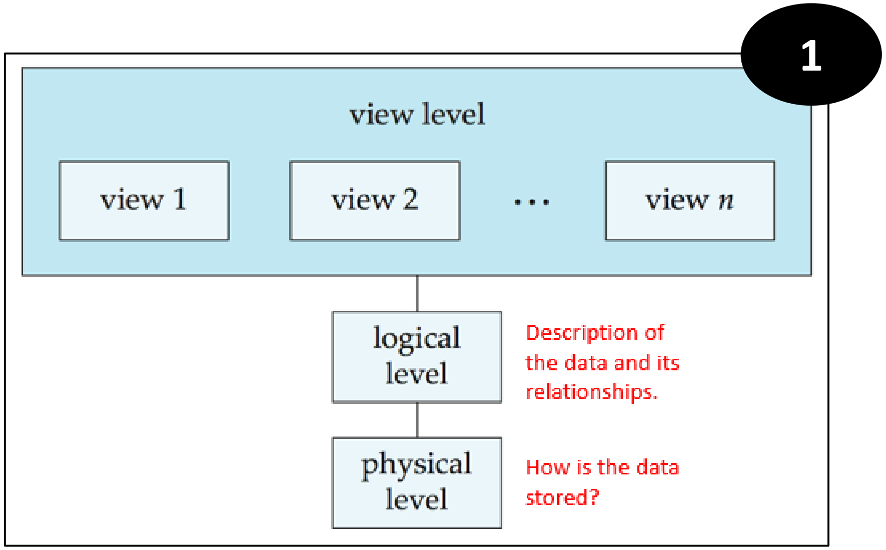
RDBMS and SQL
Functional Dependencies

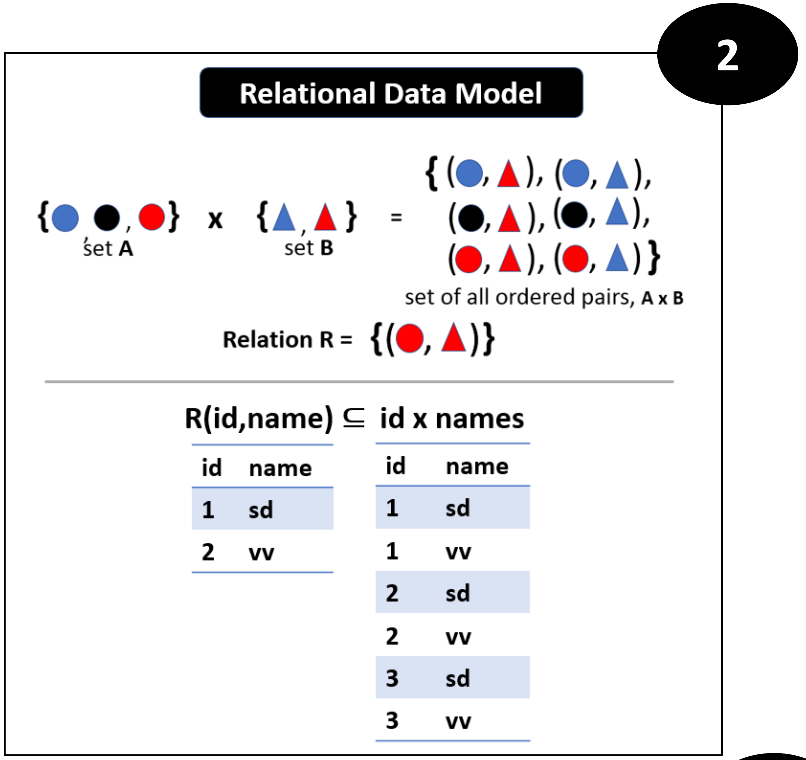
Venkatesh Vinayakarao
venkateshv@cmi.ac.in
http://vvtesh.co.in

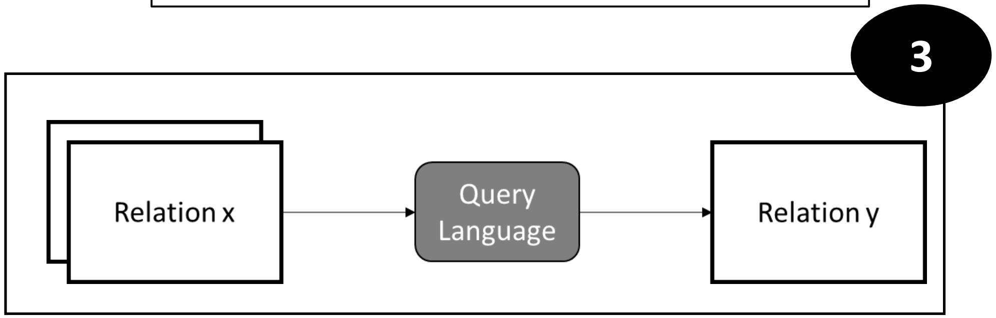
Chennai Mathematical Institute

Slide contents are borrowed from the course text. For the authors' original version of slides, visit: https://www.db-book.com/db6/slide-dir/index.html.

Review


1
view level
view 1
view 2
...
view n
logical level
Description of the data and its relationships.
physical level
How is the data stored?


2
Relational Data Model
set A
x
set B
=
set of all ordered pairs, A x B
Relation R =
R(id,name) ⊆ id x names
id name
1 sd
2 vv
id name
1 sd
1 vv
2 sd
2 vv
3 sd
3 vv


3
Relation x
Query Language
Relation y

Relational Algebra and SQL

- Find all courses taught in the Fall 2009 semester, or in the Spring 2010 semester, or in both

$$\Pi_{course\_id}\ (\sigma_{semester=\text{“Fall”}\ \land\ year=2009}\ (section))\ \cup$$

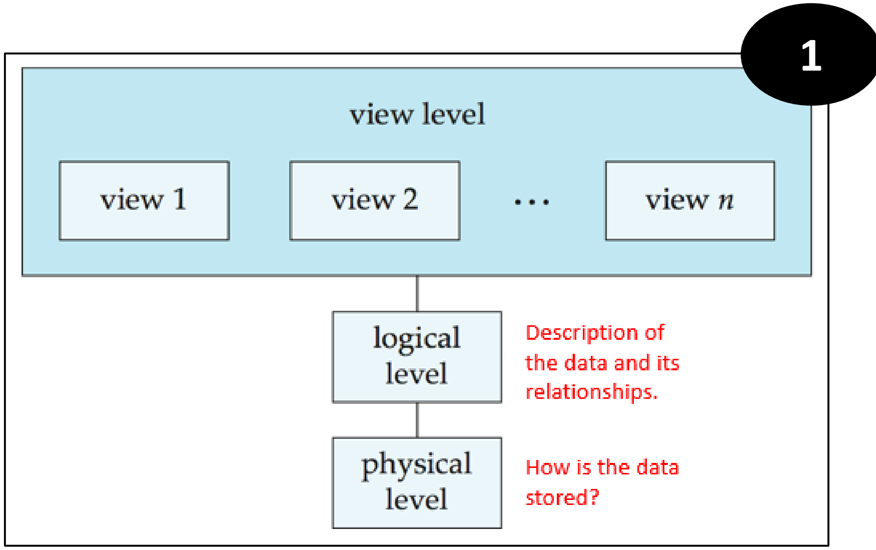
$$\Pi_{course\_id}\ (\sigma_{semester=\text{“Spring”}\ \land\ year=2010}\ (section))$$

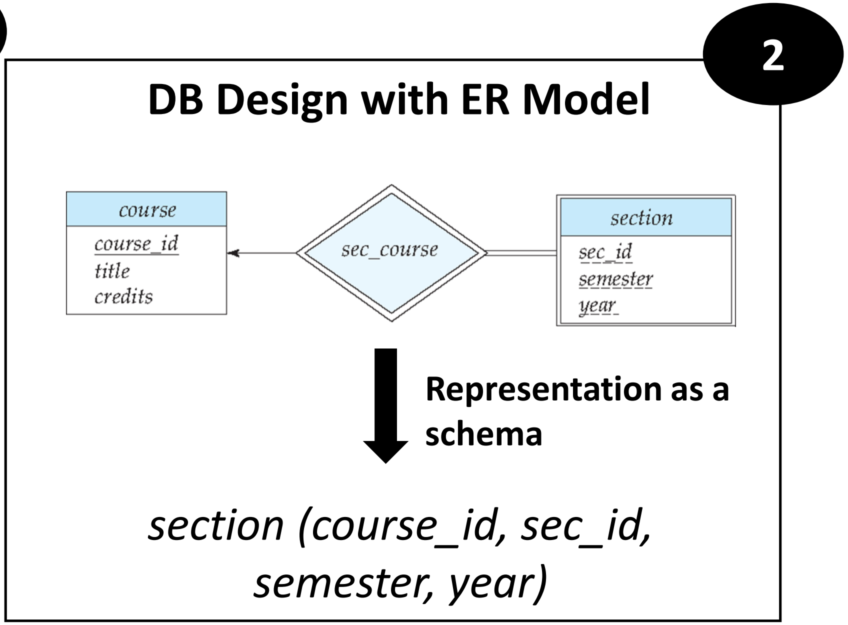
- Structured Query Language

(**select** *course_id* **from** *section* **where** *sem* = ‘Fall’ **and** *year* = 2009)
union
(**select** *course_id* **from** *section* **where** *sem* = ‘Spring’ **and** *year* = 2010)

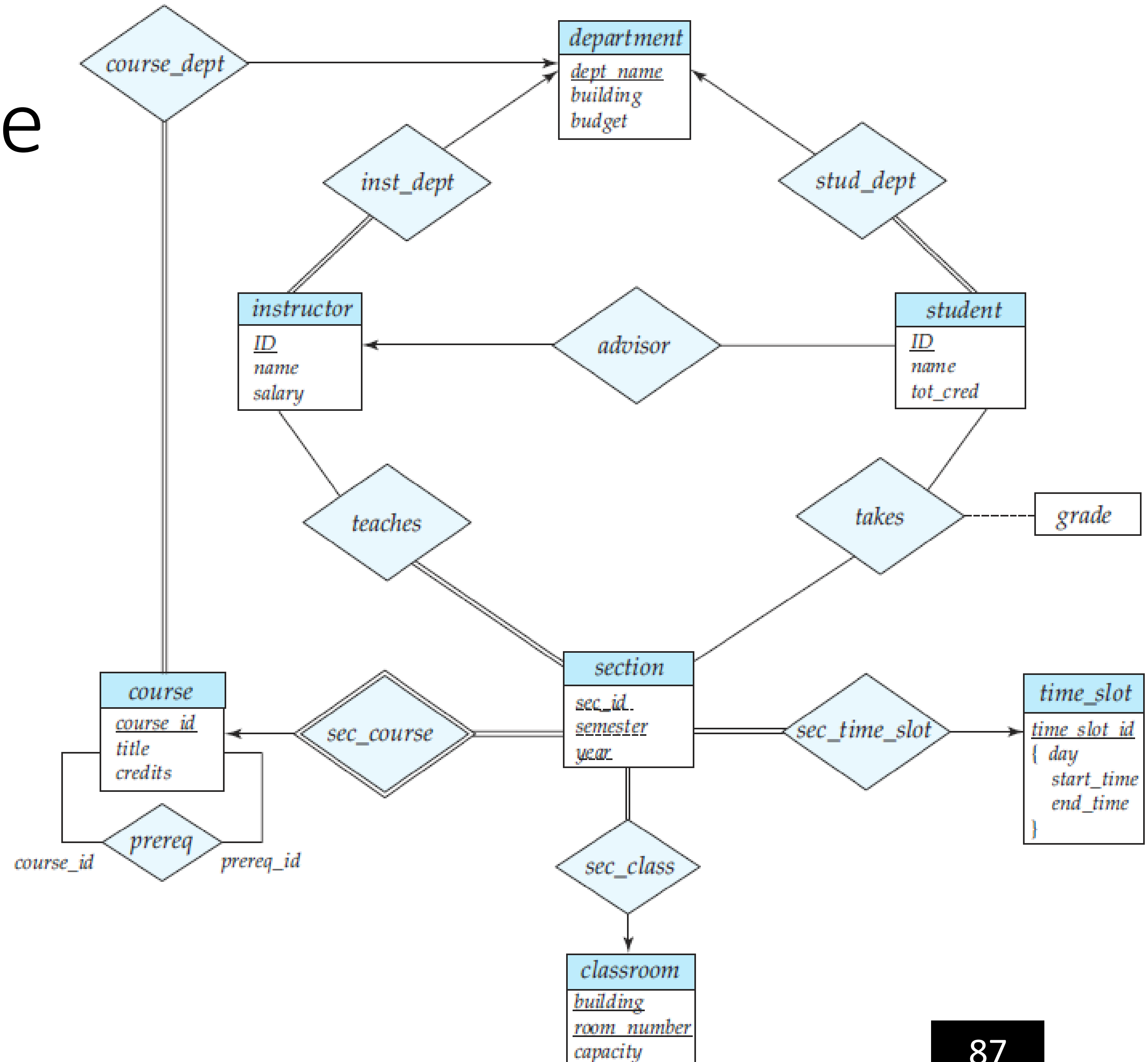
We will revisit SQL in detail in future classes.

Review


1
view level
view 1
view 2
. . .
view n
logical level
Description of the data and its relationships.
physical level
How is the data stored?


2
DB Design with ER Model
course
course_id
title
credits
sec_course
section
sec_id
semester
year
Representation as a schema
section (course_id, sec_id, semester, year)

Example

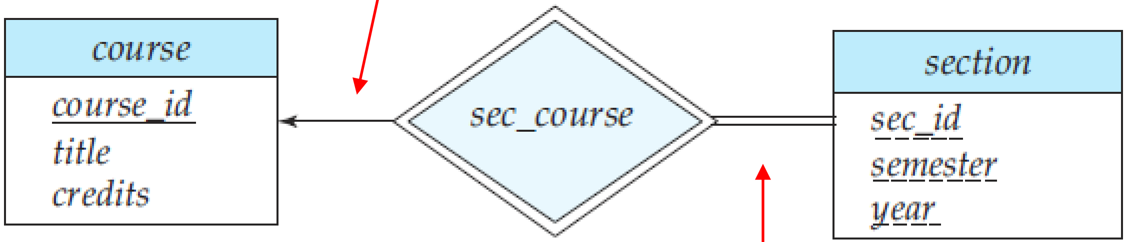


course_dept
department
dept_name
building
budget
inst_dept
stud_dept
instructor
ID
name
salary
advisor
student
ID
name
tot_cred
teaches
takes
grade
section
sec_id
semester
year
course
course_id
title
credits
sec_course
sec_time_slot
time_slot
time_slot_id
{ day
start_time
end_time
}
prereq
course_id
prereq_id
sec_class
classroom
building
room_number
capacity

Notation

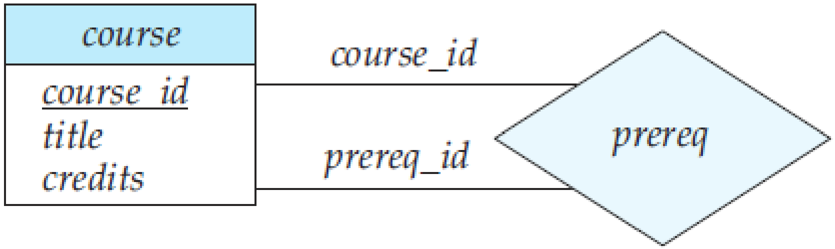
Double Diamond: section is existence dependent on course

Arrow: Many (section) to one (course) relationship.

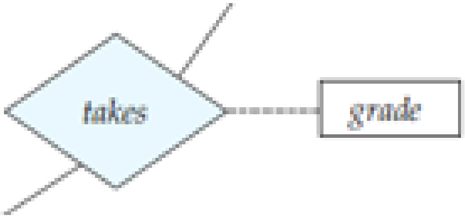


Double Line: Total Participation
Every section is associated with a course

Notation

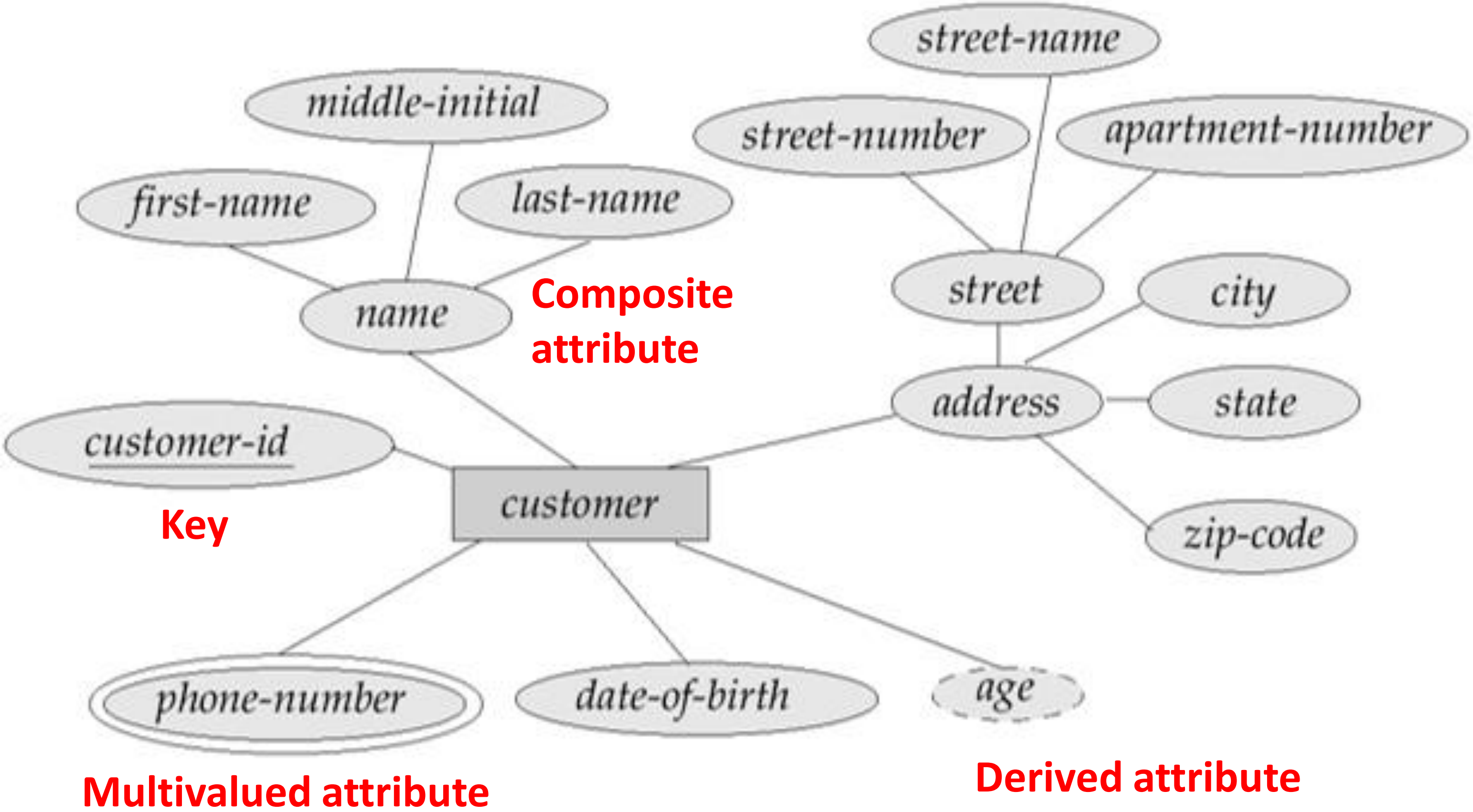


Same entity with multiple roles



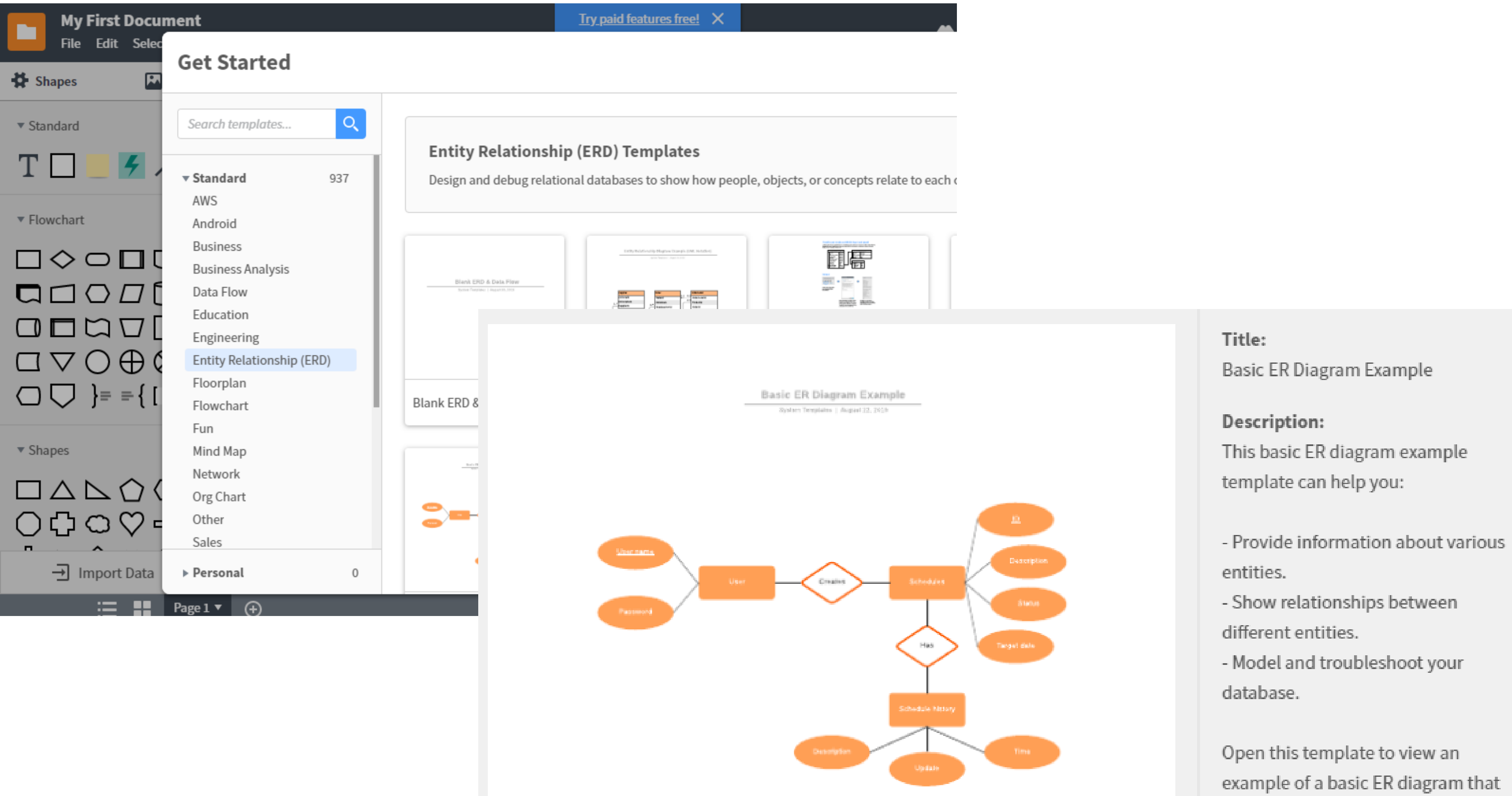
Attribute associated to a relationship

Example

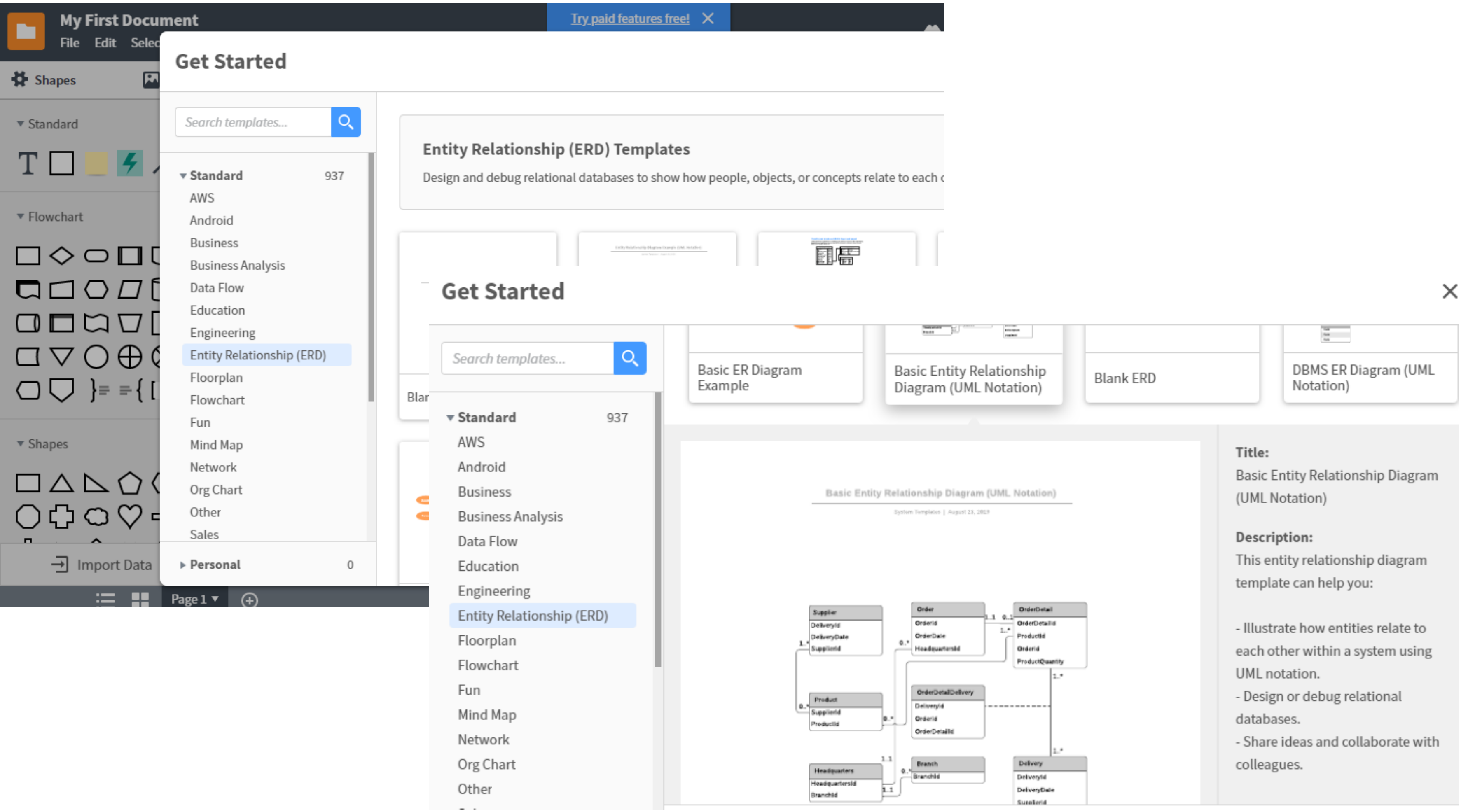


street-name
middle-initial
street-number
apartment-number
first-name
last-name
street
city
name
Composite
attribute
address
state
customer-id
customer
Key
zip-code
phone-number
date-of-birth
age
Multivalued attribute
Derived attribute

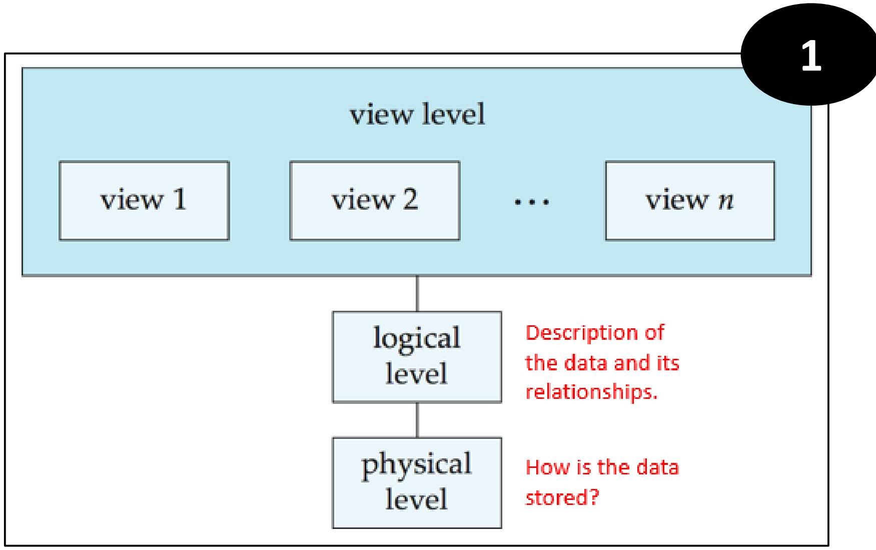
Lucidchart.com



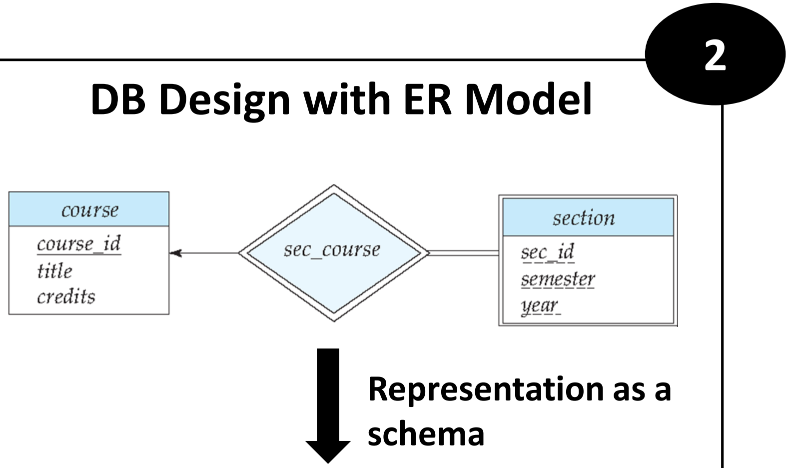
Lucidchart.com



Review



DB Design with ER Model



section (course_id, sec_id, semester, year)

What is a good design?

Redundancy

- Is there any problem with this relation?

ID	*name*	*salary*	*dept_name*	*building*	*budget*
22222	Einstein	95000	Physics	Watson	70000
12121	Wu	90000	Finance	Painter	120000
32343	El Said	60000	History	Painter	50000
45565	Katz	75000	Comp. Sci.	Taylor	100000
98345	Kim	80000	Elec. Eng.	Taylor	85000
76766	Crick	72000	Biology	Watson	90000
10101	Srinivasan	65000	Comp. Sci.	Taylor	100000
58583	Califieri	62000	History	Painter	50000
83821	Brandt	92000	Comp. Sci	Taylor	100000
15151	Mozart	40000	Music	Packard	80000
33456	Gold	87000	Physics	Watson	70000
76543	Singh	80000	Finance	Painter	120000

Functional Dependency

ID uniquely determines name.

ID → name

ID	*name*	*dept_name*	*salary*
22222	Einstein	Physics	95000
12121	Wu	Finance	90000
32343	El Said	History	60000
45565	Katz	Comp. Sci.	75000
98345	Kim	Elec. Eng.	80000
76766	Crick	Biology	72000
10101	Srinivasan	Comp. Sci.	65000
58583	Califieri	History	62000
83821	Brandt	Comp. Sci.	92000
15151	Mozart	Music	40000
33456	Gold	Physics	87000
76543	Singh	Finance	80000

(a) The *instructor* table

What are the other FDs in instructor(ID,name, dept_name,salary?

ID → ID, dept_name, salary

Quiz

- From the following instance of a relational schema R(A, B, C), we can conclude that:

A	B	C
1	1	1
1	1	0
2	3	2
2	3	2

a) A functionally determines B and B functionally determines C
b) A functionally determines B and B does not functionally determine C
c) B does not functionally determine C
d) A does not functionally determine B and B does not functionally determine C

Quiz

- From the following instance of a relational schema R(A, B, C), we can conclude that: [GATE CS 2002]

A	B	C
1	1	1
1	1	0
2	3	2
2	3	2

a) A functionally determines B and B functionally determines C
b) A functionally determines B and B does not functionally determine C
c) B does not functionally determine C
d) A does not functionally determine B and B does not functionally determine C

Key

ID → ID, name, dept_name, salary

ID	*name*	*dept_name*	*salary*
22222	Einstein	Physics	95000
12121	Wu	Finance	90000
32343	El Said	History	60000
45565	Katz	Comp. Sci.	75000
98345	Kim	Elec. Eng.	80000
76766	Crick	Biology	72000
10101	Srinivasan	Comp. Sci.	65000
58583	Califieri	History	62000
83821	Brandt	Comp. Sci.	92000
15151	Mozart	Music	40000
33456	Gold	Physics	87000
76543	Singh	Finance	80000

(a) The *instructor* table

ID is a key

Key is a set of attributes that uniquely determine an entity in the entity set.

Prime Attribute

- All attributes that form the key are called prime attributes.

R1(<u>AB</u>CD)

AB is the key.
A and B are prime attributes.

Quiz

- Consider the relation schema R = {E, F, G, H, I, J, K, L, M, N} and the set of functional dependencies

$$\{\mathbf{EF} \rightarrow \mathbf{G},\ \mathbf{F} \rightarrow \mathbf{IJ},\ \mathbf{EH} \rightarrow \mathbf{KL},\ \mathbf{K} \rightarrow \mathbf{M},\ \mathbf{L} \rightarrow \mathbf{N}\}$$

on R. What is the key for R? [GATE CS 2014]

(A) {E, F}
(B) {E, F, H}
(C) {E, F, H, K, L}
(D) {E}

Quiz

- Consider the relation scheme R = {E, F, G, H, I, J, K, L, M, N} and the set of functional dependencies {{**E**, **F**} → {**G**}, {**F**} → {**I**, **J**}, {**E**, **H**} →{**K**, **L**}, **K** → {**M**}, **L** → {**N**}} on R. What is the key for R? [GATE CS 2014]

(A) {E, F}
(B) {E, F, H}
(C) {E, F, H, K, L}
(D) {E}

*For notational convenience,
we refer to {E,F} → {G} as EF → G.

Candidate Keys

- Some relations may have many candidate keys.
  - Several sets of key attributes uniquely identify the entity.
- You may choose any one to be the primary key.

QUIZ

- Relation R has eight attributes ABCDEFGH. F = {**CH** -> **G**, **A** -> **BC**, **B** -> **CFH**, **E** -> **A**, **F** -> **EG**} is a set of functional dependencies (FDs). How many candidate keys does the relation R have? [GATE CS 2013]

QUIZ

- Relation R has eight attributes ABCDEFGH. F = {**CH** -> **G**, **A** -> **BC**, **B** -> **CFH**, **E** -> **A**, **F** -> **EG**} is a set of functional dependencies (FDs). How many candidate keys does the relation R have? [GATE CS 2013]

Solution:

Find the closure of FDs:

A+ = ABCEFGH (D is missing)

B+ = ABCEFGH (D is missing)

E+ = ABCEFGH (D is missing)

F+ = ABCEFGH (D is missing)

So, AD, BD, ED and FD are candidate keys.

Super Key

- Super Key = Candidate Key + Additional Attributes

- QUIZ: Which of the following is NOT a superkey in a relational schema with attributes V, W, X, Y, Z and primary key V Y ?
  - **(A)** V X Y Z
  - **(B)** V W X Z
  - **(C)** V W X Y
  - **(D)** V W X Y Z

Super Key

- Super Key = Candidate Key + Additional Attributes

- QUIZ: Which of the following is NOT a superkey in a relational schema with attributes V, W, X, Y, Z and primary key V Y ? [GATE CS 2016]

  (A) V X Y Z
  (B) V W X Z
  (C) V W X Y
  (D) V W X Y Z

Avoiding Redundant Dependencies

- Consider

  $R = (A, B, C)$
  with FDs, $F = \{A \rightarrow BC, B \rightarrow C, A \rightarrow B, AB \rightarrow C\}$

- Do you find any redundancy in FDs?

Computing a Canonical Cover

- *Given, R = (A, B, C). F =* $\{A \rightarrow BC,\ B \rightarrow C,\ A \rightarrow B,\ AB \rightarrow C\}$

$$\left.\begin{array}{l} A \rightarrow BC \\ A \rightarrow B \end{array}\right\} \quad A \rightarrow BC$$

Absorption

Drop $A \rightarrow B$

$$\left.\begin{array}{l} AB \rightarrow C \\ B \rightarrow C \end{array}\right\} \quad B \rightarrow C$$

Drop Extraneous Attributes

Drop $AB \rightarrow C$

$$\left.\begin{array}{l} A \rightarrow BC \\ A \rightarrow C \end{array}\right\} \quad \begin{array}{l} A \rightarrow B \\ A \rightarrow C \end{array}$$

Drop Extraneous Attributes

Change $A \rightarrow BC$ to $A \rightarrow B$

- The canonical cover is:

$$\begin{array}{l} A \rightarrow B \\ B \rightarrow C \end{array}$$

*An attribute of is said to be extraneous if we can remove it without changing the closure that includes the set of functional dependencies.

Normal Forms

First Normal Form

- A relation is in first normal form iff
  - the domain of each attribute contains only atomic values, and
  - the value of each attribute contains only a single value from that domain

Not in 1NF.
There are multiple values for telephone number.

Customer

Customer ID	First Name	Surname	Telephone Number
123	Pooja	Singh	555-861-2025, 192-122-1111
456	San	Zhang	(555) 403-1659 Ext. 53; 182-929-2929
789	John	Doe	555-808-9633

First Normal Form

- To convert it into 1NF:

Customer

Customer ID	First Name	Surname	Telephone Number
123	Pooja	Singh	555-861-2025, 192-122-1111
456	San	Zhang	(555) 403-1659 Ext. 53; 182-929-2929
789	John	Doe	555-808-9633

↓

Customer

Customer ID	First Name	Surname	Telephone Number
123	Pooja	Singh	555-861-2025
123	Pooja	Singh	192-122-1111
456	San	Zhang	182-929-2929
456	San	Zhang	(555) 403-1659 Ext. 53
789	John	Doe	555-808-9633

*Example source: Wikipedia.

Why not?

Customer

Customer ID	First Name	Surname	Telephone Number
123	Pooja	Singh	555-861-2025, 192-122-1111
456	San	Zhang	(555) 403-1659 Ext. 53; 182-929-2929
789	John	Doe	555-808-9633

Customer

Customer ID	First Name	Surname	Telephone Number1	Telephone Number2
123	Pooja	Singh	555-861-2025	192-122-1111
456	San	Zhang	(555) 403-1659 Ext. 53	182-929-2929
789	John	Doe	555-808-9633	

Is this a good design?

Second Normal Form (2NF)

- A relation is in second normal form iff
  - it is in 1nf, and
  - every non-prime attribute is not functionally dependent on any proper subset of any candidate key.

No Partial Dependency!

Electric toothbrush models

Manufacturer	Model	Model full name	Manufacturer country
Forte	X-Prime	Forte X-Prime	Italy
Forte	Ultraclean	Forte Ultraclean	Italy
Dent-o-Fresh	EZbrush	Dent-o-Fresh EZbrush	USA
Brushmaster	SuperBrush	Brushmaster SuperBrush	USA
Kobayashi	ST-60	Kobayashi ST-60	Japan
Hoch	Toothmaster	Hoch Toothmaster	Germany
Hoch	X-Prime	Hoch X-Prime	Germany

Manufacturer → Manufacturer Country
{Manufacturer, Model} → Model full name

Second Normal Form (2NF)

- To convert it into 2NF, we decompose the relation:

Electric toothbrush manufacturers

Manufacturer	Manufacturer country
Forte	Italy
Dent-o-Fresh	USA
Brushmaster	USA
Kobayashi	Japan
Hoch	Germany

Electric toothbrush models

Manufacturer	Model	Model full name
Forte	X-Prime	Forte X-Prime
Forte	Ultraclean	Forte Ultraclean
Dent-o-Fresh	EZbrush	Dent-o-Fresh EZbrush
Brushmaster	SuperBrush	Brushmaster SuperBrush
Kobayashi	ST-60	Kobayashi ST-60
Hoch	Toothmaster	Hoch Toothmaster
Hoch	X-Prime	Hoch X-Prime

Third Normal Form (3NF)

- A relation is in third normal form iff
  - it is in 2NF, and
  - there is no transitive dependency for non-prime attributes

<u>ID</u>	name	zip	state	city
1	Srinivas	600014	TN	Chennai
2	Lakshman	500012	TS	Hyderabad
3	Razaq	400012	MH	Mumbai
4	Karthik	601103	TN	Chennai
5	John	110023	DL	Delhi

ID → zip
zip → state, city

is a transitive dependency

Third Normal Form (3NF)

- Note that for each of its functional dependencies $X \rightarrow Y$, at least one of the following conditions holds:
  - X is a super key
  - Y is a prime attribute, i.e., each element of Y is part of some candidate key.

<u>ID</u>	name	zip
1	Srinivas	600014
2	Lakshman	500012
3	Razaq	400012
4	Karthik	601103
5	John	110023

zip	state	city
600014	TN	Chennai
500012	TS	Hyderabad
400012	MH	Mumbai
601103	TN	Chennai
110023	DL	Delhi

Third Normal Form (3NF)

- R(ABCD); F={AB -> C, ABD -> C, ABC -> D, AC -> D}
- What is the key?
- What are the prime attributes?
- Is R in 3NF?

Third Normal Form (3NF)

- R(ABCD); F={AB -> C, ABD -> C, ABC -> D, AC -> D}
- What is the key?
  - AB
- What are the prime attributes?
  - A, B
- Testing for 3NF
  - AB -> C : LHS is a key and hence a superkey.
  - ABD -> C : LHS is a superkey.
  - ABC -> D : LHS is a superkey.
  - AC -> D : LHS is not a superkey. Right side is not a prime attribute
- Hence, R is not in 3NF.

Thank You