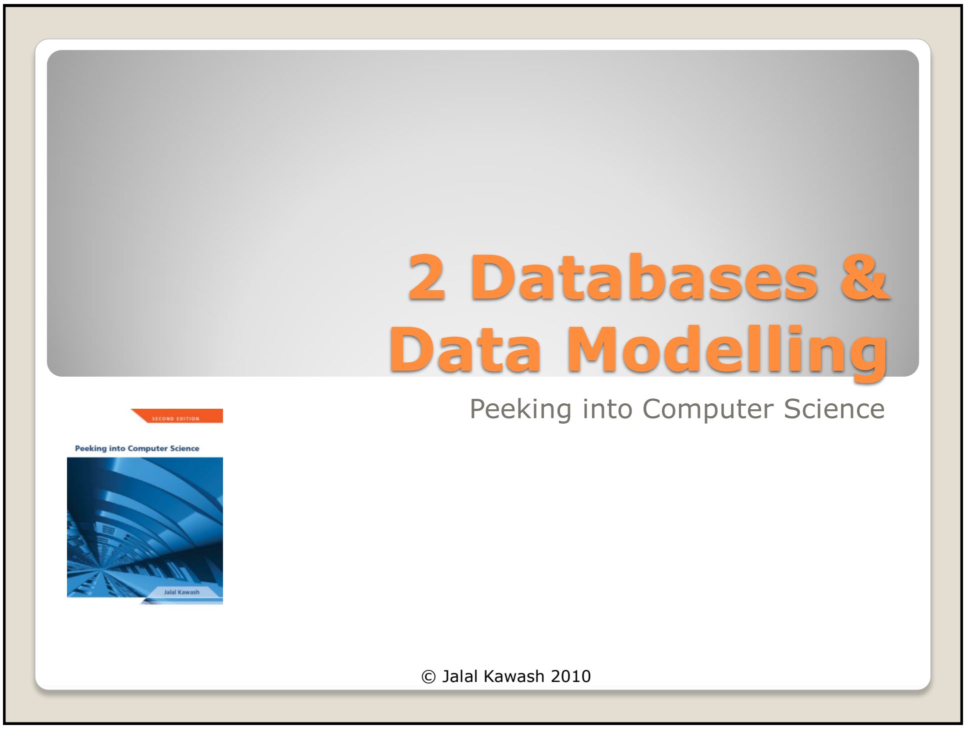
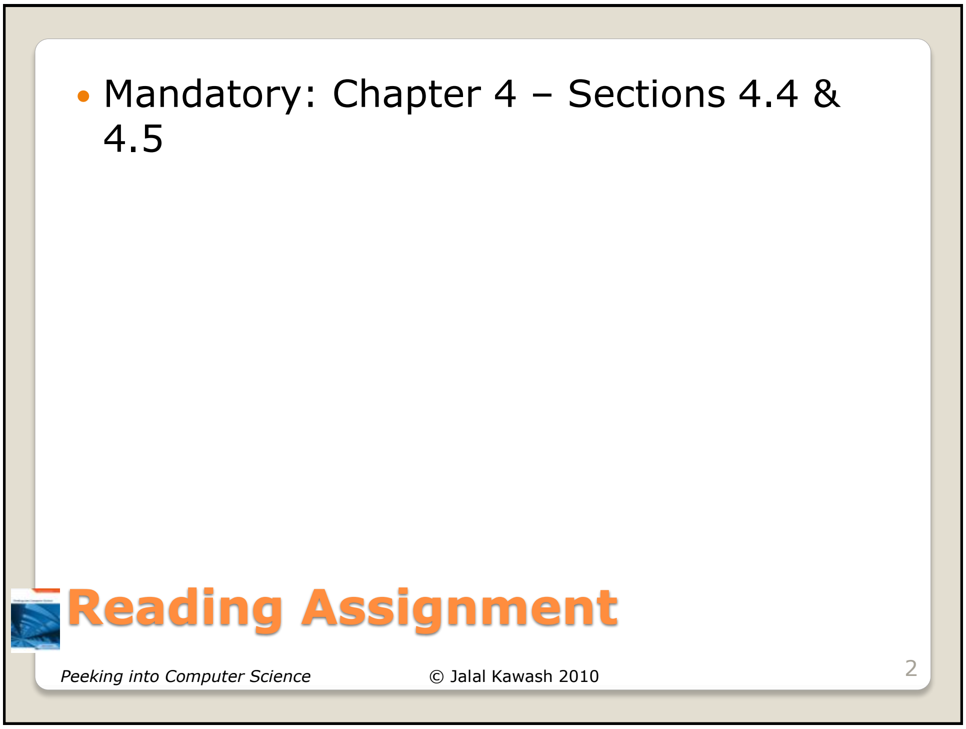

SECOND EDITION
Peeking into Computer Science
Jalal Kawash

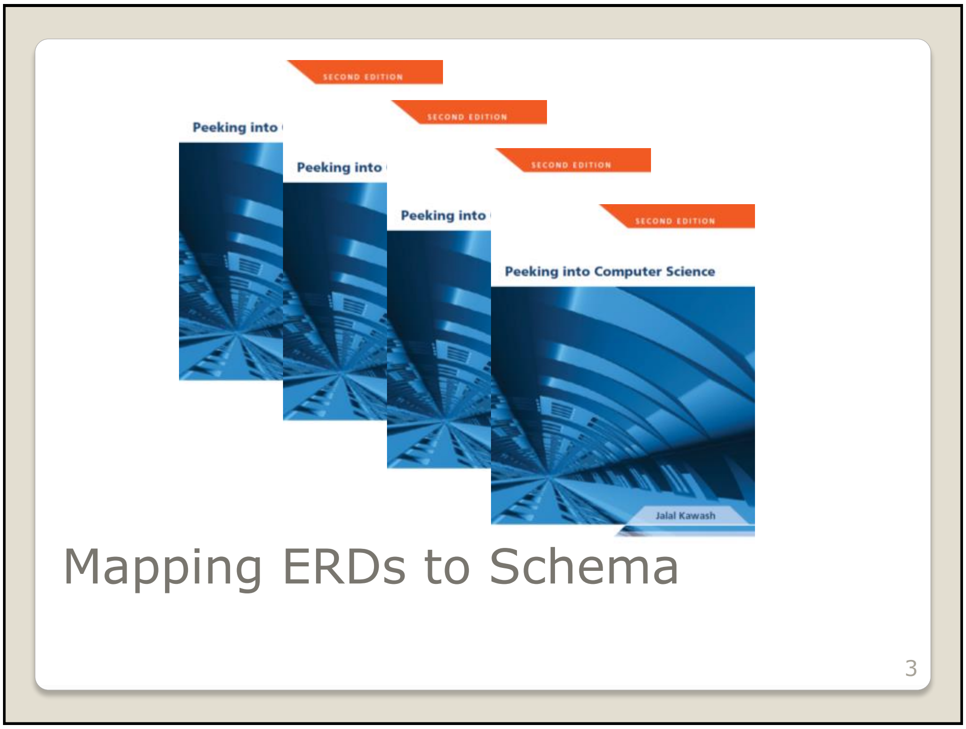
2 Databases & Data Modelling

Peeking into Computer Science



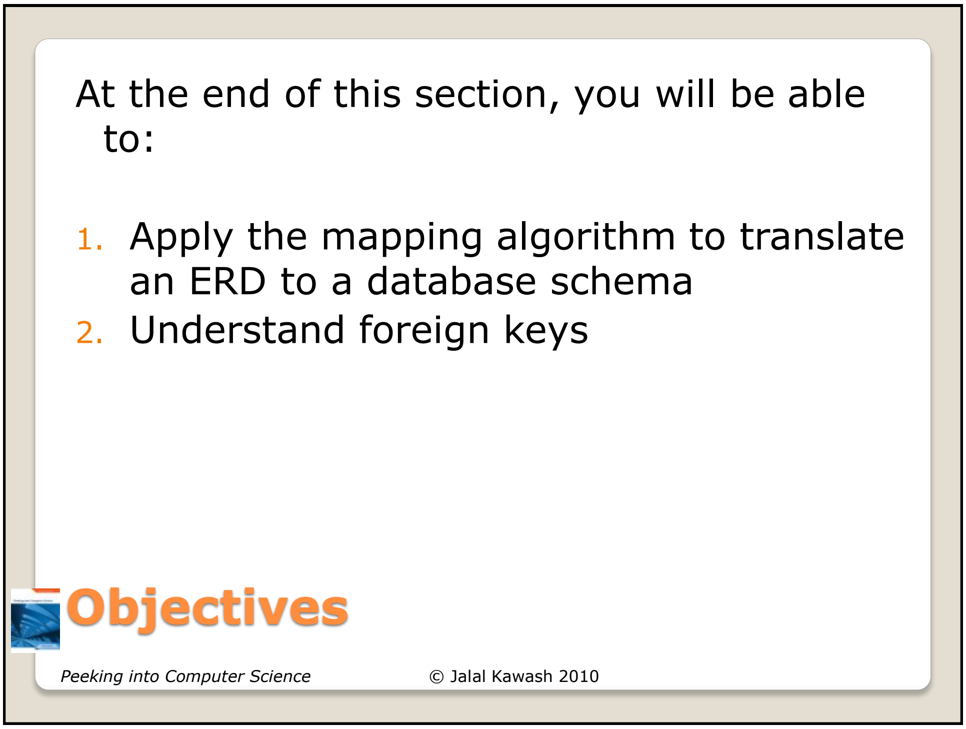
Reading Assignment

- Mandatory: Chapter 4 – Sections 4.4 & 4.5



Mapping ERDs to Schema

3



At the end of this section, you will be able to:

1. Apply the mapping algorithm to translate an ERD to a database schema
2. Understand foreign keys

Objectives

Peeking into Computer Science © Jalal Kawash 2010

Mapping Algorithm (4.1)

1. Each **entity type** is translated to a table; its attributes become columns
2. Each **many-to-many relationship** type becomes a table; the columns are the primary keys of the participating entity types
   - JT: Recall the example from previous notes (Slide #38): Student-Classes
3. For each **one-to-many relationship** type, add the primary keys of the entity type on the one side as columns in the table corresponding to the entity type on the many side

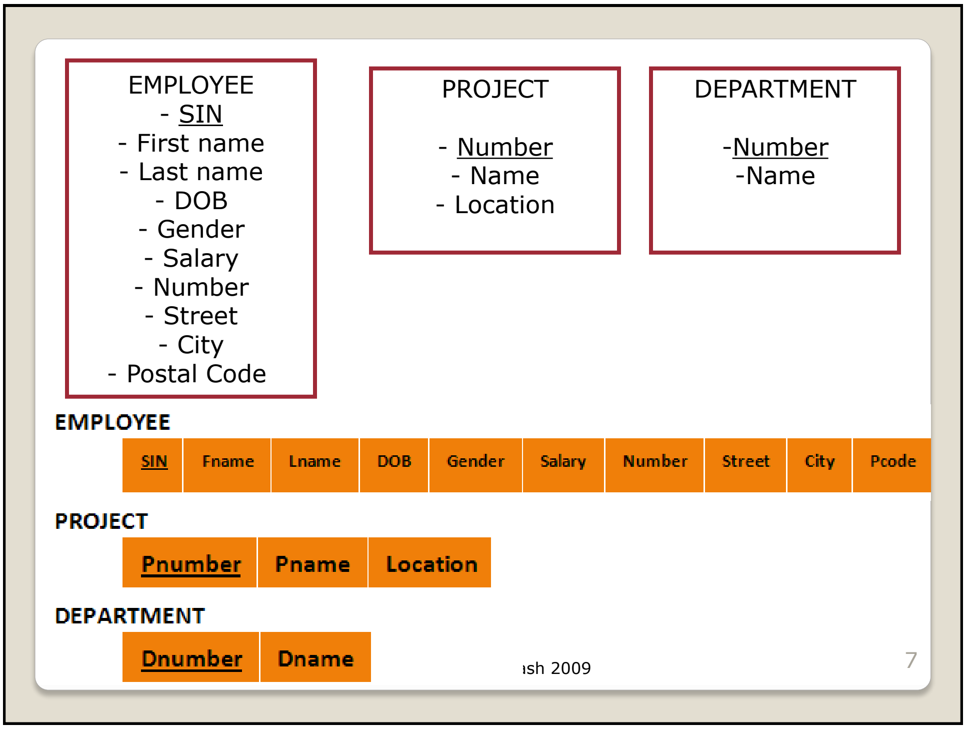
Mapping Entity Types

1. Each entity type is translated to a table; its attributes become columns

EMPLOYEE
- SIN
- First name
- Last name
- DOB
- Gender
- Salary
- Number
- Street
- City
- Postal Code

PROJECT
- Number
- Name
- Location

DEPARTMENT
-Number
-Name



EMPLOYEE
- SIN
- First name
- Last name
- DOB
- Gender
- Salary
- Number
- Street
- City
- Postal Code

PROJECT

- Number
- Name
- Location

DEPARTMENT

-Number
-Name

EMPLOYEE

SIN	Fname	Lname	DOB	Gender	Salary	Number	Street	City	Pcode

PROJECT

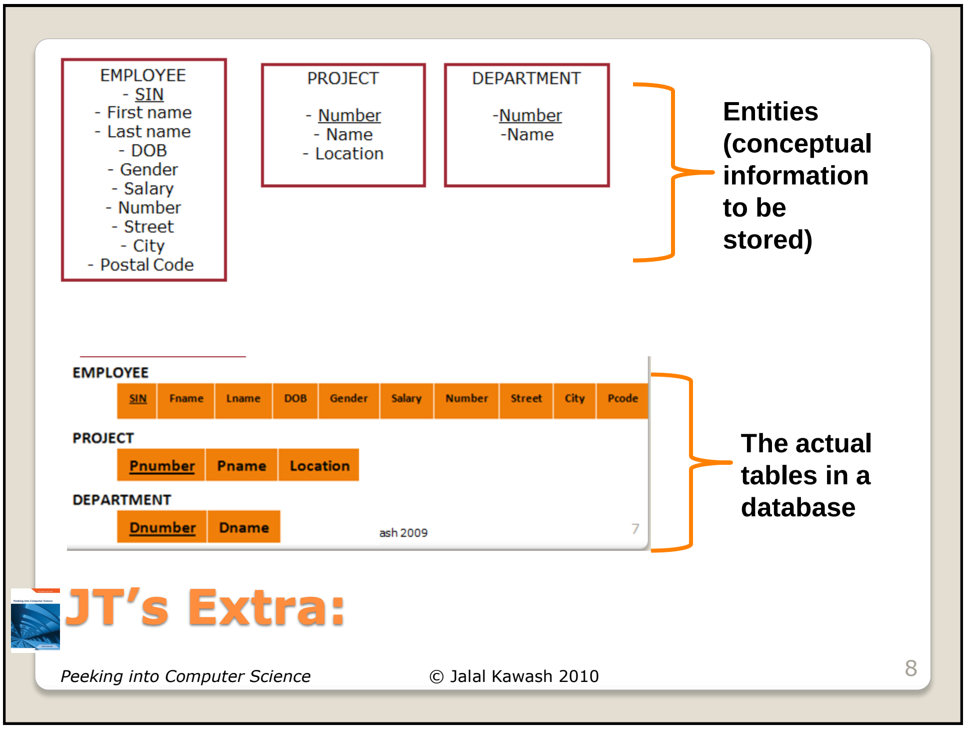
Pnumber	Pname	Location

DEPARTMENT

Dnumber	Dname

ısh 2009

7



EMPLOYEE
- SIN
- First name
- Last name
- DOB
- Gender
- Salary
- Number
- Street
- City
- Postal Code

PROJECT

- Number
- Name
- Location

DEPARTMENT

-Number
-Name

Entities (conceptual information to be stored)

EMPLOYEE

SIN	Fname	Lname	DOB	Gender	Salary	Number	Street	City	Pcode

PROJECT

Pnumber	Pname	Location

DEPARTMENT

Dnumber	Dname

ash 2009

7

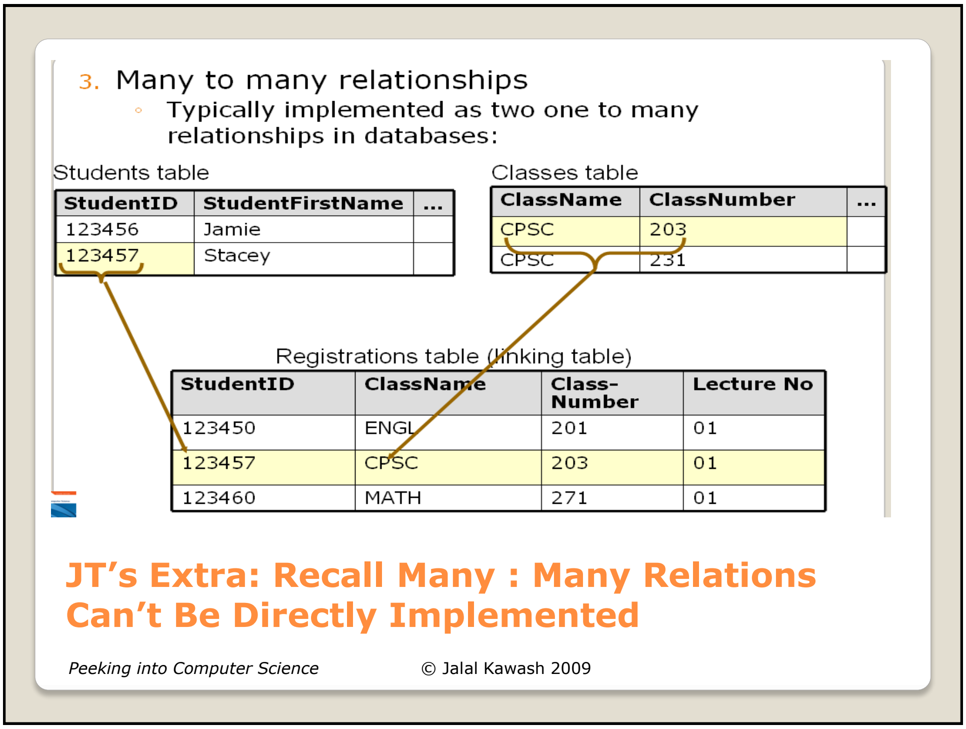
The actual tables in a database

JT's Extra:

Peeking into Computer Science

© Jalal Kawash 2010

8



JT's Extra: Recall Many : Many Relations Can't Be Directly Implemented

3. Many to many relationships
   - Typically implemented as two one to many relationships in databases:

Students table

StudentID	StudentFirstName	…
123456	Jamie	
123457	Stacey	

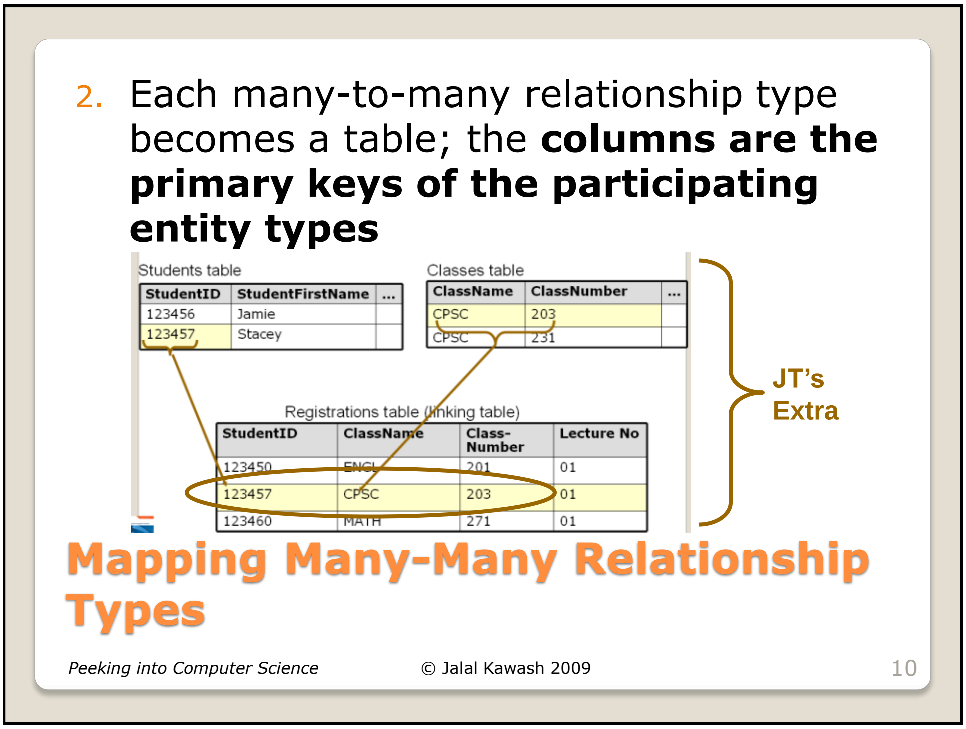
Classes table

ClassName	ClassNumber	…
CPSC	203	
CPSC	231	

Registrations table (linking table)

StudentID	ClassName	Class-Number	Lecture No
123450	ENGL	201	01
123457	CPSC	203	01
123460	MATH	271	01

Peeking into Computer Science © Jalal Kawash 2009



Mapping Many-Many Relationship Types

2. Each many-to-many relationship type becomes a table; the **columns are the primary keys of the participating entity types**

Students table

StudentID	StudentFirstName	…
123456	Jamie	
123457	Stacey	

Classes table

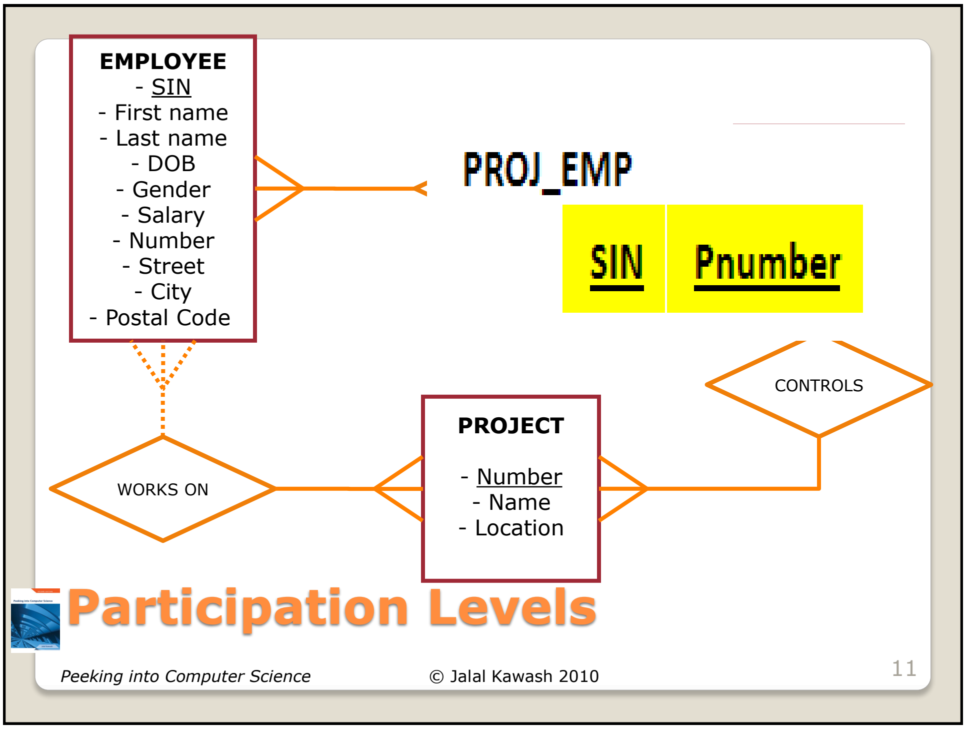
ClassName	ClassNumber	…
CPSC	203	
CPSC	231	

Registrations table (linking table)

StudentID	ClassName	Class-Number	Lecture No
123450	ENGL	201	01
123457	CPSC	203	01
123460	MATH	271	01

JT's Extra

Peeking into Computer Science © Jalal Kawash 2009 10



Participation Levels

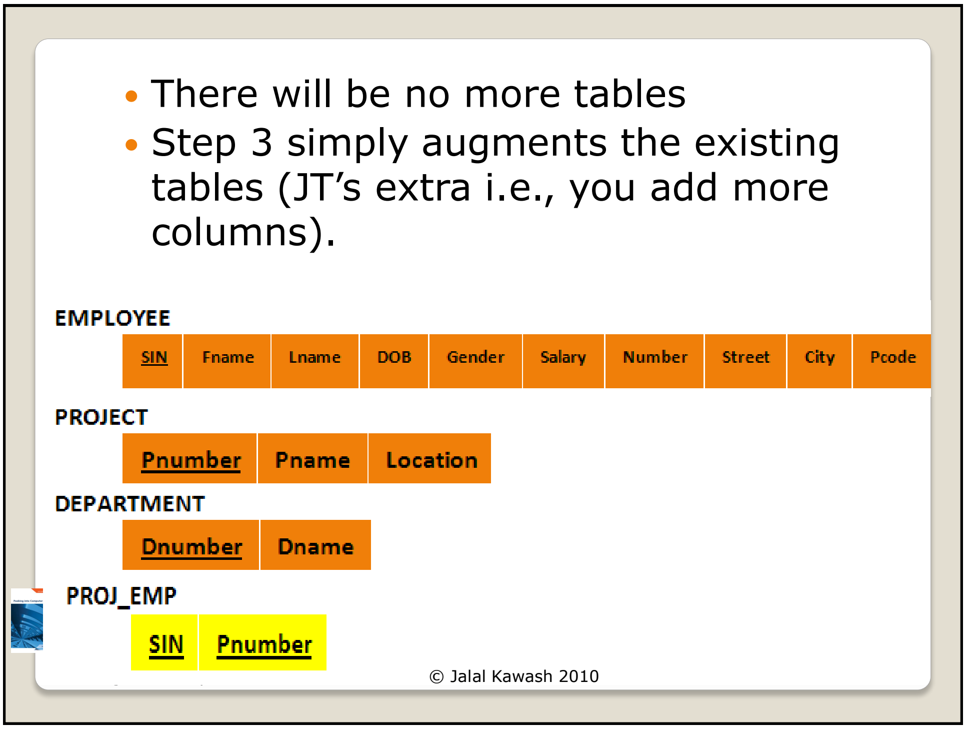
EMPLOYEE
- SIN
- First name
- Last name
- DOB
- Gender
- Salary
- Number
- Street
- City
- Postal Code

PROJ_EMP

SIN	Pnumber

PROJECT
- Number
- Name
- Location

Peeking into Computer Science © Jalal Kawash 2010 11



- There will be no more tables
- Step 3 simply augments the existing tables (JT's extra i.e., you add more columns).

EMPLOYEE

SIN	Fname	Lname	DOB	Gender	Salary	Number	Street	City	Pcode

PROJECT

Pnumber	Pname	Location

DEPARTMENT

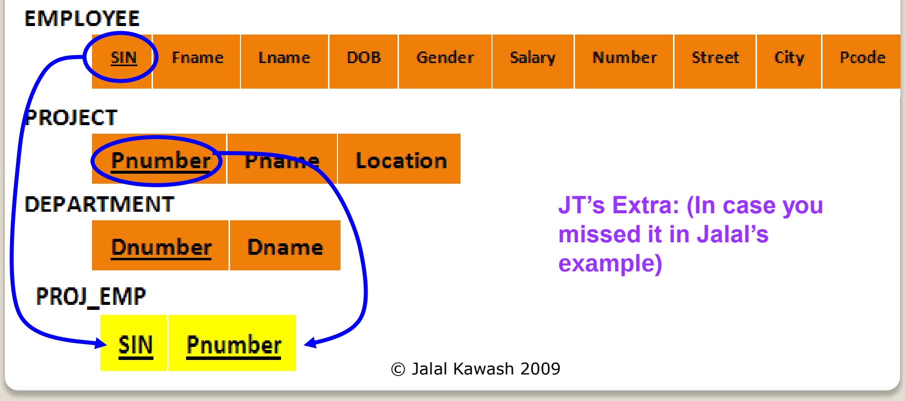
Dnumber	Dname

PROJ_EMP

SIN	Pnumber

© Jalal Kawash 2010

- There will be no more tables
- Step 3 simply augments the existing tables



EMPLOYEE

SIN	Fname	Lname	DOB	Gender	Salary	Number	Street	City	Pcode

PROJECT

Pnumber	Pname	Location

DEPARTMENT

Dnumber	Dname

PROJ_EMP

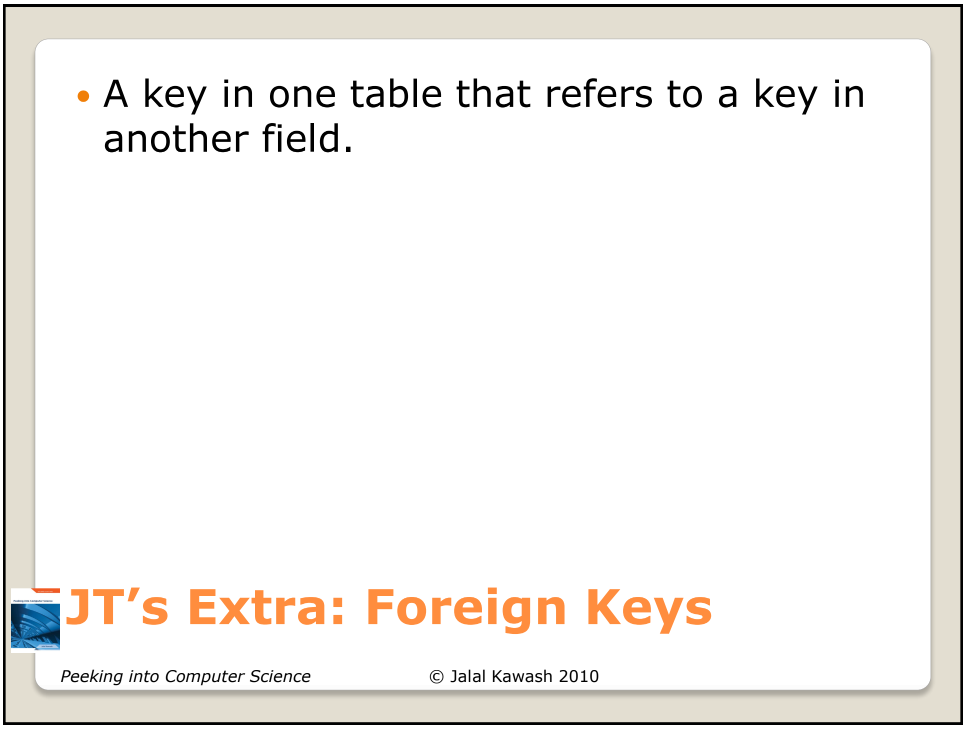
SIN	Pnumber

JT's Extra: (In case you missed it in Jalal's example)

© Jalal Kawash 2009

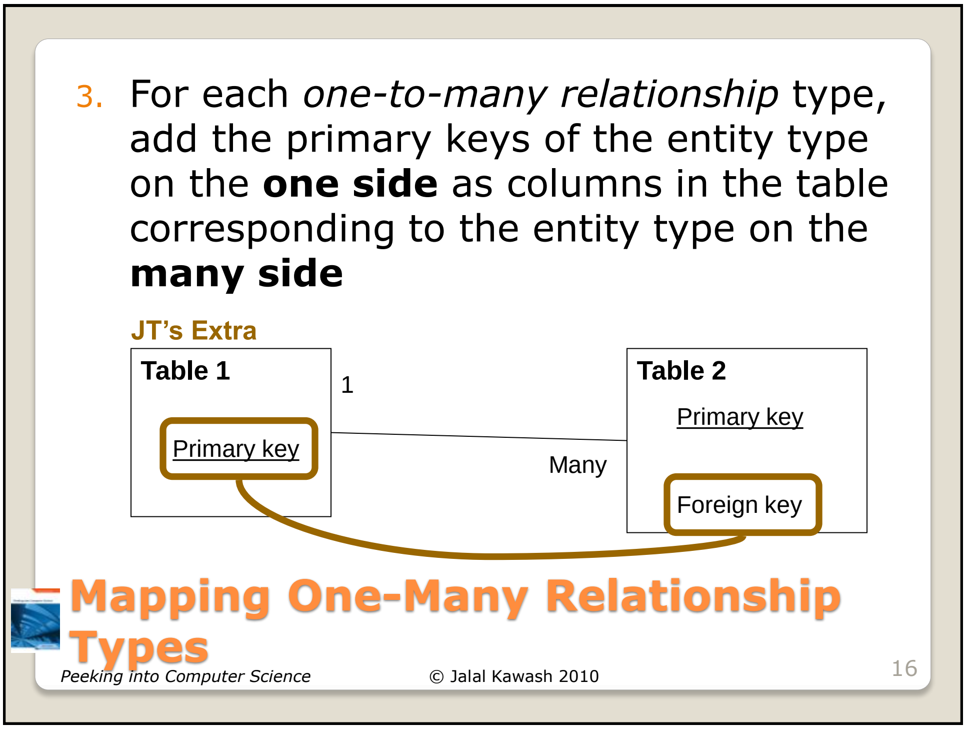
3. For each one-to-many relationship type, add the primary keys of the entity type on the **one side** as columns in the table corresponding to the entity type on the **many side**

Mapping One-Many Relationship Types



JT’s Extra: Foreign Keys

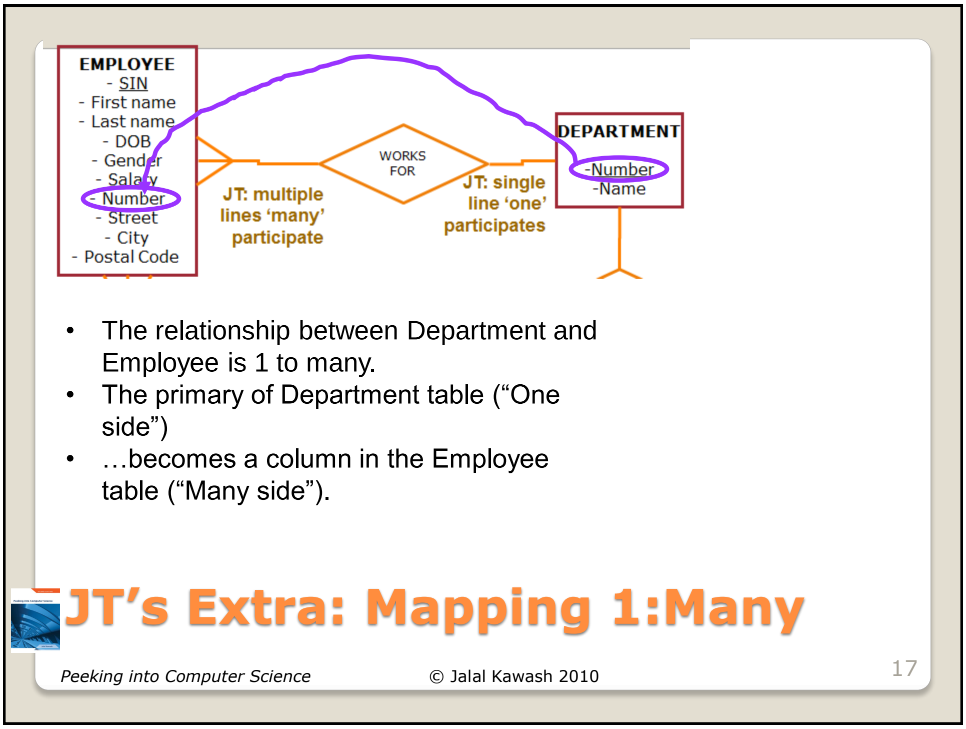
- A key in one table that refers to a key in another field.



Mapping One-Many Relationship Types

3. For each *one-to-many relationship* type, add the primary keys of the entity type on the one side as columns in the table corresponding to the entity type on the **many side**

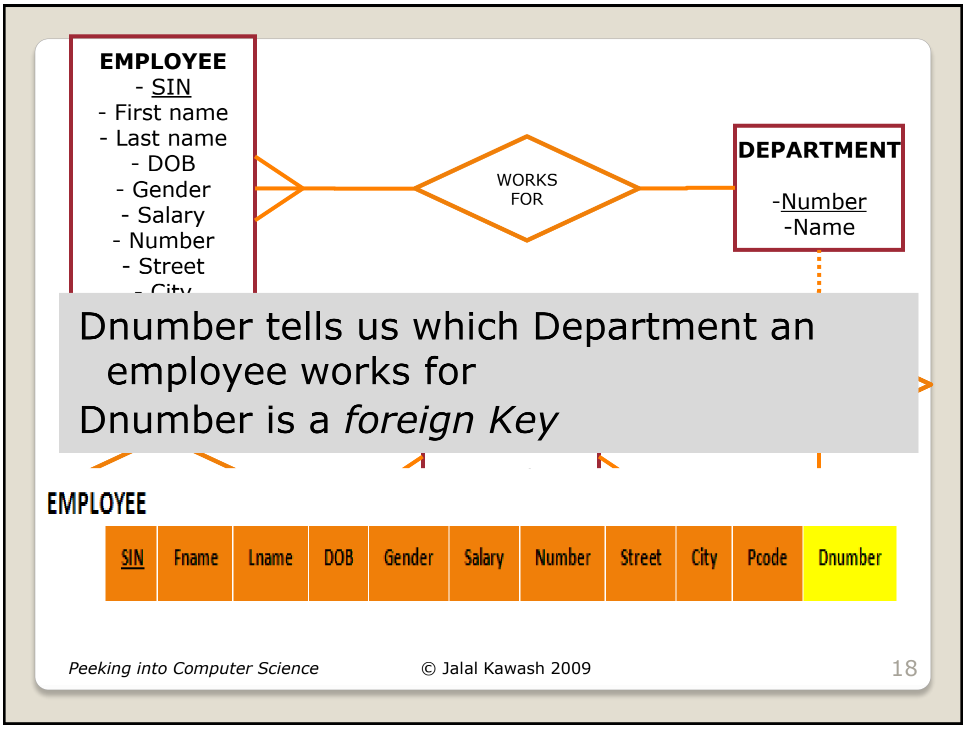
JT’s Extra



- The relationship between Department and Employee is 1 to many.
- The primary of Department table ("One side")
- …becomes a column in the Employee table ("Many side").

JT's Extra: Mapping 1:Many

Peeking into Computer Science © Jalal Kawash 2010 17



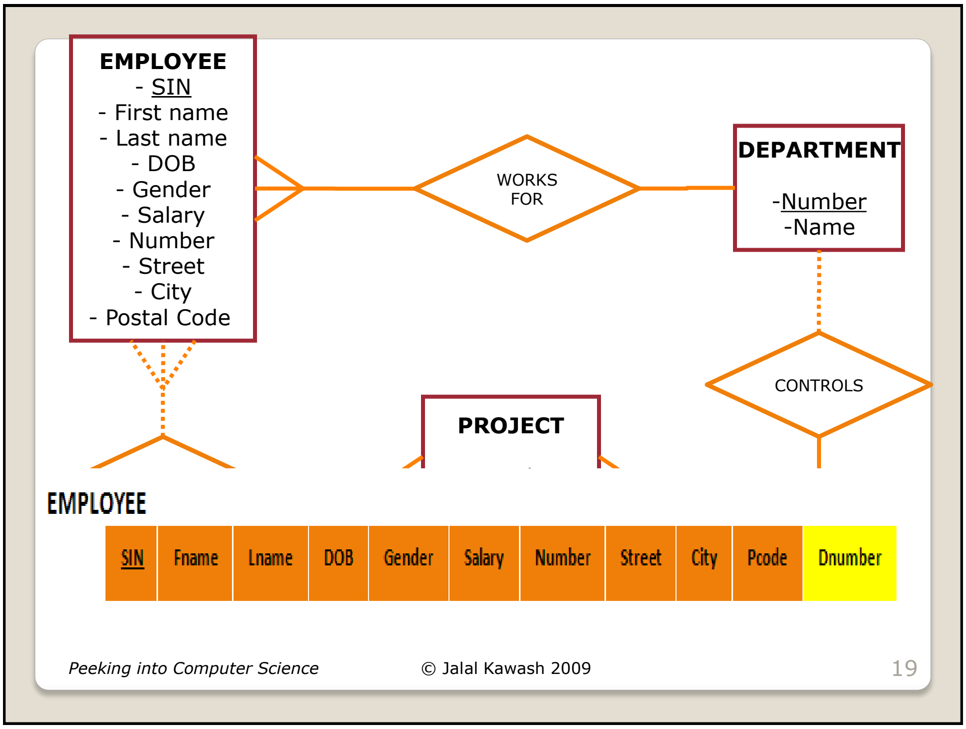
Dnumber tells us which Department an employee works for

Dnumber is a *foreign Key*

EMPLOYEE

SIN	Fname	Lname	DOB	Gender	Salary	Number	Street	City	Pcode	Dnumber

Peeking into Computer Science © Jalal Kawash 2009 18



EMPLOYEE
- <u>SIN</u>
- First name
- Last name
- DOB
- Gender
- Salary
- Number
- Street
- City
- Postal Code

WORKS FOR

DEPARTMENT

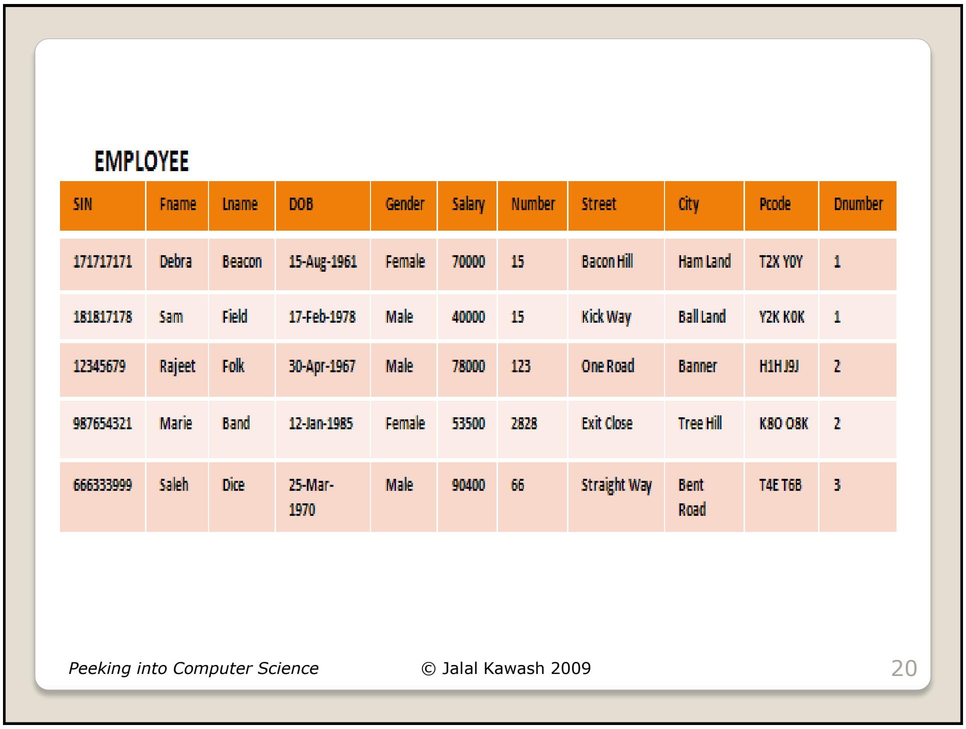
-<u>Number</u>
-Name

CONTROLS

PROJECT

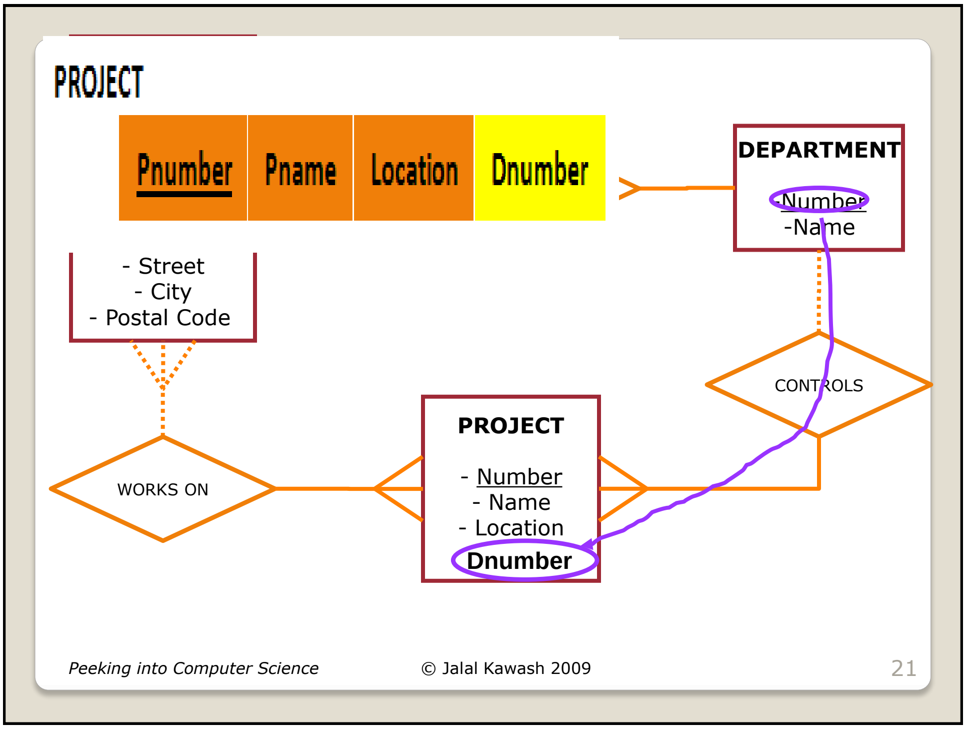
EMPLOYEE

<u>SIN</u>	Fname	Lname	DOB	Gender	Salary	Number	Street	City	Pcode	Dnumber



EMPLOYEE

SIN	Fname	Lname	DOB	Gender	Salary	Number	Street	City	Pcode	Dnumber
171717171	Debra	Beacon	15-Aug-1961	Female	70000	15	Bacon Hill	Ham Land	T2X Y0Y	1
181817178	Sam	Field	17-Feb-1978	Male	40000	15	Kick Way	Ball Land	Y2K K0K	1
12345679	Rajeet	Folk	30-Apr-1967	Male	78000	123	One Road	Banner	H1H J9J	2
987654321	Marie	Band	12-Jan-1985	Female	53500	2828	Exit Close	Tree Hill	K8O O8K	2
666333999	Saleh	Dice	25-Mar-1970	Male	90400	66	Straight Way	Bent Road	T4E T6B	3



PROJECT

Pnumber	Pname	Location	Dnumber

DEPARTMENT

-Number

-Name

- Street
- City
- Postal Code

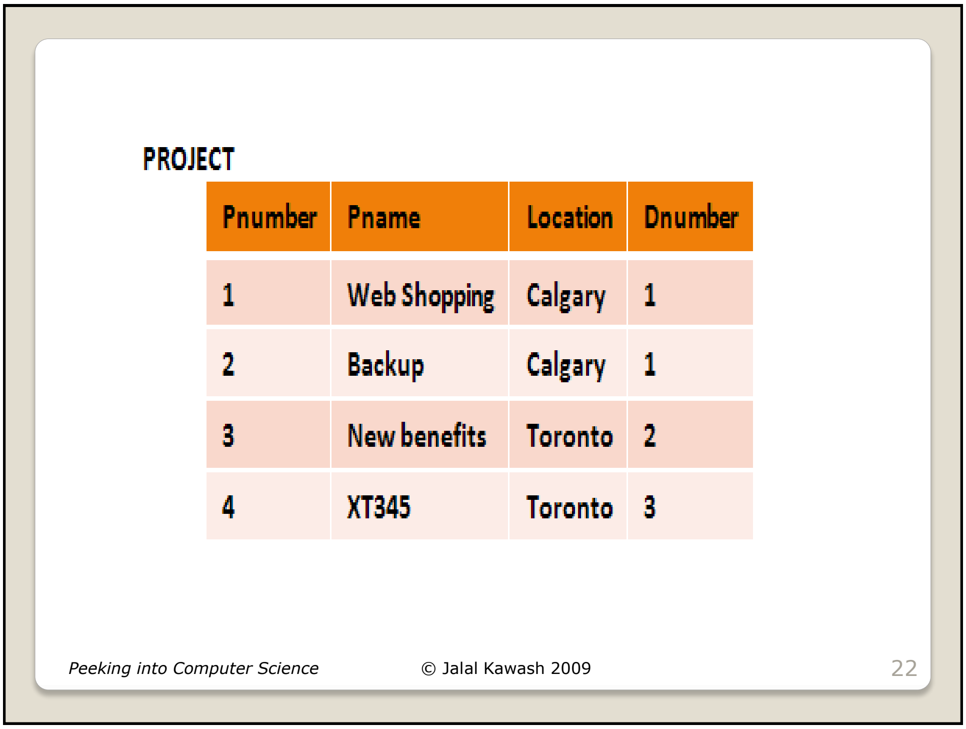
CONTROLS

WORKS ON

PROJECT

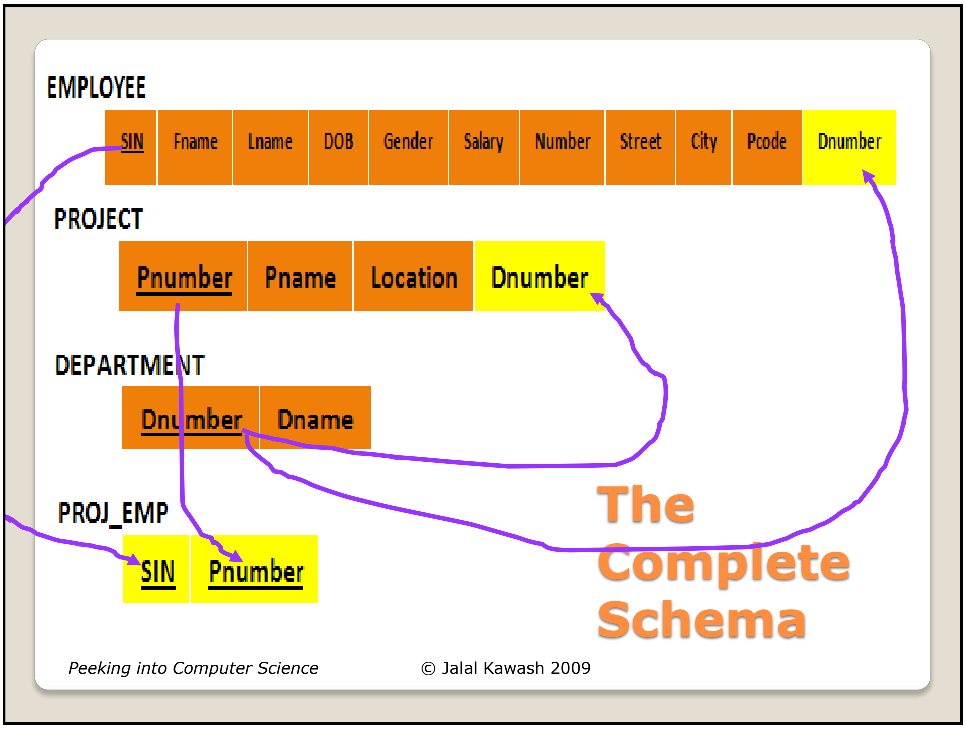
- Number
- Name
- Location

Dnumber



PROJECT

Pnumber	Pname	Location	Dnumber
1	Web Shopping	Calgary	1
2	Backup	Calgary	1
3	New benefits	Toronto	2
4	XT345	Toronto	3



The Complete Schema

EMPLOYEE

SIN	Fname	Lname	DOB	Gender	Salary	Number	Street	City	Pcode	Dnumber

PROJECT

Pnumber	Pname	Location	Dnumber

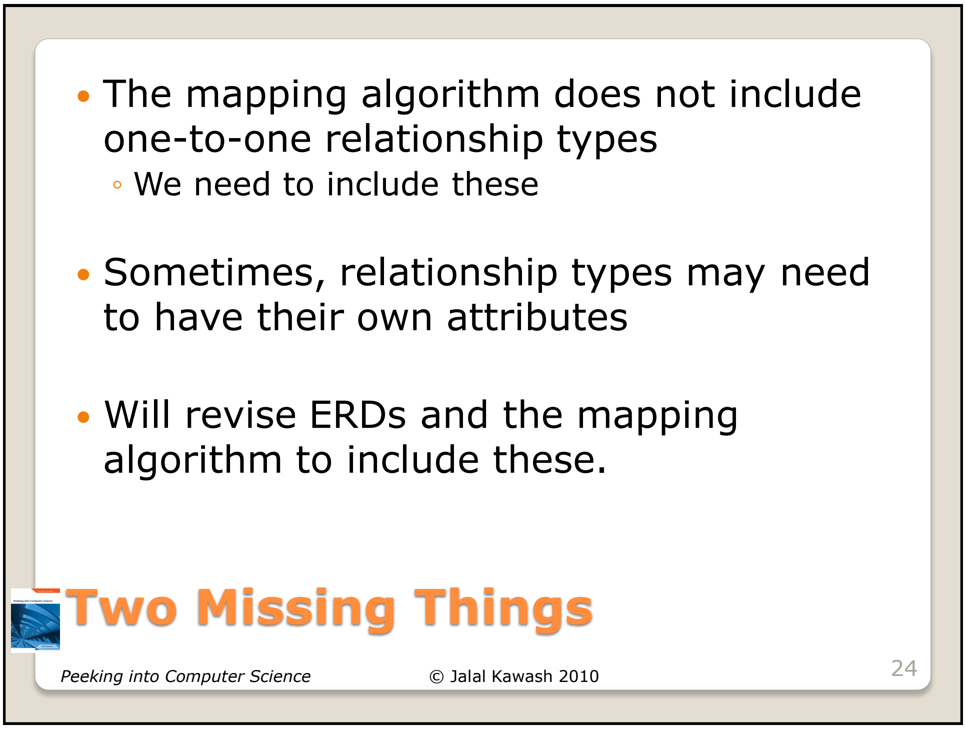
DEPARTMENT

Dnumber	Dname

PROJ_EMP

SIN	Pnumber

Peeking into Computer Science © Jalal Kawash 2009



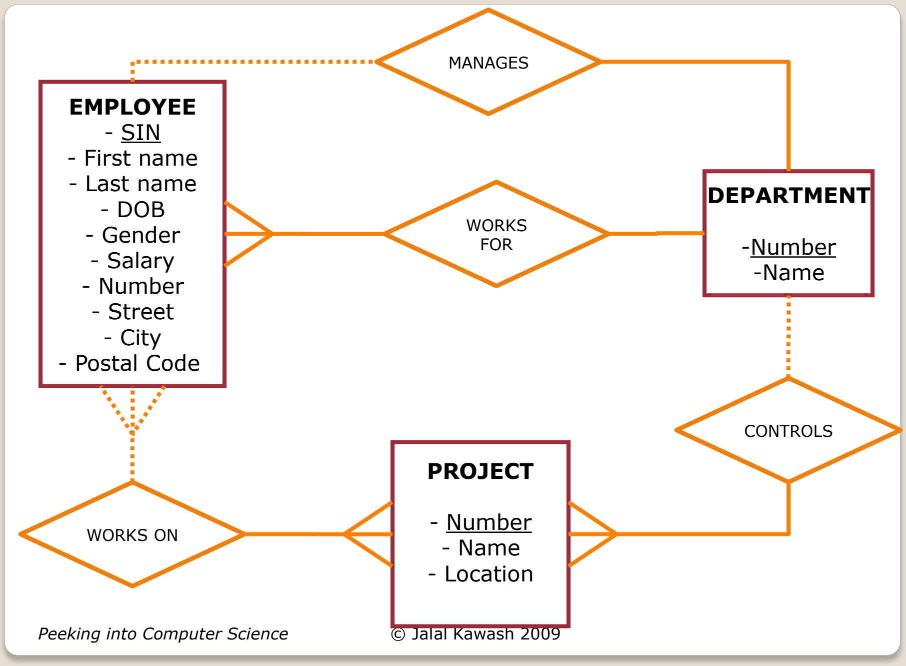
Two Missing Things

- The mapping algorithm does not include one-to-one relationship types
  - We need to include these
- Sometimes, relationship types may need to have their own attributes
- Will revise ERDs and the mapping algorithm to include these.

Peeking into Computer Science © Jalal Kawash 2010

MANAGES Relationship Type

- A department is ***managed*** by one employee
- Each department has a manager which is also an employee
- EMPLOYEE and DEPARTMENT are related by the MANAGES relationship type
- Each department can have only one manager, and each employee can manage at most one department. This is a one-to-one relationship



MANAGES

EMPLOYEE
- SIN
- First name
- Last name
- DOB
- Gender
- Salary
- Number
- Street
- City
- Postal Code

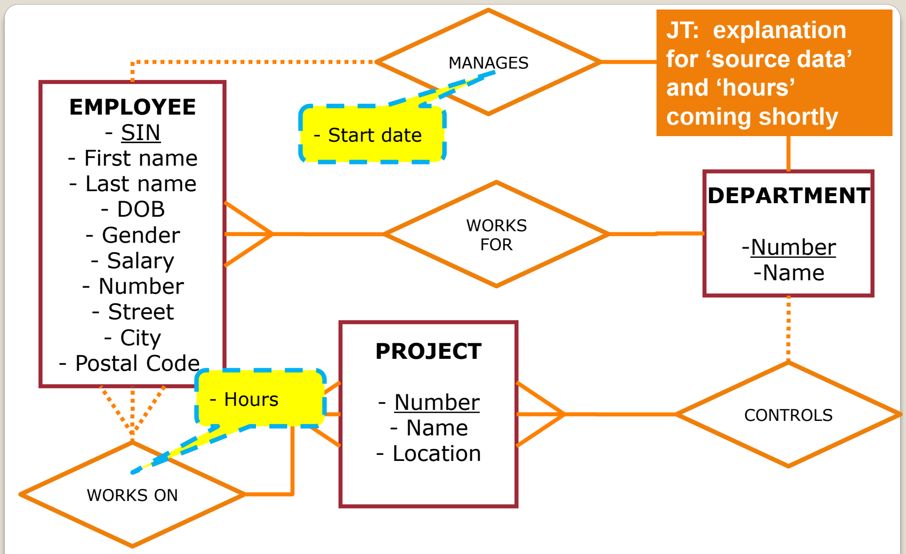
WORKS FOR

DEPARTMENT
-Number
-Name

CONTROLS

PROJECT
- Number
- Name
- Location

WORKS ON



MANAGES

- Start date

JT: explanation for ‘source data’ and ‘hours’ coming shortly

EMPLOYEE
- SIN
- First name
- Last name
- DOB
- Gender
- Salary
- Number
- Street
- City
- Postal Code

WORKS FOR

DEPARTMENT
-Number
-Name

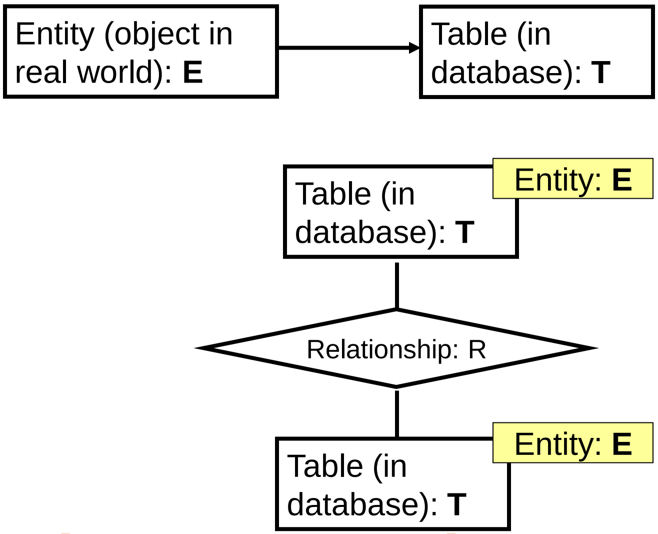
PROJECT
- Number
- Name
- Location

- Hours

CONTROLS

WORKS ON

Attributes for Relationship types



Entity (object in real world): **E**

Table (in database): **T**

Table (in database): **T**

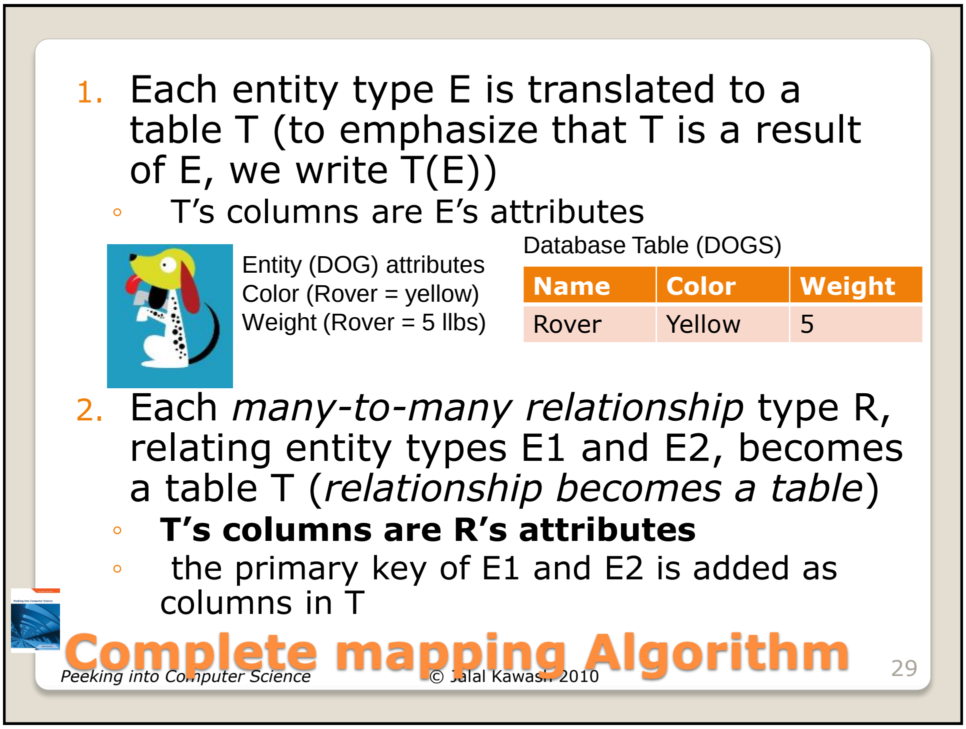
Entity: **E**

Relationship: R

Table (in database): **T**

Entity: **E**

JT’s Extra: Notation Reminder



Complete mapping Algorithm

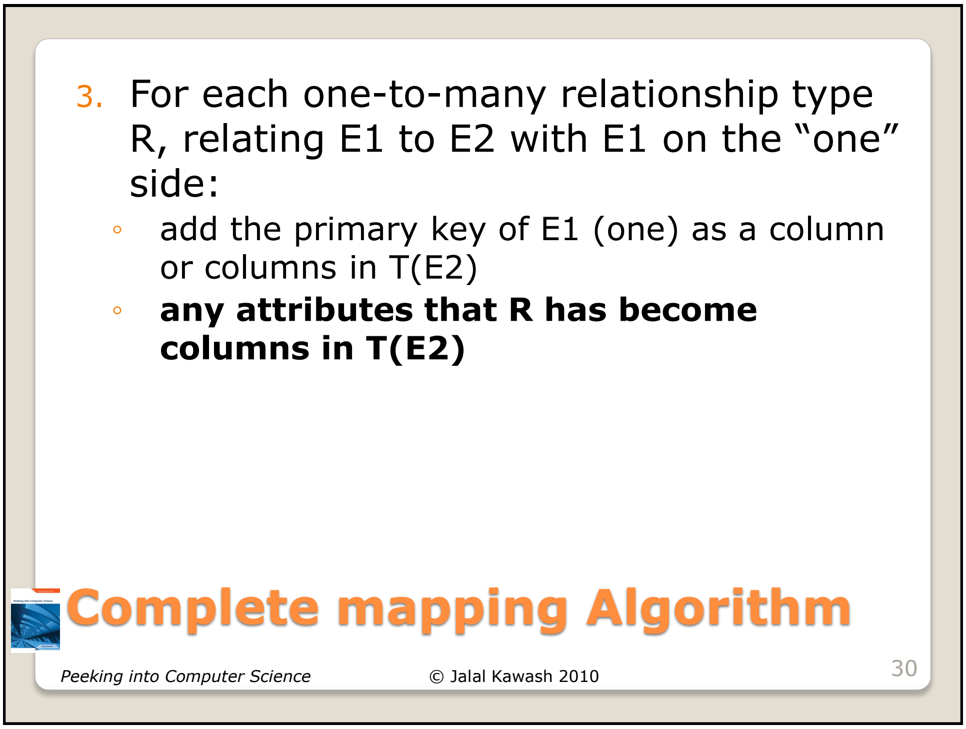
1. Each entity type E is translated to a table T (to emphasize that T is a result of E, we write T(E))
   - T's columns are E's attributes

Entity (DOG) attributes
Color (Rover = yellow)
Weight (Rover = 5 llbs)

Database Table (DOGS)

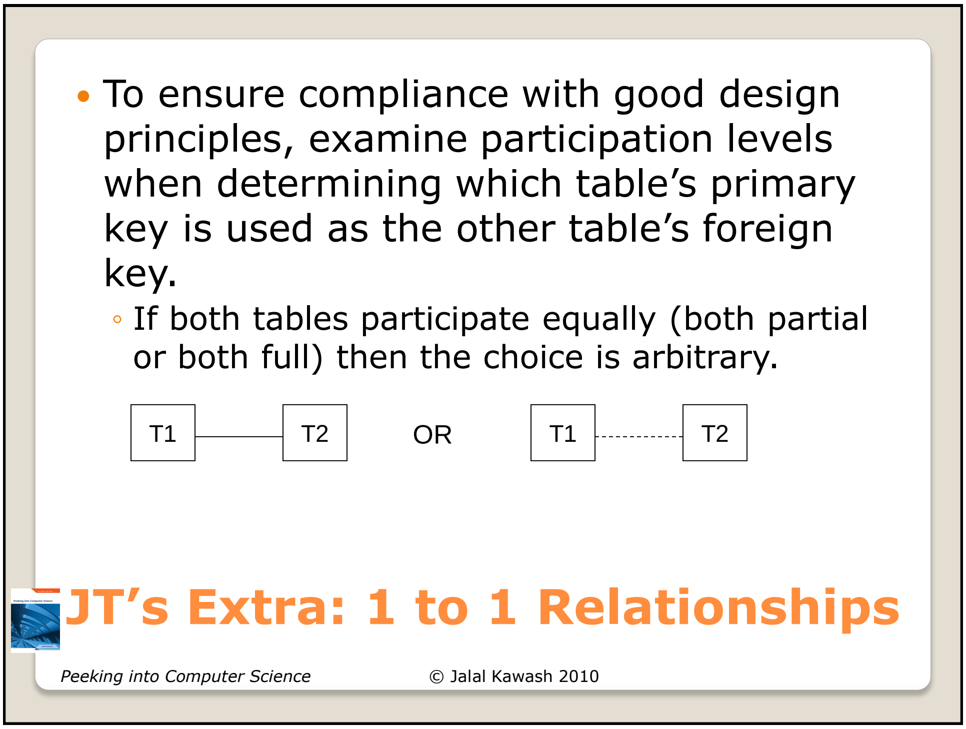
Name	Color	Weight
Rover	Yellow	5

2. Each *many-to-many relationship* type R, relating entity types E1 and E2, becomes a table T (*relationship becomes a table*)
   - **T's columns are R's attributes**
   - the primary key of E1 and E2 is added as columns in T



Complete mapping Algorithm

3. For each one-to-many relationship type R, relating E1 to E2 with E1 on the "one" side:
   - add the primary key of E1 (one) as a column or columns in T(E2)
   - **any attributes that R has become columns in T(E2)**

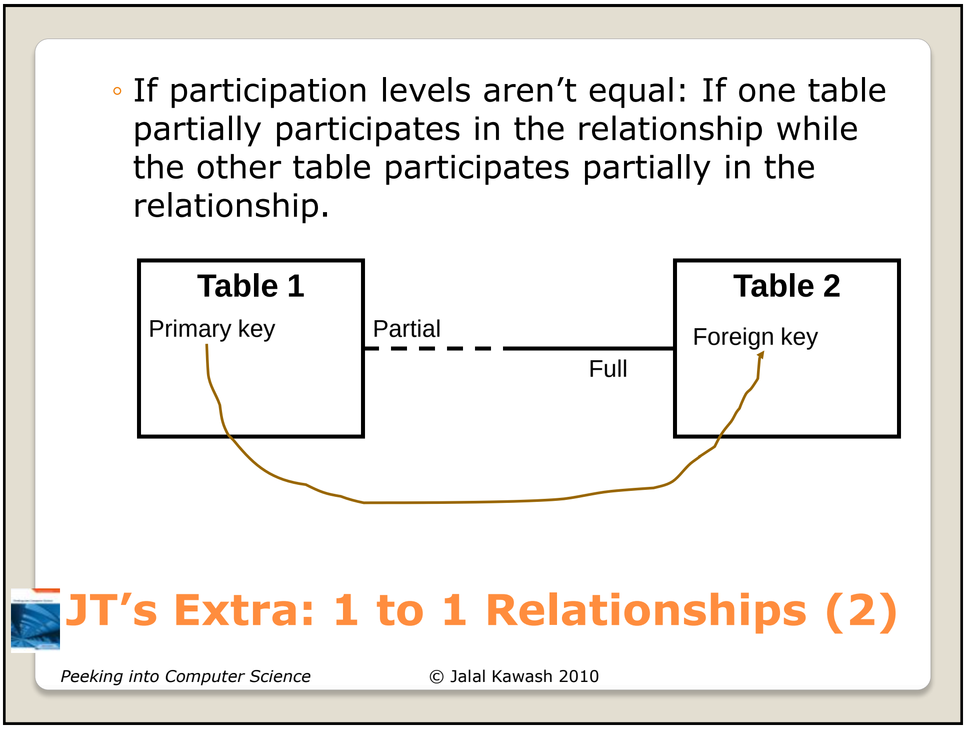


JT’s Extra: 1 to 1 Relationships

- To ensure compliance with good design principles, examine participation levels when determining which table’s primary key is used as the other table’s foreign key.
  - If both tables participate equally (both partial or both full) then the choice is arbitrary.

T1 — T2 OR T1 ---- T2

Peeking into Computer Science © Jalal Kawash 2010



JT’s Extra: 1 to 1 Relationships (2)

- If participation levels aren’t equal: If one table partially participates in the relationship while the other table participates partially in the relationship.

Table 1

Primary key

Partial

Full

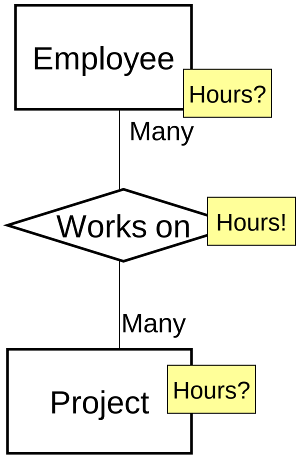
Table 2

Foreign key

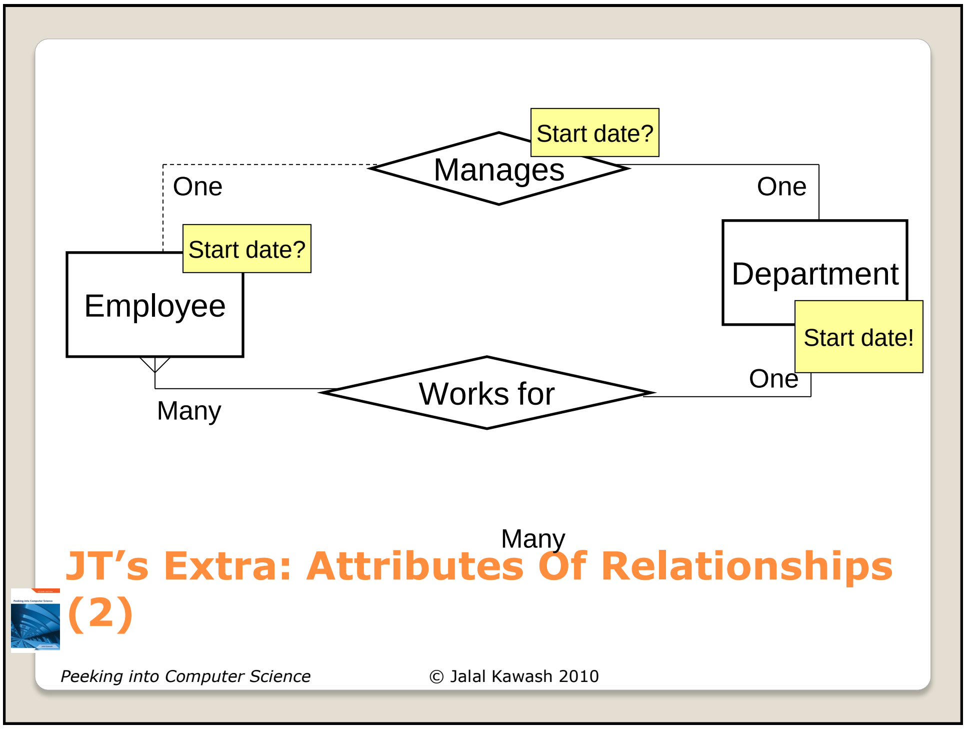
Peeking into Computer Science © Jalal Kawash 2010

4. For each one-to-one relationship type R, relating E1 to E2 with E1 on a partial participation side or both E1 and E2 fully participate in R:
   - add the primary key of E1 (JT: partial) as a column or columns in T(E2, JT: full)
   - any attributes that R has become columns in T(E2)

Complete mapping Algorithm

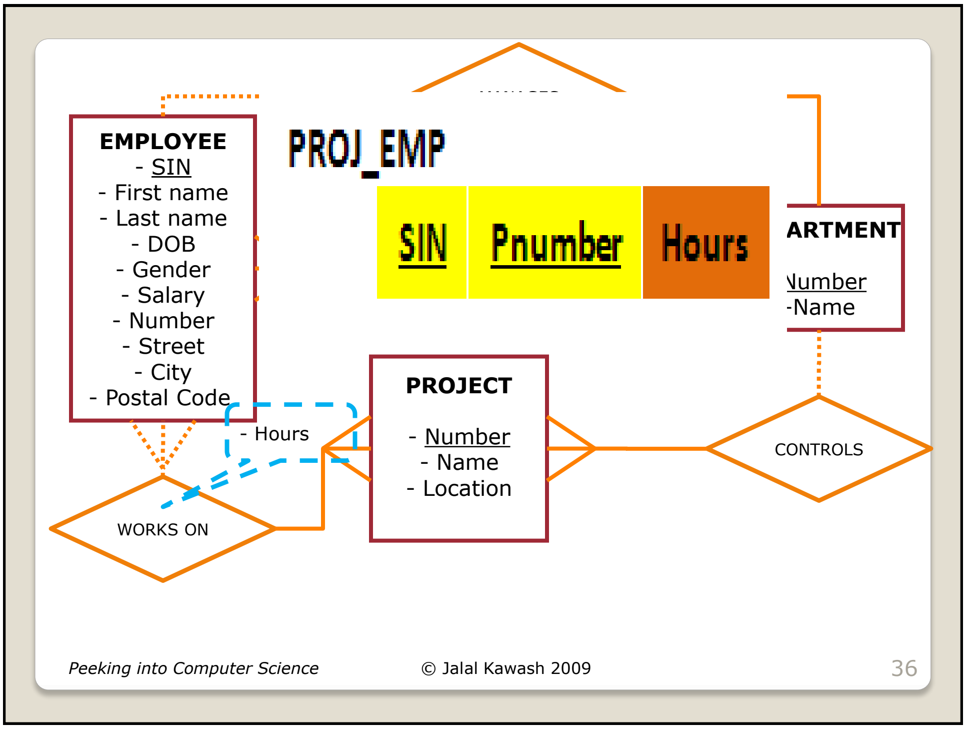


JT's Extra: Attributes Of Relationships



JT's Extra: Attributes Of Relationships (2)

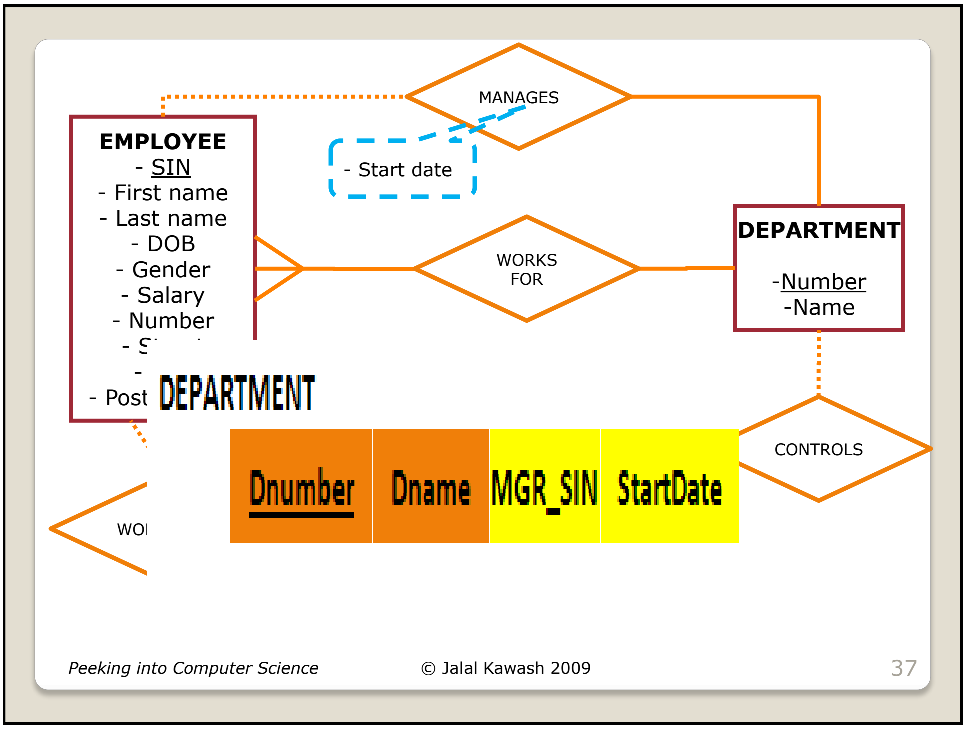
Peeking into Computer Science © Jalal Kawash 2010



PROJ_EMP

SIN	Pnumber	Hours

Peeking into Computer Science © Jalal Kawash 2009 36



EMPLOYEE
- SIN
- First name
- Last name
- DOB
- Gender
- Salary
- Number
- S
-
- Post

MANAGES

- Start date

WORKS FOR

DEPARTMENT

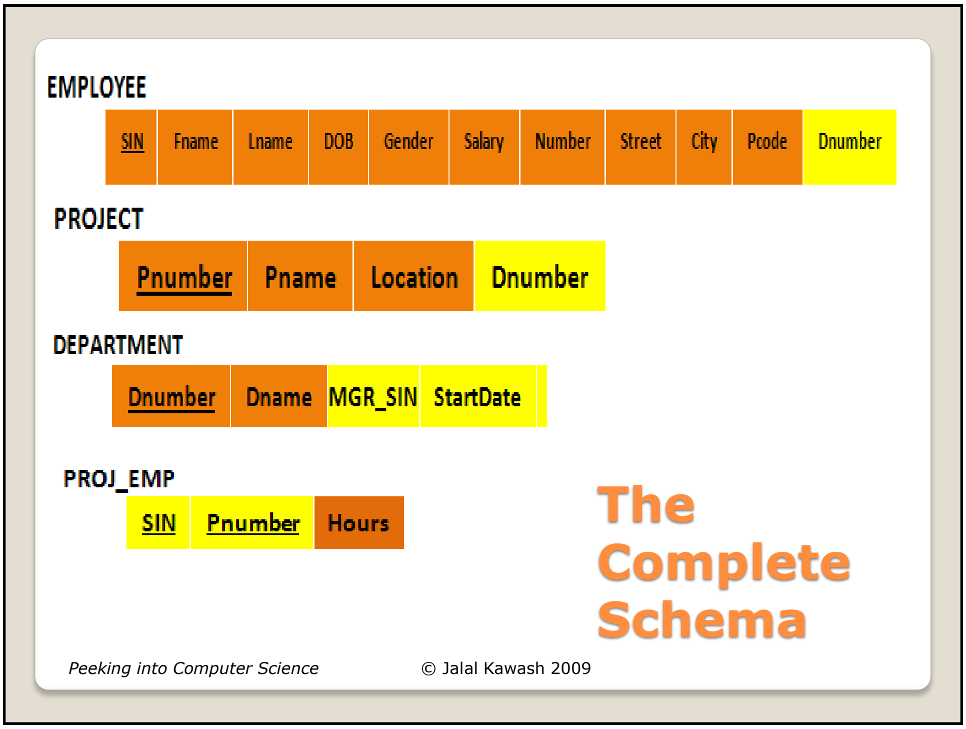
-Number
-Name

CONTROLS

WO

DEPARTMENT

Dnumber	Dname	MGR_SIN	StartDate



EMPLOYEE

SIN	Fname	Lname	DOB	Gender	Salary	Number	Street	City	Pcode	Dnumber

PROJECT

Pnumber	Pname	Location	Dnumber

DEPARTMENT

Dnumber	Dname	MGR_SIN	StartDate

PROJ_EMP

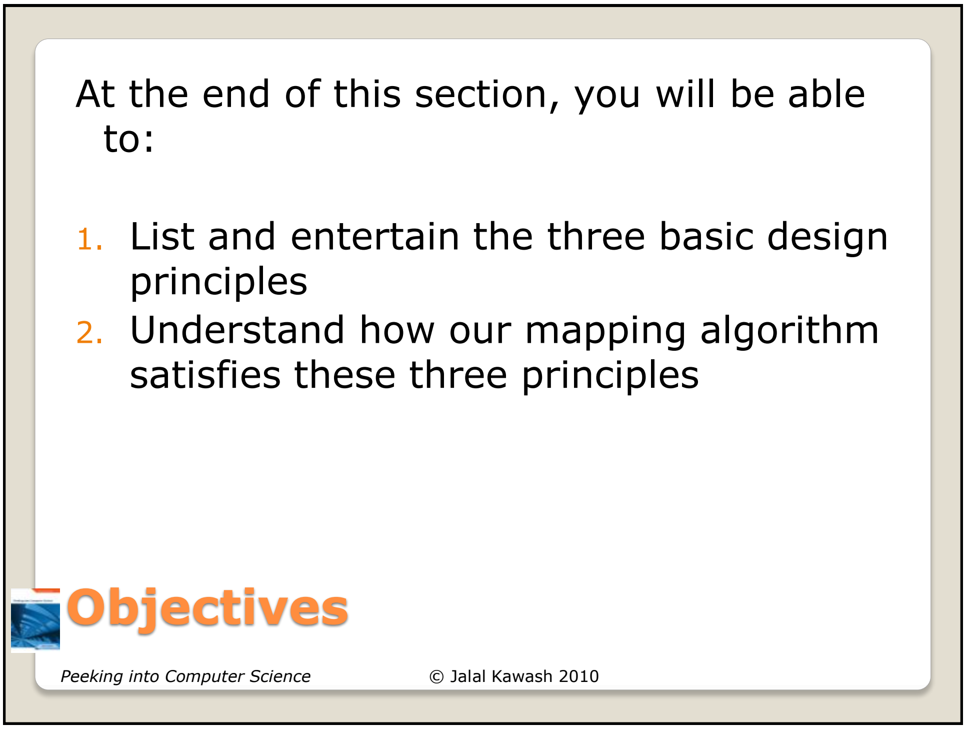
SIN	Pnumber	Hours

The Complete Schema

Design Principles

What makes a Good Design?

39



At the end of this section, you will be able to:

1. List and entertain the three basic design principles
2. Understand how our mapping algorithm satisfies these three principles

Objectives

Peeking into Computer Science © Jalal Kawash 2010

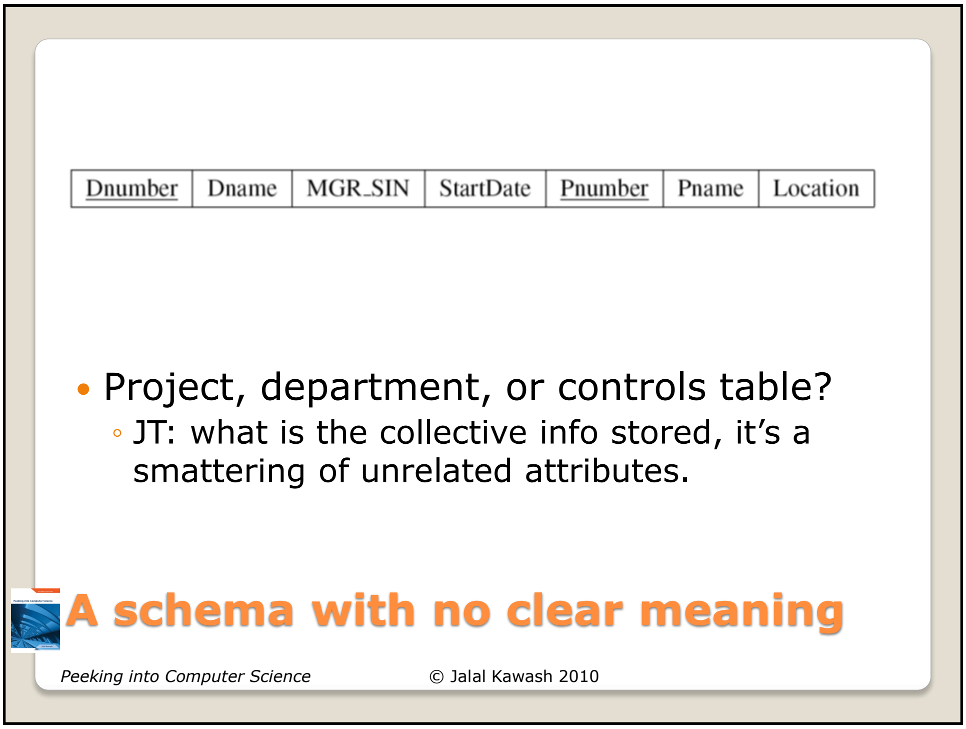
Basic Design Principles

1. Meaning of a Schema should be easily explained
2. Reduce Redundancy
3. Reduce NULL values

Design Principle (1)

1. Design a schema so that its meaning can be easily explained

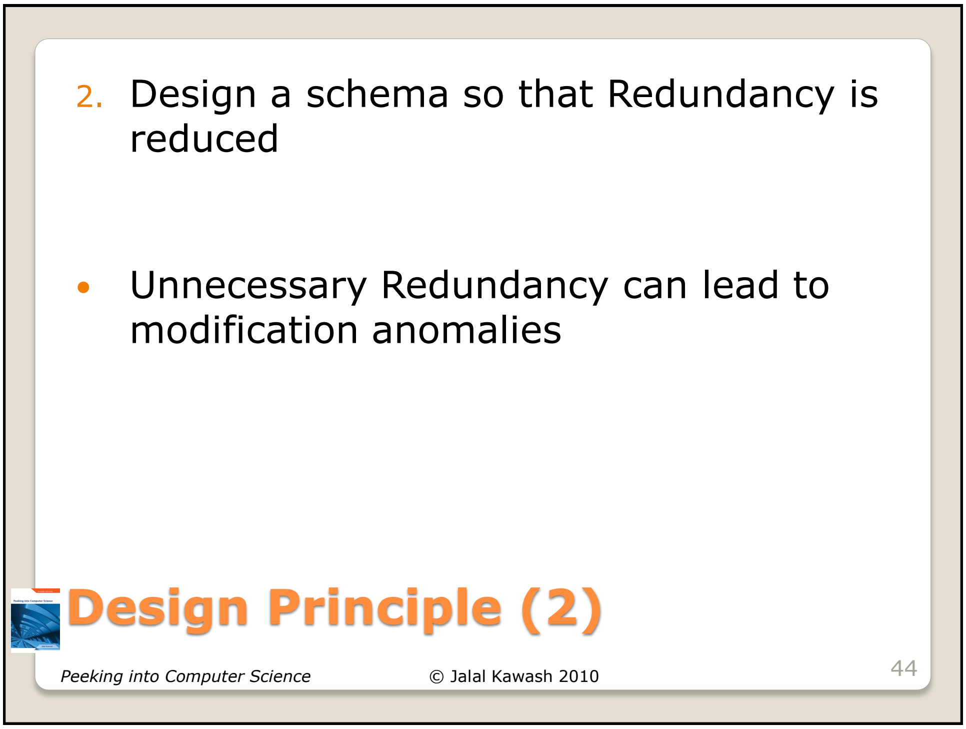
- Do NOT combine attributes from different entity types into a single table



A schema with no clear meaning

Dnumber	Dname	MGR_SIN	StartDate	Pnumber	Pname	Location

- Project, department, or controls table?
  - JT: what is the collective info stored, it’s a smattering of unrelated attributes.

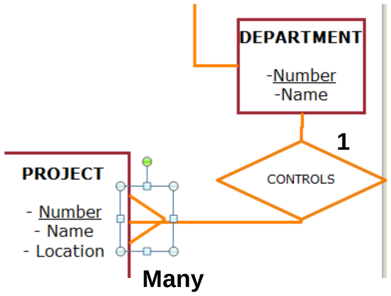


Design Principle (2)

2. Design a schema so that Redundancy is reduced

- Unnecessary Redundancy can lead to modification anomalies

JT: Recall Department And Project



With 1:Many relationship the primary key of the '1' became foreign key on the 'many' side

PROJECT

Pnumber	Pname	Location	Dnumber

DEPARTMENT

Dnumber	Dname

Unnecessary Redundancy

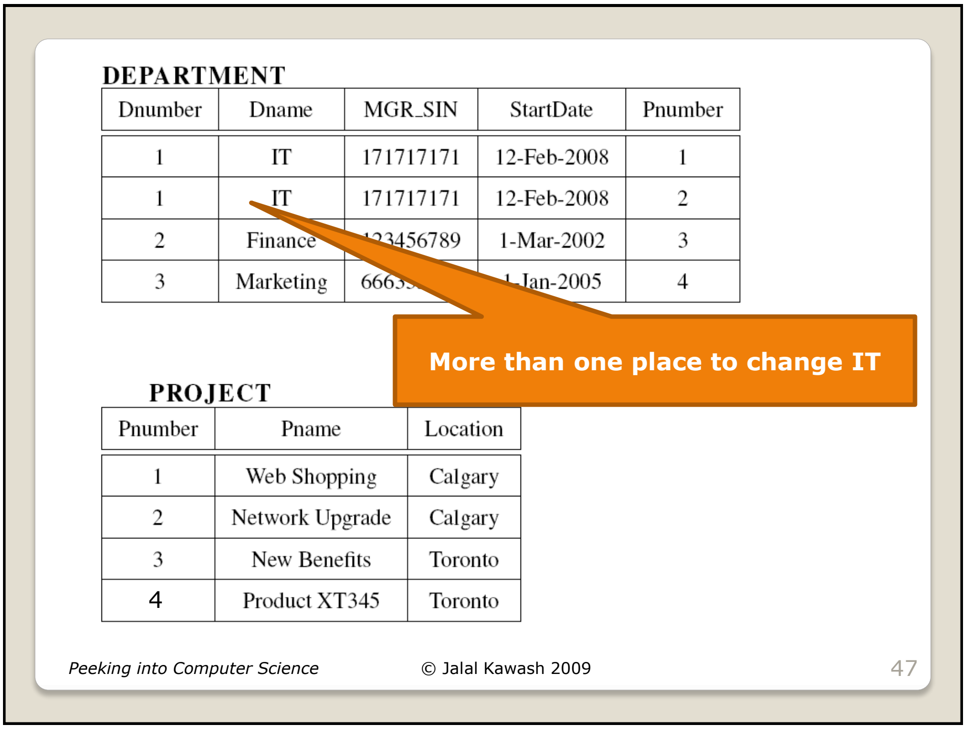
- Assume the schema definition for Project and Department
  - migrating Pnumber to DEPARTMENT

PROJECT

Pnumber	Pname	Location

DEPARTMENT

Dnumber	Dname	MGR_SIN	StartDate	Pnumber



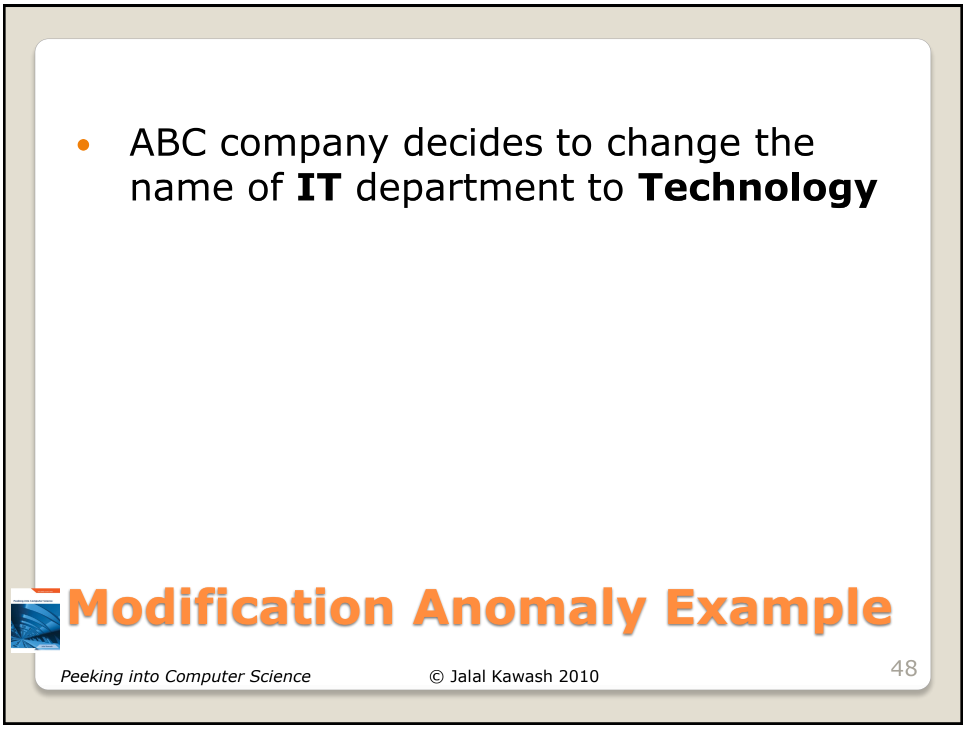
DEPARTMENT

Dnumber	Dname	MGR_SIN	StartDate	Pnumber
1	IT	171717171	12-Feb-2008	1
1	IT	171717171	12-Feb-2008	2
2	Finance	123456789	1-Mar-2002	3
3	Marketing	6663	1-Jan-2005	4

More than one place to change IT

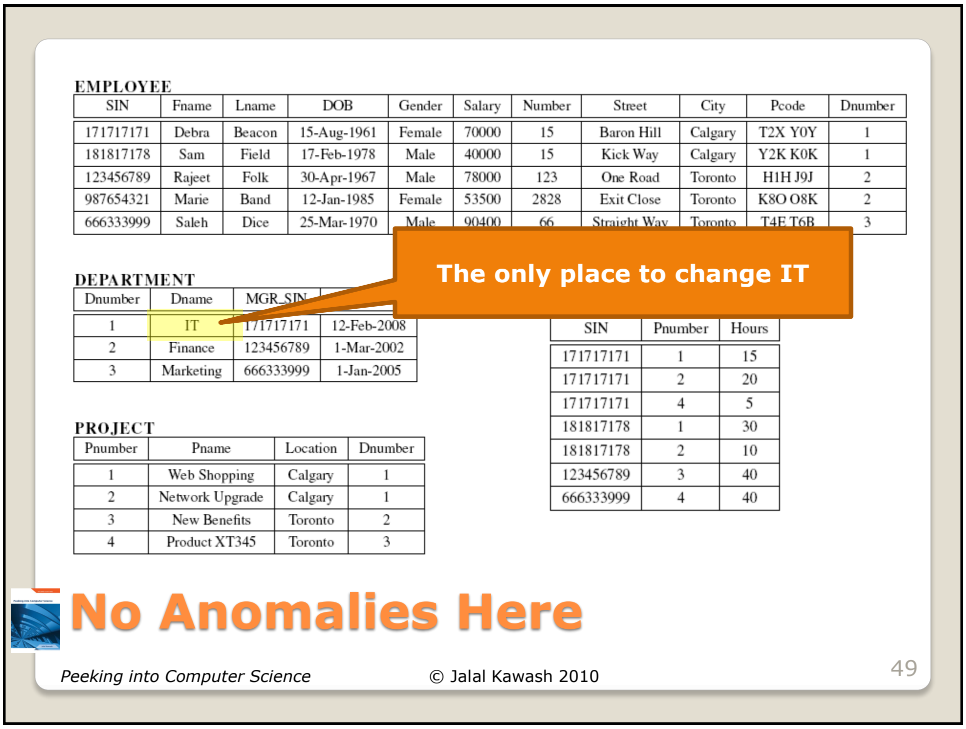
PROJECT

Pnumber	Pname	Location
1	Web Shopping	Calgary
2	Network Upgrade	Calgary
3	New Benefits	Toronto
4	Product XT345	Toronto



- ABC company decides to change the name of **IT** department to **Technology**

Modification Anomaly Example



EMPLOYEE

SIN	Fname	Lname	DOB	Gender	Salary	Number	Street	City	Pcode	Dnumber
171717171	Debra	Beacon	15-Aug-1961	Female	70000	15	Baron Hill	Calgary	T2X Y0Y	1
181817178	Sam	Field	17-Feb-1978	Male	40000	15	Kick Way	Calgary	Y2K K0K	1
123456789	Rajeet	Folk	30-Apr-1967	Male	78000	123	One Road	Toronto	H1H J9J	2
987654321	Marie	Band	12-Jan-1985	Female	53500	2828	Exit Close	Toronto	K8O O8K	2
666333999	Saleh	Dice	25-Mar-1970	Male	90400	66	Straight Way	Toronto	T4E T6B	3

DEPARTMENT

Dnumber	Dname	MGR_SIN	
1	IT	171717171	12-Feb-2008
2	Finance	123456789	1-Mar-2002
3	Marketing	666333999	1-Jan-2005

The only place to change IT

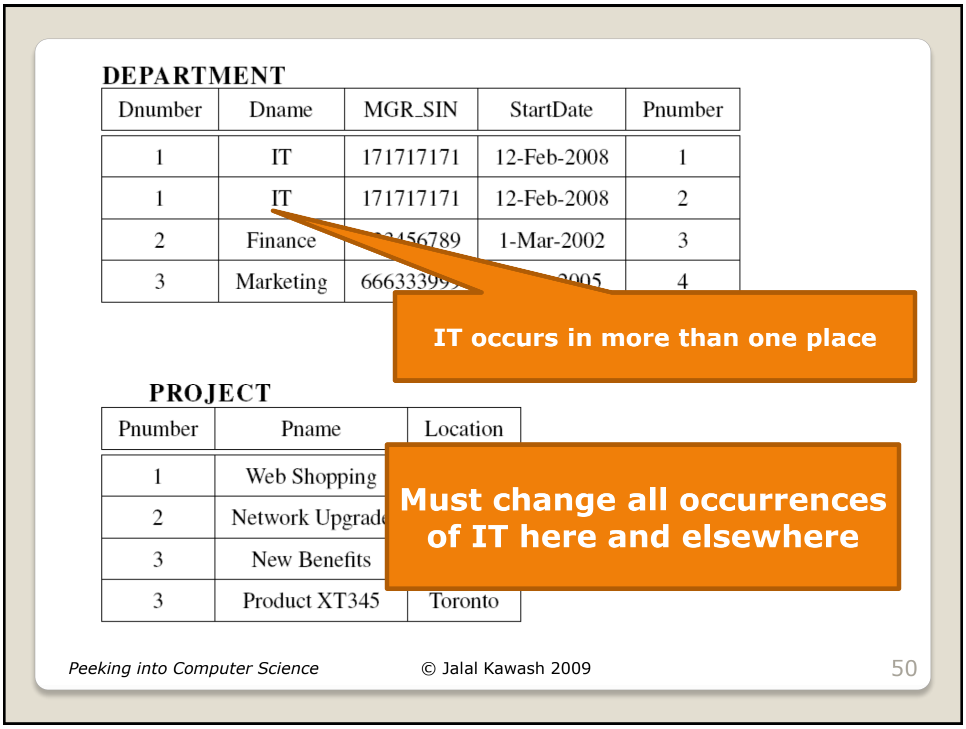
SIN	Pnumber	Hours
171717171	1	15
171717171	2	20
171717171	4	5
181817178	1	30
181817178	2	10
123456789	3	40
666333999	4	40

PROJECT

Pnumber	Pname	Location	Dnumber
1	Web Shopping	Calgary	1
2	Network Upgrade	Calgary	1
3	New Benefits	Toronto	2
4	Product XT345	Toronto	3

No Anomalies Here

Peeking into Computer Science © Jalal Kawash 2010 49



DEPARTMENT

Dnumber	Dname	MGR_SIN	StartDate	Pnumber
1	IT	171717171	12-Feb-2008	1
1	IT	171717171	12-Feb-2008	2
2	Finance	123456789	1-Mar-2002	3
3	Marketing	666333999	1-Jan-2005	4

IT occurs in more than one place

PROJECT

Pnumber	Pname	Location
1	Web Shopping	
2	Network Upgrade	
3	New Benefits	
3	Product XT345	Toronto

Must change all occurrences of IT here and elsewhere

Peeking into Computer Science © Jalal Kawash 2009 50

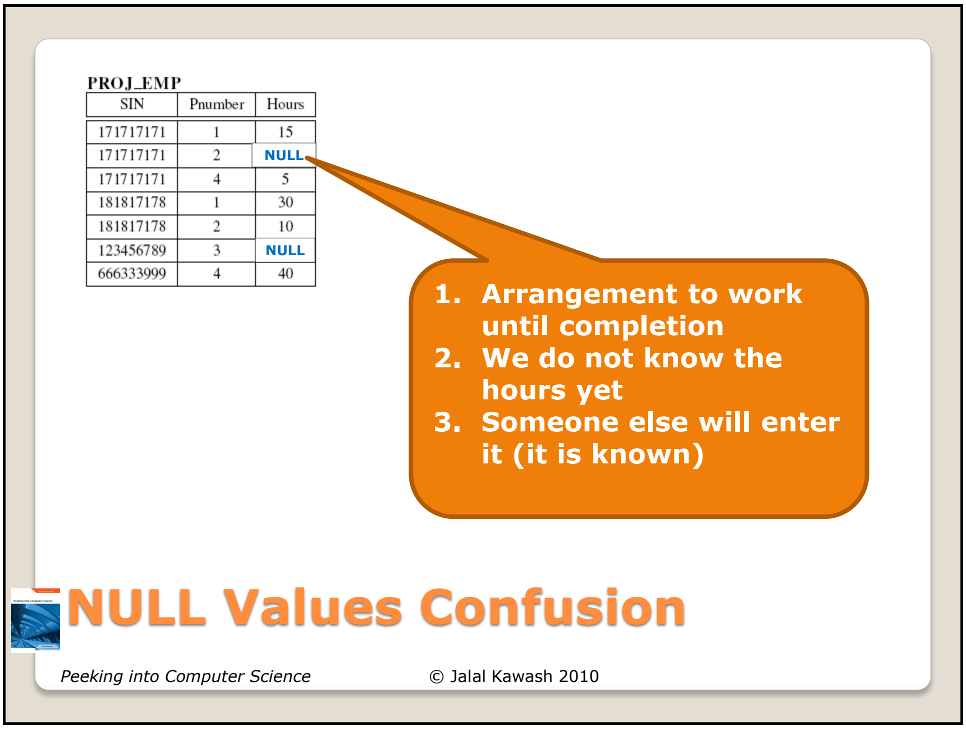
JT’s Extra: Null Values

- Refers to empty fields of a record.
- Primary keys cannot be null but other fields may be null.

Design Principle (3)

3. Design a schema so that NULL values are minimized as much as possible

- Waste space
- Result in confusion:
  - A NULL value could mean:
    - Does not apply
    - Unknown
    - To be recorded

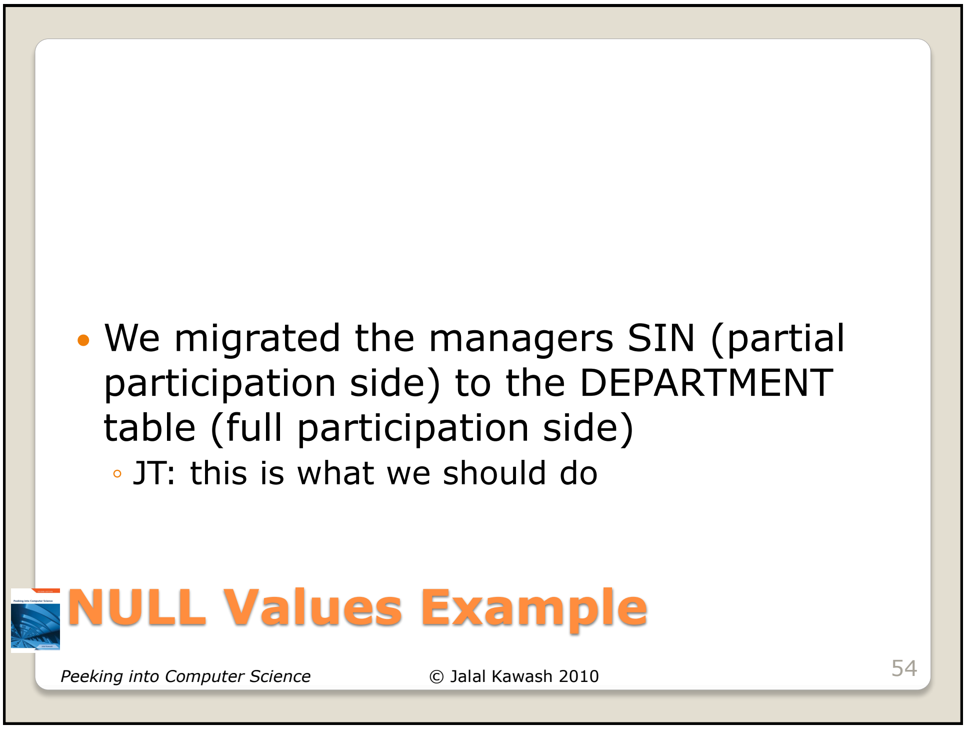


NULL Values Confusion

PROJ_EMP

SIN	Pnumber	Hours
171717171	1	15
171717171	2	NULL
171717171	4	5
181817178	1	30
181817178	2	10
123456789	3	NULL
666333999	4	40

1. Arrangement to work until completion
2. We do not know the hours yet
3. Someone else will enter it (it is known)



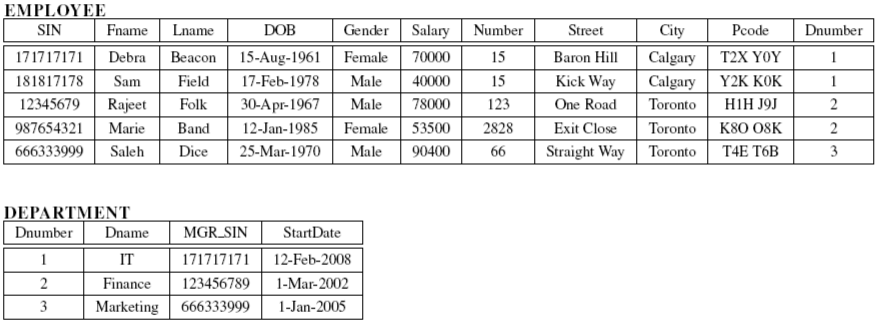
NULL Values Example

- We migrated the managers SIN (partial participation side) to the DEPARTMENT table (full participation side)
  - JT: this is what we should do

NULL Values Example

- What if we migrated the department Number to the EMPLOYEE table?
  - JT: again this is what should be done (1:many, primary key of ‘one’ becomes foreign key of many)

Original Design

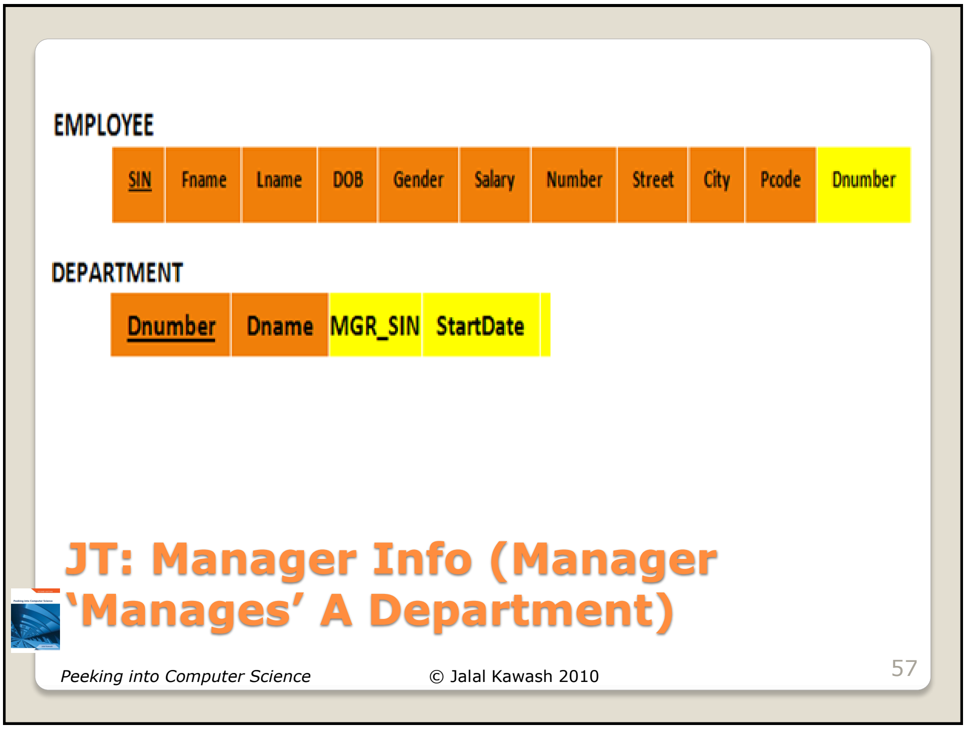


EMPLOYEE

SIN	Fname	Lname	DOB	Gender	Salary	Number	Street	City	Pcode	Dnumber
171717171	Debra	Beacon	15-Aug-1961	Female	70000	15	Baron Hill	Calgary	T2X Y0Y	1
181817178	Sam	Field	17-Feb-1978	Male	40000	15	Kick Way	Calgary	Y2K K0K	1
12345679	Rajeet	Folk	30-Apr-1967	Male	78000	123	One Road	Toronto	H1H J9J	2
987654321	Marie	Band	12-Jan-1985	Female	53500	2828	Exit Close	Toronto	K8O O8K	2
666333999	Saleh	Dice	25-Mar-1970	Male	90400	66	Straight Way	Toronto	T4E T6B	3

DEPARTMENT

Dnumber	Dname	MGR_SIN	StartDate
1	IT	171717171	12-Feb-2008
2	Finance	123456789	1-Mar-2002
3	Marketing	666333999	1-Jan-2005



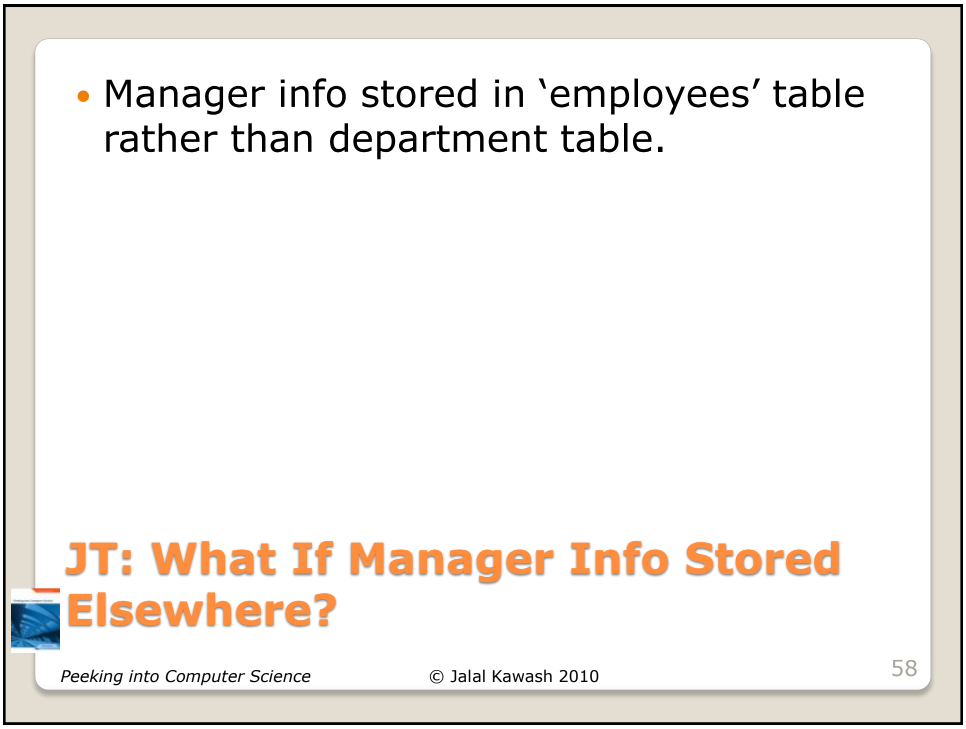
JT: Manager Info (Manager 'Manages' A Department)

EMPLOYEE

SIN	Fname	Lname	DOB	Gender	Salary	Number	Street	City	Pcode	Dnumber

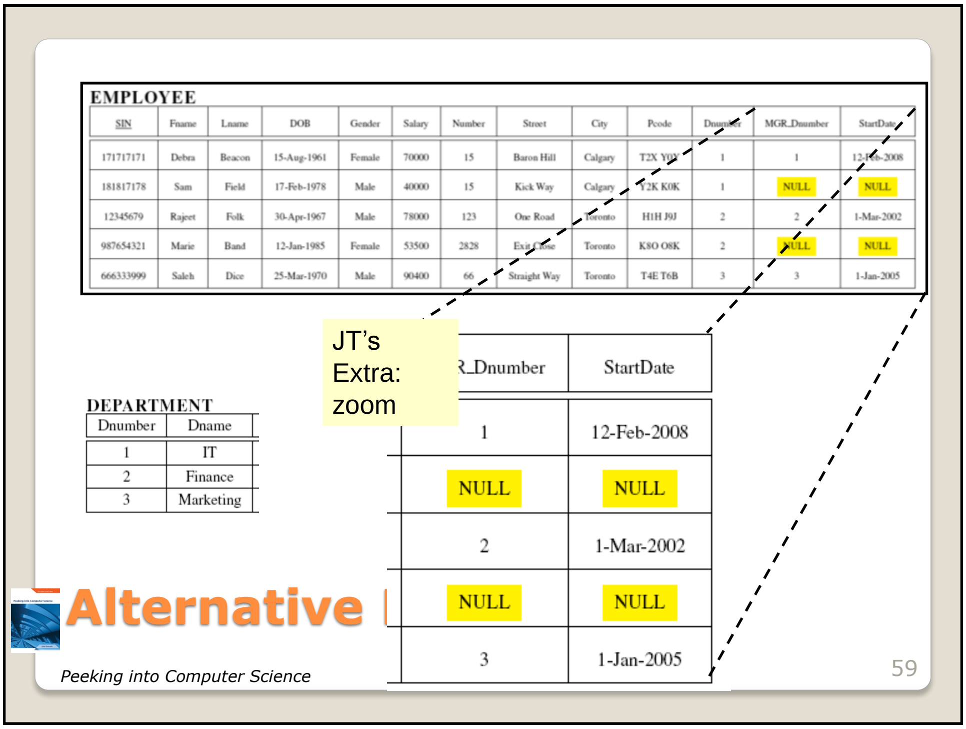
DEPARTMENT

Dnumber	Dname	MGR_SIN	StartDate



JT: What If Manager Info Stored Elsewhere?

- Manager info stored in 'employees' table rather than department table.



EMPLOYEE

SIN	Fname	Lname	DOB	Gender	Salary	Number	Street	City	Pcode	Dnumber	MGR_Dnumber	StartDate
171717171	Debra	Beacon	15-Aug-1961	Female	70000	15	Baron Hill	Calgary	T2X Y0Y	1	1	12-Feb-2008
181817178	Sam	Field	17-Feb-1978	Male	40000	15	Kick Way	Calgary	Y2K K0K	1	NULL	NULL
12345679	Rajeet	Folk	30-Apr-1967	Male	78000	123	One Road	Toronto	H1H J9J	2	2	1-Mar-2002
987654321	Marie	Band	12-Jan-1985	Female	53500	2828	Exit Close	Toronto	K8O O8K	2	NULL	NULL
666333999	Saleh	Dice	25-Mar-1970	Male	90400	66	Straight Way	Toronto	T4E T6B	3	3	1-Jan-2005

JT's Extra: zoom

R_Dnumber	StartDate
1	12-Feb-2008
NULL	NULL
2	1-Mar-2002
NULL	NULL
3	1-Jan-2005

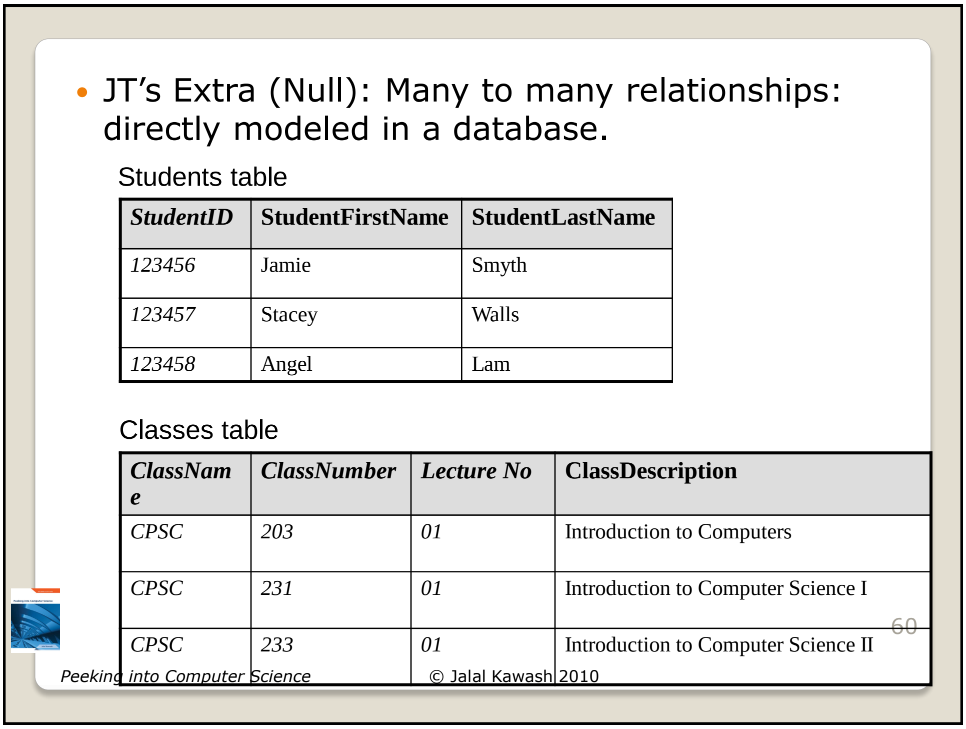
DEPARTMENT

Dnumber	Dname
1	IT
2	Finance
3	Marketing

Alternative

Peeking into Computer Science

59



- JT's Extra (Null): Many to many relationships: directly modeled in a database.

Students table

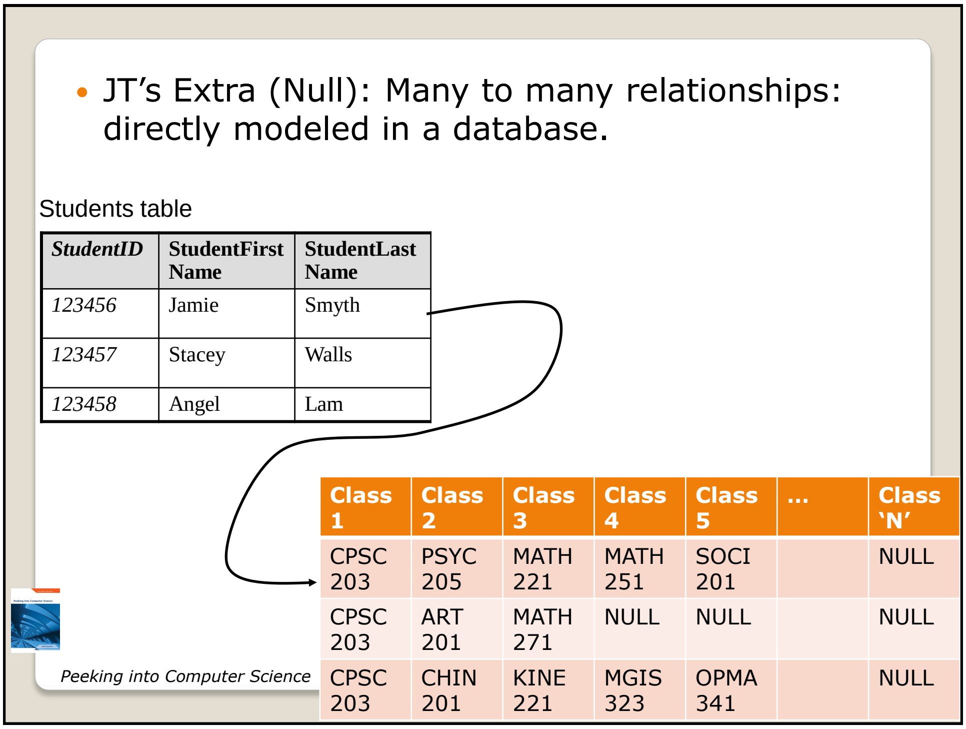
StudentID	StudentFirstName	StudentLastName
123456	Jamie	Smyth
123457	Stacey	Walls
123458	Angel	Lam

Classes table

ClassName	*ClassNumber*	*Lecture No*	ClassDescription
CPSC	*203*	*01*	Introduction to Computers
CPSC	*231*	*01*	Introduction to Computer Science I
CPSC	*233*	*01*	Introduction to Computer Science II

Peeking into Computer Science © Jalal Kawash 2010

60



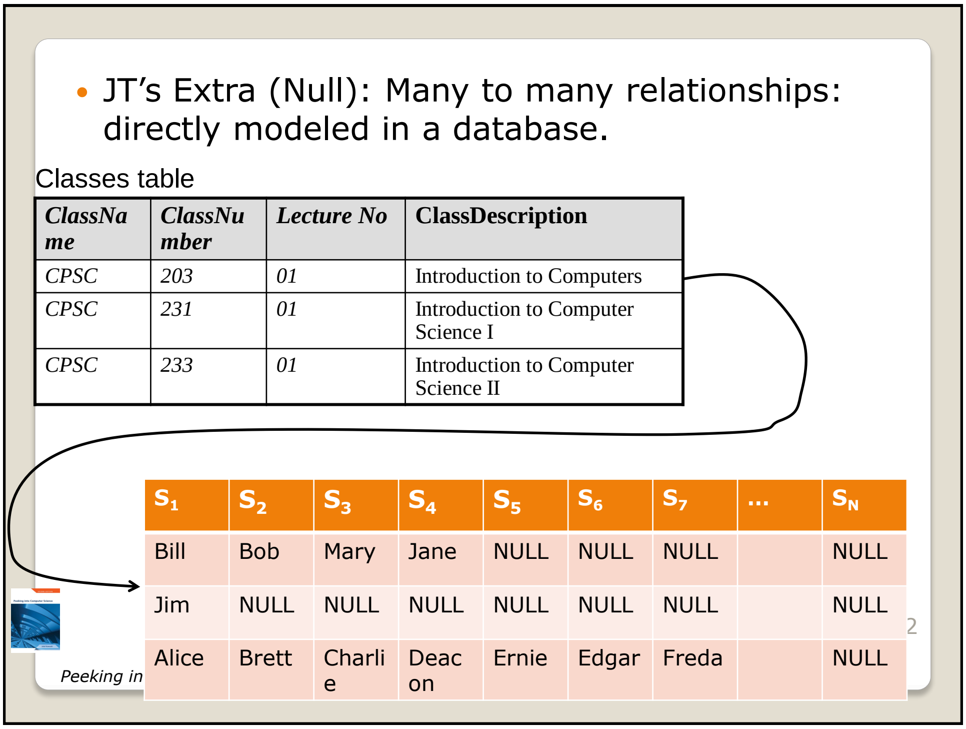
- JT's Extra (Null): Many to many relationships: directly modeled in a database.

Students table

StudentID	**StudentFirst Name**	**StudentLast Name**
123456	Jamie	Smyth
123457	Stacey	Walls
123458	Angel	Lam

Class 1	**Class 2**	**Class 3**	**Class 4**	**Class 5**	**…**	**Class 'N'**
CPSC 203	PSYC 205	MATH 221	MATH 251	SOCI 201		NULL
CPSC 203	ART 201	MATH 271	NULL	NULL		NULL
CPSC 203	CHIN 201	KINE 221	MGIS 323	OPMA 341		NULL

Peeking into Computer Science



- JT's Extra (Null): Many to many relationships: directly modeled in a database.

Classes table

ClassName	***ClassNumber***	***Lecture No***	**ClassDescription**
CPSC	*203*	*01*	Introduction to Computers
CPSC	*231*	*01*	Introduction to Computer Science I
CPSC	*233*	*01*	Introduction to Computer Science II

S_1	**S_2**	**S_3**	**S_4**	**S_5**	**S_6**	**S_7**	**…**	**S_N**
Bill	Bob	Mary	Jane	NULL	NULL	NULL		NULL
Jim	NULL	NULL	NULL	NULL	NULL	NULL		NULL
Alice	Brett	Charlie	Deacon	Ernie	Edgar	Freda		NULL

Peeking in

2