the names and the salary of staff Relation.

=> $\Pi_{name,\ salary}(staff)$

* Show the names of the staff whose salary is greater than 30,000 in the relation staff.

=> $\Pi_{name}\left(\sigma_{salary > 30,000}(staff)\right)$

* Projection operation is used to show the columns of a relation.

```
Select Name
From staff
where .....
```

22 May 2017

Task

Customer

Name	Street	City	Phone	Age
Ruhi	abc	DHAKA	123	12
Adnan	def	KHULNA	456	16
Upoma	ghi	RAJSHAHI	789	17
Sazzad	jkl	DHAKA	135	19

i) Find the customers name who lives in Dhaka.

Ans. $\Pi_{name}(\sigma_{city='dhaka'}(customer))$

ii) Find the customer name whose age is more than 15 and lives Dhaka where his/her phone numbers starts with 1.

Ans. $\Pi_{name}(\sigma_{age>15 \wedge city='dhaka' \wedge phone='1\%'}(customer))$

☐ Binary Operation Compatibility : To perform union, intersection, difference operations, relations should be UNION compatible. 2 relations are UNION compatible if they have some number of attributes and belong to the same domain.

→ Column number need to be same.

→ Domain Type need to be same.

R

A	B

S

A	B

Compatible

R

A	B

S

C	D

Not Compatible

Suppose,

R

A	B
α	1
α	2
β	1

S

A	B
α	2
β	3

Union Operation :

$R \cup S$ is

A	B
α	1
α	2
β	1
β	3

Intersection Operation :

$R \cap S$ is

A	B
α	2

Difference Operation :

$R - S$ is

A	B
α	1
β	1

$S - R$ is

A	B
β	3

⊞ Rename Operation :- [table ক্ষেত্রে এই instance create করে। Attribute " " Attribute Name change করে]

→ When tables are not compatible

→ When need to compare same relation

To rename a table. $\rho_A(B)$ → old table-name
↓ new table-name

To rename attribute

$$\rho_{\text{old name} \rightarrow \text{new name}}(R)$$

Example:

Employee (name, branch, salary)

after,

$$\rho_{\text{branch, salary} \rightarrow \text{location, pay}}(\text{Employee})$$

So, we will get,

Employee (name, location, pay)

Division Operation :

$$\frac{R}{S} = \frac{AB\not{\alpha}}{\not{\alpha}}$$

= AB Column

R

A	B
a_4	b_1
a_1	b_2
a_2	b_1
a_3	b_2

ভাগ্য

S

B
b_1
b_2

ভাজক

R/S =

A
a_1

ভাগফল

$$\frac{R}{S} = \frac{A\not{B}}{\not{B}} = A \text{ Col}$$

* এখন A হল Ans Col.
* Answer হল ভাজকের কোন কোন element গুলো ভাগ্য এর সাথে shared আছে তা।
* Common না থেকে Ans 0 হবে (এই Ans Col এর Attribute রাখা যাবেনা!

(*)

A

Sno	Pno
S_1	P_1
S_1	P_2
S_1	P_3
S_1	P_4
S_2	P_1
S_2	P_2
S_3	P_2
S_4	P_2
S_4	P_4

B_1

Pno
P_2

B_2

Pno
P_4

A/B_1 =

Sno
S_1
S_2
S_3
S_4

A/B_2 =

Sno.
S_1
S_4

Cartesian Product Operation:

R

A
a
b

S

B
1
2
3

R x S =

A	B
a	1
a	2
a	3
b	1
b	2
b	3

*

$[2 \times 2 = 4]$

R

A	B
α	1
α	2

S

B	C
1	x
2	4

R x S =

A	B	D	C
α	1	1	x
α	1	2	4
α	2	1	x
α	2	2	4

$\rho_{B \to D}(S)$

* কোনটা কোন table থেকে আসছে তা বোঝার জন্য rename করতে হবে (in case of common name)

25 May 2017

Example:

Borrower

Name	Loan no
KAMAL	L-17
JAMAL	L-23

Loan

Loan no	Branch	AMOUNT
L-14	KAKRAIL	75,000
L-23	MOTIJHEEL	50,000

Relational Algebra

Find the name of the Customers of a Bank (in-example) who have a loan at MOTIJHEEL Branch.

$\rho_{Loan.no \rightarrow borrow.Loan}(borrower)$

$\rho_{Loan.no \rightarrow Loan.Loan\,no}(Loan)$

(Borrower × Loan) = new_table → optional
(না করলেও চলে)

Name	Borrow.Loan no	Loan.Loan no	Branch	Amount
KAMAL	L-17	L-14	KAKRAIL	75,000
KAMAL	L-17	L-23	MOTIJHEEL	50,000
JAMAL	L-23	L-14	KAKRAIL	75,000
JAMAL	L-23	L-23	MOTIJHEEL	50,000

Borrow.Loan no = Loan.Loan no

Branch = MOTIJHEEL

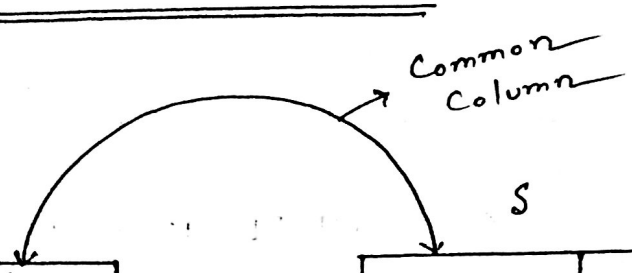
$$\Pi_{name}\left(\sigma_{borrow.Loan\ no\ =\ Loan.loan\ no\ \wedge\ Branch = MOTIJHEEL}\left(Borrow \times Loan\right)\right)$$

* table গুলো গুণ করলে => "Cartesian"
table যুক্ত করলে => "Join"
* Cartesian এ row গুলো সবগুলো row গুলোর সাথে match করবে।

28 May 2017

Natural Join (⋈)

Example:



R

Name	Department
Smith	CSE
John	EEE
Paul	EEE

S

Department	Head
EEE	Brown
CSE	Alen
MCE	White

Joining এর সময় কোনো একটি table এর specific কোনো row বাদ গেলে, তাকে "Dangling Tuple" বলে।

R ⋈ S

Bridge ⟷

Name	Department	Head
Smith	CSE	Alen
John	EEE	Brown
Paul	EEE	Brown

Use the tables of the last class:

Borrower (Name, Loan no)

Loan (Loan No, Branch, Amount)

Find the name of the customer who have a loan at MOTIJHEEL Branch using natural Join.

⇒

Name	Loan No	Branch	Amount
JAMAL	L-23	MOTIJHEEL	75,000

$$\Pi_{name}\left(\sigma_{Branch = MOTIJHEEL}(Borrow \bowtie Loan)\right)$$

Theta Join:

Notation $(R \bowtie S)$

or,

$$\sigma_{condition}(R \bowtie S)$$

Example:

Depositor (CustomerNo, name, accountno)

Account (accountno, branch, balance)

— Find the names of the customer who have an account in the bank having balance greater than 5000?

$$\Pi_{name}(\sigma_{Balance > 5000}(Depositor \bowtie Account))$$

or,

$$\Pi_{name}(Depositor \bowtie Account)_{Balance > 5000}$$

Semi Join

(Left side Priority বেশি, resulting table এ 3 left side এর column থাকবে right side এর column থাকবে না)

Half

Notation $R \ltimes_F S$, F is the condition

Say,

R

E.No	E. name	Designation
E_1	Rupa	Programmer
E_2	Nipa	Analyst
E_3	Jhuma	DB Admin
E_4	Sadat	Consultant

S

Designation	Salary
Programmer	25000
Consultant	50,000
Manager	75000

$R \bowtie S$

R designation = S designation

E.No.	E. Name	Designation
E_1	Rupa	Programmer
E_4	Sadat	Consultant

Outer Join

(Left এলাম table full আসবে, Right এর value match আসলে হবে, otherwise NULL হবে)

1) Left outer Join (⟕)
2) Right Outer Join (⟖)
3) Full Outer Join (⟗)

Example

R1

E. Name	Department
Smith	Sales
Black	Production
white	Production

R2

Department	Head
Production	Mary
Purchasing	Brown

1) Left Outer Join

$R_1 \text{ ⟕ } R_2$

E. Name	Department	Head
Smith	Sales	NULL
Black	Production	Mary
white	Production	Mary

2) Right Outer Join

$R_1 \text{ ⟖ } R_2$

E. Name	Department	Head
Black	Production	Mary
NULL	Purchasing	Brown
White	Production	Mary

29 May 2017

3) Full Outer Join — (Left Outer + Right Outer)

Name	Department	Head
Smith	Sales	NULL
Black	Production	Mary
White	Production	Mary
NULL	Purchase	Brown

Aggregation : Aggregate function take a collection of values and return a single value as a result.

(একাধিক value নিয়ে একটি value return করে)

Column/Attribute Value (same Domain)

Notation

$$_{G}\mathcal{G}_{F(A)}(R)$$

$\mathcal{G}$ → Aggregate Symbol
F → Functions
A → Attribute

R → Relation
G → group By

Functions can be

- max
- min
- sum
- count
- count_distinct
- average

etc.

Example

Instruction

ID	Name	Department	Salary
01	Alen	CSE	20,000
21	Brown	EEE	25,000
51	Cook	EEE	35,000
22	Dawson	ME	10,000
43	Enly	CE	15,000
56	Frank	CSE	40,000
65	Gibson	ME	30,000

- Find the total salary of the Instructor?

⇒ $\mathcal{G}_{SUM(salary)}(Instruction)$

- How many instruction are there in Instruction Relation?

⇒ $\mathcal{G}_{count\text{-}distinct(ID)}(Instructor)$

- How many departments are there in Instruction Relation?

⇒ $\mathcal{G}_{count\text{-}distinct(department)}(Instructor)$

- Find the average salary of the instructors according to the department?

⇒ $\mathcal{G}_{average(salary)}(Instructor)$

Department → Group By (as footer-back)

ID	Name	Department	Salary
43	Enly	CE	15,000
01	Alen	CSE	20,000
56	Frank	CSE	40,000
21	Brown	EEE	25,000
51	Cook	EEE	35,000
22	Dawson	ME	10,000
65	Gibson	ME	30,000

(Group) ⟶

Department	Average Salary
CE	15,000
CSE	30,000
EEE	30,000
ME	20,000

01 June 2017

CHAPTER: NORMALIZATION

Functional Dependency: Attribute of a relation depends on other attributes of the same relation.

- determinant
- objects of determinant } not vice versa.

Example:

Student ID → {Firstname, lastname}

Course number → Course name, teacher name

Student ID, Course number → marks

marks → grade

Course number → Teacher name

Course Number, Course Name → Teacher name

Composite determinant (কমপক্ষে Determinant)

☐ Partial Functional Dependency: যদি an attribute, is removed from the Composite determinant but the dependency is not impacted.

course name, course number → Teacher name

* কমপক্ষে Attribute বাদ দিলে Database "Impacted" হয়।

⊞ Full Functional Dependency: If an attribute is removed from the composite determinant and the dependency is impacted.

Student ID, Course Number ⟶ Marks

⊞ Transitive Dependency: When attributes are creating a chain of dependency.

$R(A, B, C)$

$A \rightarrow B$, $B \rightarrow C$

$\therefore A \rightarrow C$

Staff

$\overset{A}{\text{Staff no}} \rightarrow \overset{B}{\text{designation}} \rightarrow \overset{C}{\text{Salary}}$

$\overset{A}{\text{Staff no}} \rightarrow \overset{C}{\text{salary}}$

⊞ Normal Form (data inconsistent)

1. 1NF (First Normal Form)
2. 2NF
3. 3NF
4. BCNF (Boyce Codd)
5. 4NF
6. 5NF
7. 6NF

Normalization

Normalization is needed to stop data inconsistency. For 3 reasons problem is created.

- Insert anomaly
- Delete anomaly
- Update anomaly

Person ID	Project Budget	Project	Time Spent by person for project
S75	60 (32)	P1	7
S75	40	P2	3
S79	32	P1	4
S79	27	P3	1
S96	40	P2	5
→ S96	50	P1	7

Steps in NORMALIZATION

1) Specify the primary key of the relation.

2) Specify the Functional Dependency of a relation.

3) Apply the definition of each Normal Form (Starting with INF)

4) If a relation fails to meet the definition of Normal Form, change relation until it meets the definition.

5) Re-Test the modified/new relation to ensure they meet the defination of each Normal Form.

05 June 2017

First Normal Form (1NF)

A relation (→ Table) is in 1NF if it is satisfied the following conditions.

→ Contains only atomic (→ single) values

→ There are no repeating groups

Product

ProductID	Colour	Price
1	red, green	15.99
2	yellow	23.95
3	green	17.50
4	blue, green	9.99
5	red	29.99

Book

Book.id	Author 1	Author 2	Author 3
1	Abraham	John	Paul
2	Candlon	Smith	Jack
3	Bella	Naomi	John

* Atomic value নয়, so 1NF ও আনতে হবে।

then "Duplicacy" check করতে হবে।

* table গুলার name- table-name change হয়ে যাবে।

determinant → object of determinant

Product ID → Colour
Product ID → price
} break করতে হবে

Product-colour

ProductID	Colour
1	red
1	green
2	yellow
3	green
4	blue
4	green
5	red

Product-Price

ProductID	Price
1	15.99
2	23.95
3	17.50
4	9.99
5	29.99

Book-Author

BookID	Author
1	Abraham
1	John
1	Paul
2	Candlon
2	Smith
2	Jack
3	Jack
3	Jack
3	Jack

Second Normal Form (2NF)

A relation is in 2NF if it satisfies the following conditions -

→ It is in 1NF

→ All non-key attributes are fully functionally dependent on primary key.

*"Normalization" mainly "Data-Inconsistency" দূর করে।

Purchase Detail

Customer. ID	Store ID	Location
1	1	DHAKA
1	3	KHULNA
2	1	DHAKA
3	2	SYLHET
4	3	KHULNA

* Multiple Value Database ↳ single Row ↳ insert না করাই ভালো, Because ↳ ↳ NORMALIZATION করা যায় না।

Determinant — Object of Determinant

CustomerID, StoreID → Location

CustomerID → Location ✗

StoreID → Location ✓

CustomerID → StoreID ✓

Purchase

Customer ID	Store ID
1	1
1	3
2	1
3	2
4	3

Store

Store ID	Location
1	DHAKA
2	Sylhet
3	KHULNA

* এখানে primary key গঠনকারী attribute ব্যবহার করা যায়—
(Duplicacy Possible)

06 July 2017

Third Normal Form (3NF) : A relation

is in 3NF if it follows:

- It is in 2NF.
- No transitive dependency exists within the relation.

Transitive

$A \rightarrow B$

$B \rightarrow C$

$\therefore A \rightarrow C$

Book_details

Book_id	Genre_id	Genre Type	Price
1	1	Story	120
2	2	Sports	140
3	1	Story	130
4	3	Travel	350
5	2	Sports	250

Functional Dependency

Book_id (A) → Genre_id, price (B)

Genre_id (B) → GenreType (C)

∴ A → C

Book → GenreType

Transitive হচ্ছে। So, 3NF হচ্ছে না। অতএব, Convert করতে হবে।

Book_price

Book_id	Price	Genre_id
1	120	1
2	140	2
3	130	1
4	350	3
5	250	2

Genre_details

Genre_id	Genre_type
1	Story
2	Sports
3	Travel

Not transitive. So, 3NF.

Employee

Emp_id	Emp_name	Emp_zip	Emp_state	Emp_city	Emp_district
1001	John	201212	NR	RAJ	RANGPUR
2001	Lora	201208	SU	DHK	MYMENSING
2019	Steve	192001	SU	DHK	MANIKGANJ
1012	Mark	201219	ES	KHL	SATKHIRA
1051	Smith	202002	WS	CTG	COX'SBAZAR

Q. Normalisation কেন দরকার?

Q. Normalisation এর steps গুলো লিখ।

Q. Partial Functional Dependency & Fully Functional Dependency এর পার্থক্য লিখ।

Q. Composite Determinant কাকে বলে?

Q. Normalization এর কয়টা form আছে?

Q. কোন Normal এ আছে? কোনটায় নাই? কেন নাই? Normal কর। (7 Marks)

ঐ- Rule সাপেক্ষে যে Normal.

10 July 2017

Boyce - Codd Normal Form (BCNF):

A relation is in BCNF if and only if every determinant in the relation is a candidate key (where candidate keys are composite and overlapping)

⇒ Candidate-key ?? recall

⇒ determinant ?? recall

$R = \{C, B, D, F\}$

2^4

$A \rightarrow B, C$

Determinant — Object of Det.

$X \rightarrow Y$

Example:

Client Interview

client no.	interview date	interview time	staff no.	room no.
C76	10-7-17	10:30	SG7	G101
C54	10-7-17	11:20	SG7	G101
C74	10-7-17	12:30	SG9	G102
C54	10-7-17	10:30	SG7	G102

Primary key:

Client No, interview_date.

...

Candidate Key (Question এ দেয়া থাকবে)

- { staffno, interview_date, interview_time }
- { room_no, interview_date, interview_time }

Functional Dependency:

(Determinant) (Object of Det.)

✓ i) clientno, interview_date → interview_time, staffno, room_no

✓ 2) Staff no, interview_date, interview_time → clientno

✓ 3) room_no, interview_date, interview_time → Staff no, client no.

✗ 4) staff_no, interview_date → room_no

Not Candidate Key
Not in BCNF

* যদি সব Condition fullfil করবে → Yes / ✓

যদি সব Condition full fil করবে না → No / ✗

* যে কোন Rule মানে না ⇒ PRIMARY KEY
* যে কোন Rule মানে ⇒ CANDIDATE KEY

X 5) Staff_no, room_no → interview_time

Not in BCNF

* Question এর data চাহিদা না করে; extra কোন "functional dependency" create করে দিতে হবে।

So, new table is below:—

Staff_no	P.K interview_date	room_no

staff_no	interview_date	interview_time	client_no

(*) Employee

Emp_no	Emp_nationality	Emp_dept	dept_type	No. of Emp
1001	AUS	Production	D1	150
1001	AUS	Design	D1	45
1002	NZ	Sales	D4	55
1002	NZ	Marketing	D4	80

Primary Key

Emp_no

Candidate Key

- {Emp_no, Emp_dept}.

Functional Dependency

1) Emp_id → Emp_nationality
2) Emp_id → Emp_dept
3) Emp_dept → Dept_type, no_of_Emp
4) Emp_id, Emp_dept → Emp_nationality, Dept_type, no_of_Emp

* Extra Class

Wednesday (3:30 – 4:20)

7A04

12 July 2017 (Extra Class)

B^+ Tree : (final ৯ 9 Marks এসেছে)

Rules:

1. All data are stored in leaf node.
2. Every leaf node will be in same level.
3. All leave node have links to each other. (→) [পরবর্তী leaf এ যাবে]
4. There will be a threshold value which declare the size of leaf.
5. markup will stay in the uppere part of leaf
6. markup directed the path where the data leaf stored.
7. Left of a markup will be always < less than markup value.
8. Right of a markup will be always ≥ greater or equal of markup value.

⑨ For each of threshold value (n) we need to find out.

∴ maximum value contain in a leaf = $n-1$

[ceiling]

∴ minimum value contains in a leaf = $\frac{n-1}{2}$

[floore]

Example (a)

Q. Construct the following B^+ tree and perform the following operations (Insert, Delete, Threshold value)

[Asc] Insert: 1, 4, 7, 10, 17, 21, 28, 31, 42

[Dese] Delete: 42, 4

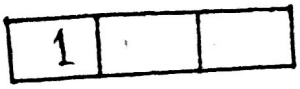
; সবসময় leftside থেকে Insert করব।

Threshold $n = 4$

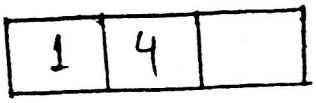
$$\max = n-1$$
$$= 4-1$$
$$= 3$$

$$\min = \frac{n-1}{2}$$
$$= \frac{3}{2}$$
$$= 1.5$$
$$= 1$$

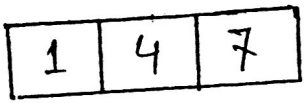
Insert : 1



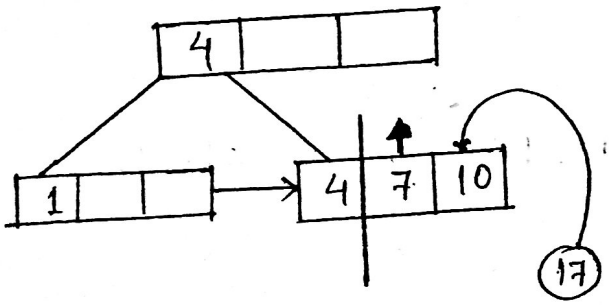
Insert : 4



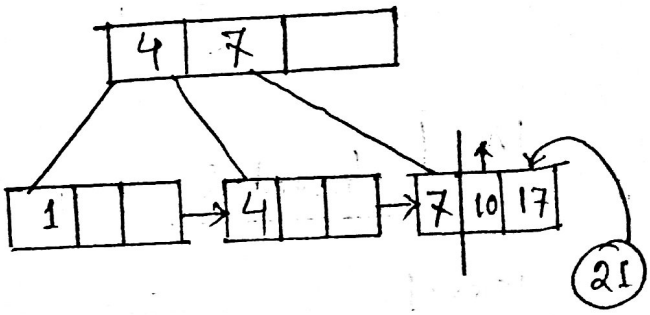
Insert : 7



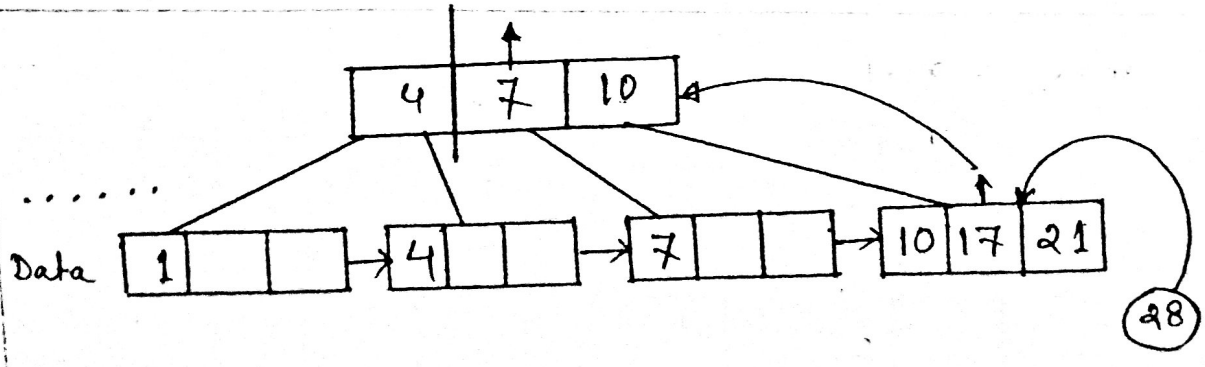
Insert : 10



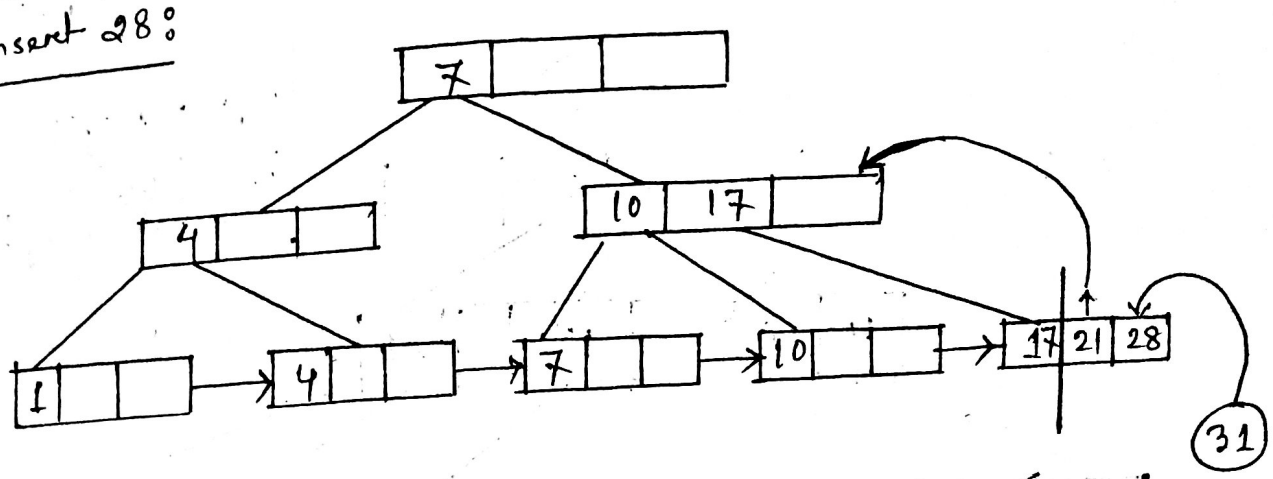
Insert : 17



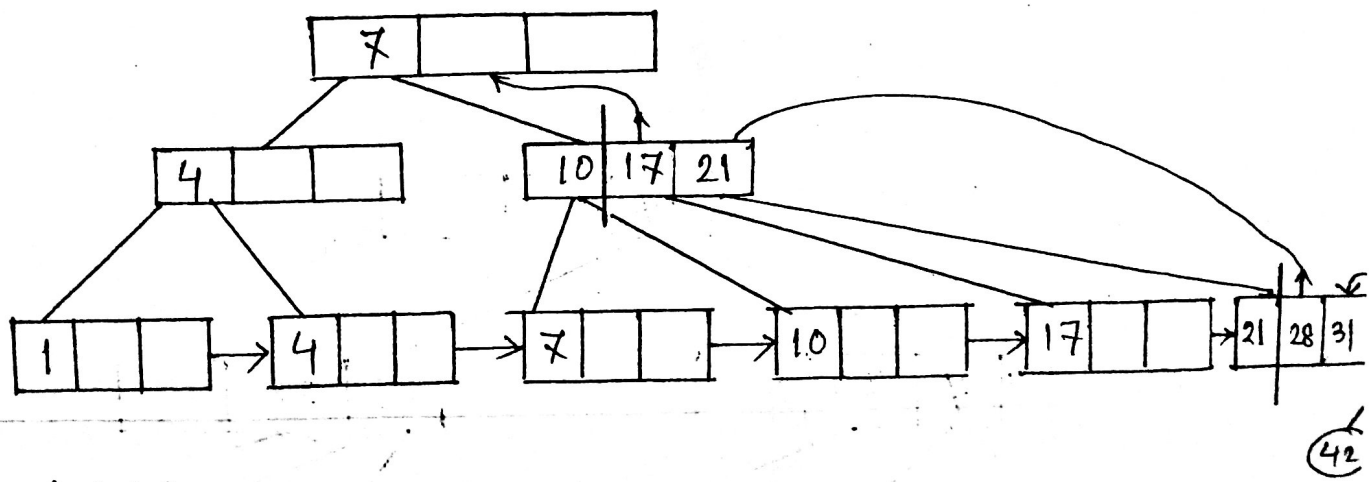
Insert : 21



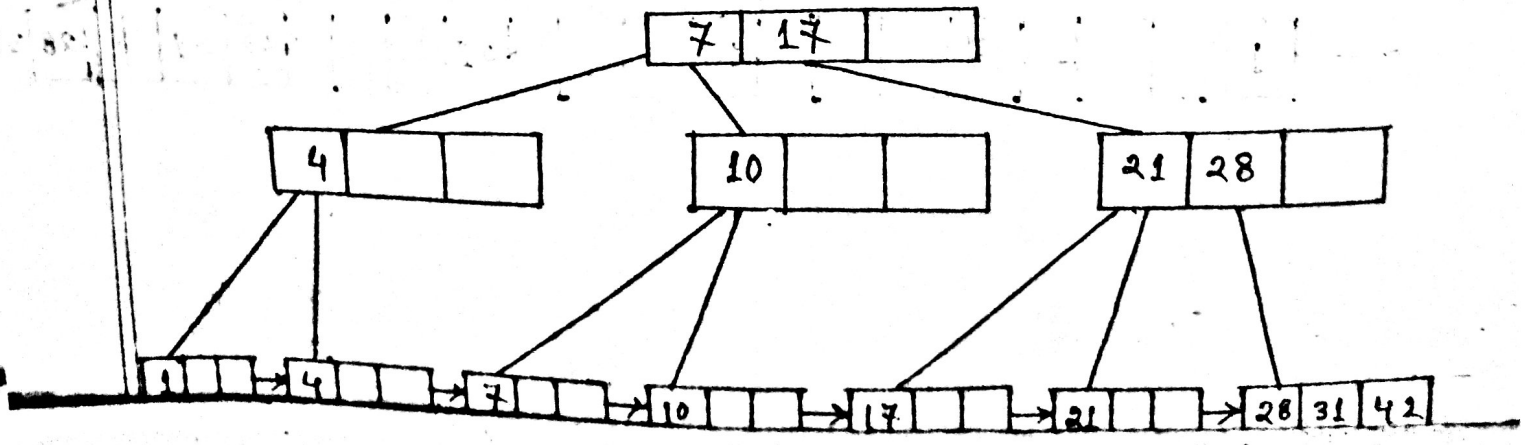
Insert 28:



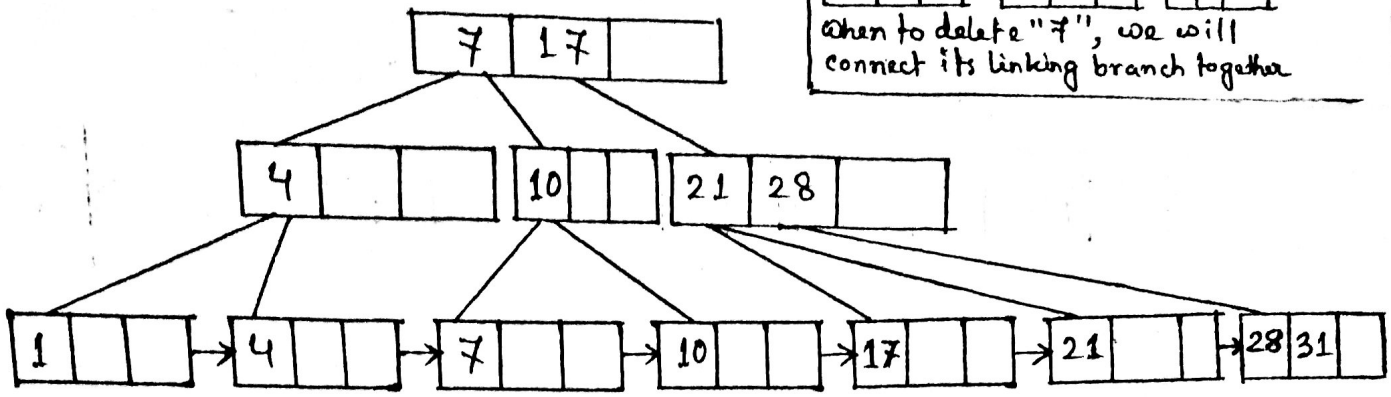
Insert 31:

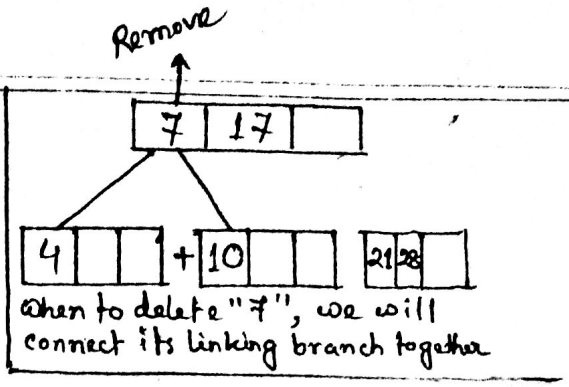


Insert 42:

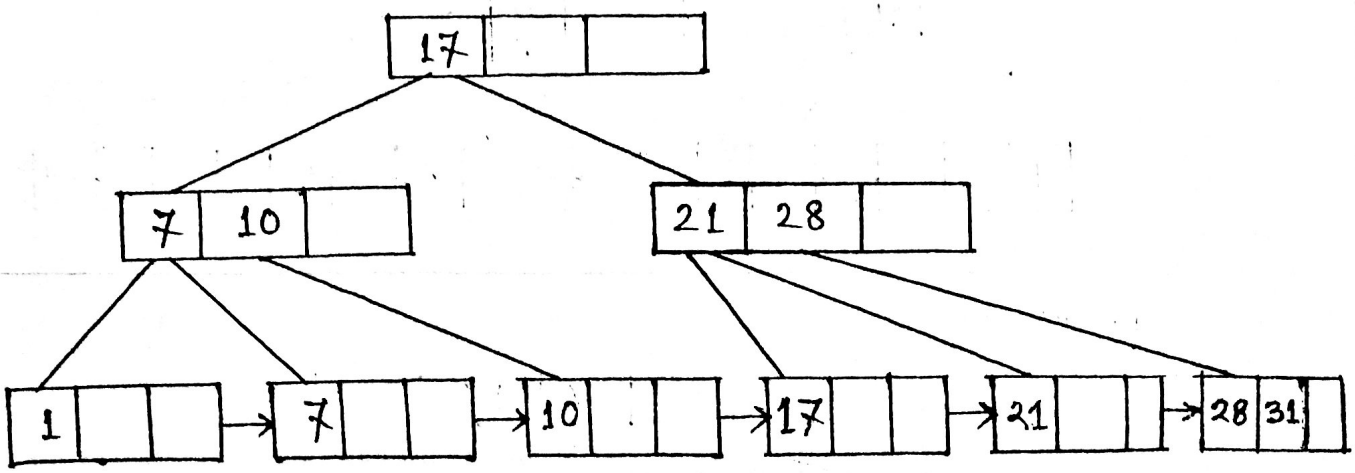


Delete 42:





Delete 4:



13 July 2017

Basic Concepts

* 6ষ্ঠ parameter হিসেবে যাবে => searchKey ( )

Index Evaluation Matrics

* Space Overhead : কতটুকু Database ও কতগুলো people connected হবে at a time.

* Indexing file is very important!

Ordered Indices

→ Primary Index [Sorted অবস্থায়]

→ Secondary Index (Non-clustering Index)
[Different order এ থাকে]

Dense Index file

→ প্রত্যেক id এর জন্য pointer দেওয়া থাকে which selects row.

→ Internal data এর জন্য বারবার sorting লাগে।

Sparse Index file.

→ কিছু কিছু search key value নিয়ে কাজ করে।

Important Questions [3-4 Marks করবে]

i) Indexing কাকে বলে ? why it is important [2]

ii) Indexing evolution এর ওপর একটি Matrix করবে? [3]

iii) Indexing basically কত-প্রকার? [2]

iv) Dense Index file & Sparse Index file এর difference. [4] or [5] (follow book)

v) what is Multilevel Indexing ? Give proper example. [2+1]

and,

B+ Tree থেকে 8-9 Marks কমন।

(এই এর example follow করবে)

17 July 2017

Query Processing

Basic Steps in Query Processing

```
Select name
From instruction
where salary < 25000;
```

$\Pi_{name}(instruction)$

$\sigma_{salary < 25000}$

$\Pi_{name}(\sigma_{salary<25000}(Instruction))$

$\sigma_{salary<25000}(\Pi_{name}(Instruction))$

Instruction

ID	name	department	salary

$0+(A+B)\times C$

20 July 2017

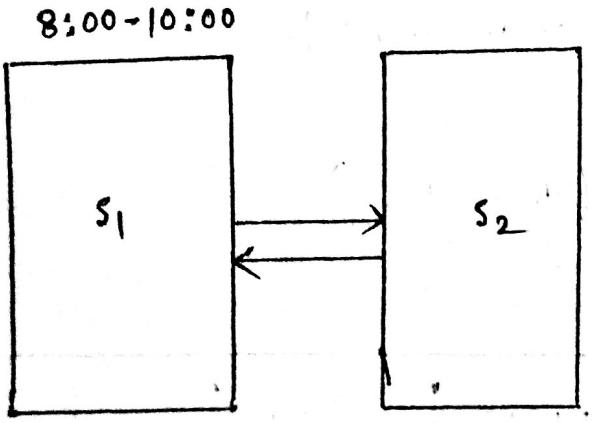
Transaction

* Database-কে consistent রাখা দুরুহ।

* Transaction করার সময় Database inconsistent অবস্থায় চলে যায়। সময়ে Database-এর consistency check করতে।

* Consistency check করার ভাল একটি way →

"Data Mirroring"



Server 1 Server 2

Server 1	Server 2
→ 6:00 - 10:00	Backup
→ Full run	6:00-10:00
→ 10 টার পর	10 টার পর
→ Backup	Full load

* isolation এর অনেকটা কাজ হচ্ছে isolate way create করা যেন
কারোরটা কেউ দেখতে না পারে যে যে কাজ আর কেউ করছে।

** Q. What are the ACID properties of Database?

* Consistency এর জন্য Database কে isolation property
-দিতে হবে।

* Database যখন কাজ করে তখন সেটা তার Active state!

* Committed => Both party যখন খবর বুঝে নেয় যে, fully
Done!

* Abort → Kill
→ Restart

30 July 2017

Schedule

→ Schedule 1
→ Schedule 2
→ schedule 3
→ Schedule 4

⊞ Serializability

→ Conflict Serializability
→ View Serializability

⊞ Conflicting Instructions

যখন দুটি instruction 'Construct' (Read) অবস্থায়

থাকে।

⊞ Classification of physical Storage Media

→ speed ও size এর ওপর depend করে
price নির্ধারণ করা হয়।

31 July 2017

⊞ Magnetic Hard Disk Mechanism

⊞ RAID : Redundant Arrays of Independent Disks

RAID 0 : সকল Hard Drive কে stripping করে।

RAID 1 : Mirroring করে (with block striping)
↓
Copying

RAID 2,3 : সবচেয়ে বাজে

RAID 4 : extra Hard drive থাকে, parity হিসেবে কাজ করে, Hot-swappable নয়।

RAID 5 : Block-Interleaved Distributed Parity, প্রত্যেকটি parity বিভিন্ন Geo location এ distributed, No-switch-off needed (Hot swappable)

* Latency কমাতে পারলে, Hard Drive or any Component এর আয়ু প্রায় দ্বিগুণ বাড়ে।

* RAID ২ ধরনের পাওয়া যায় → software RAID
Hardware RAID

* RAID 5, RAID 10 → World এর সবচেয়ে popular version
↓
one zero
(not Ten)

SAN ⇒ Storage Area Network
([illegible])

03 August 2017

Chapter – 17

Database System Architectures

Client-Server Systems (Front-end)

→ অনেকগুলো client থাকবে

Server (Backend)

→ কোনো user interface নেই, data access এর ability থাকে।

* যখনই page এর সাথে Connection Establish করা যায়।

* API (Application Program Interface)

→ Frontend এবং Back-end এর মধ্যে connection establish করে।

☐ Interconnection Architectures

(a) bus

(b) mesh

(c) hypercube [faster than 'mesh']

* Shared Disk এ প্রতিটি processor এর individual memory থাকে।

10 August 2017

(Question Pattern Instructions)

* কোনো Mandatory question no. থাকবে না।

1টি Question-এ 5টি পর্যন্ত ছোট থাকতে পারে।

1. Time
2. How Much shall I write
3. Seriality
4. Diagram (Landscape-এ আঁকবে)

[কোশ্চিন দিবে]

* সবগুলো Question mixed set আসবে, শুধু SQL থেকে full 1 set আসবে।

* Relational Algebra থেকে 11/12 marks-এর Question আসবে।

* ERD → 8/9 +
* B+ Tree → 8/9 +
* Normalization → 8/9 +
* Transaction → 7/8 +
* RAID → 4/5

} বাকি Marks short Question থেকে আসবে!

** "Database System Concepts" বই পড়ব +

→ How to draw an ERD ; [7 examples]

→ B^+ Tree (example দেখব)

→ Relational Algebra + SQL এর বই পড়ব
   ↳ (No Partial Marking)

⊞ Last 2-3 টা Semester এর Carry + Clearance + main Question দেখব !!