Chapter 15

Normalization for Relational Database

Addison-Wesley
is an imprint of
PEARSON

Data Normalization

- Primarily a tool to validate and improve a logical design so that it satisfies certain constraints that avoid unnecessary duplication of data
- The process of decomposing relations with anomalies to produce smaller, well-structured relations

Well-Structured Relations

- A relation that contains minimal data redundancy and allows users to insert, delete, and update rows without causing data inconsistencies
- Goal is to avoid anomalies
  - **Insertion Anomaly**–adding new rows forces user to create duplicate data
  - **Deletion Anomaly**–deleting rows may cause a loss of data that would be needed for other future rows
  - **Modification Anomaly**–changing data in a row forces changes to other rows because of duplication

General rule of thumb: A table should not pertain to more than one entity type

Addison-Wesley
is an imprint of
PEARSON

Example

EmpID	Name	Salary	Course#	CourseTitle	Date
100	Alaa	32000	459	SPSS	9/9/2016
100	Alaa	32000	876	Surveys	7/8/2016
140	Atheer	40000	333	Visual Basic	1/1/2016
150	Aisha	23000	459	SPSS	9/9/2016
150	Aisha	23000	901	C++	12/8/2016
140	Atheer	40000	901	C++	12/8/2016

Is this a relation?

Yes: Unique rows and no multivalued attributes

What’s the primary key?

Composite: Emp_ID, Course#

Anomalies in this Table

Insertion–can't enter a new employee without having the employee take a class. In addition, adding course for existing employee duplicate employees' data and course data

Deletion–if we remove employee 140, we lose information about the existence of a Visual Basic class

Modification–giving a salary increase to employee 100 forces us to update multiple records

Why do these anomalies exist?

Because there are two themes (entity types) in this one relation. This results in data duplication and an unnecessary dependency between the entities

Addison-Wesley
is an imprint of
PEARSON

Functional Dependency

- Functional dependency, denoted by $X \longrightarrow Y$ between two sets of attributes X, Y means that value of Y is determined by the value of X.
- The value of the *X* of a tuple uniquely (or **functionally)** ***determine the value of Y***
- Y is functionally dependent on X

Addison-Wesley
is an imprint of
PEARSON

Functional Dependency (2)

- Ssn→Ename
- The value of an employee's Social security number uniquely determines the employee name
- Pnumber →{Pname, Plocation}
- The value of a project's number (Pnumber) uniquely determines the project name and location
- {Ssn, Pnumber}→Hours
- A combination of Ssn and Pnumber values uniquely determines the number of hours the employee currently works on specific project

Normal Forms Based on Primary Key

- Most practical relational design projects take one of the following two approaches:
  - Perform a conceptual schema design using a conceptual model such as ER or EER and map the conceptual design into a set of relations
  - Design the relations based on external knowledge derived from an existing implementation of files or forms or reports

Normalization of Relations

- The normalization process takes a relation schema through a series of tests to certify whether it satisfies a certain **normal form.**
- There are three normal forms, which are *first, second, and third normal form*
- It is a purifying process that makes the design have better quality and minimizes redundancy

Normalization Steps

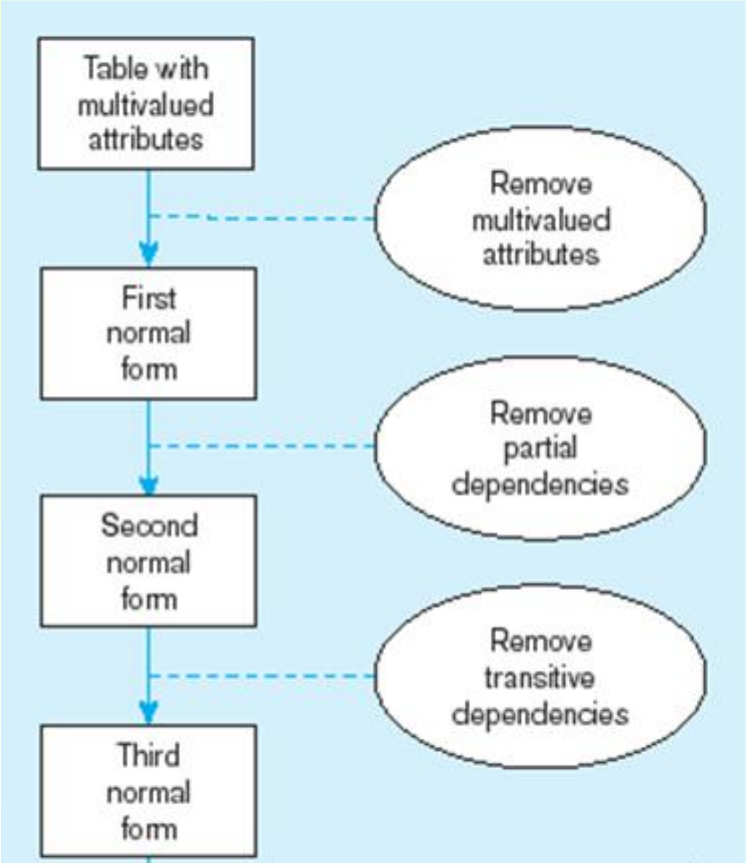


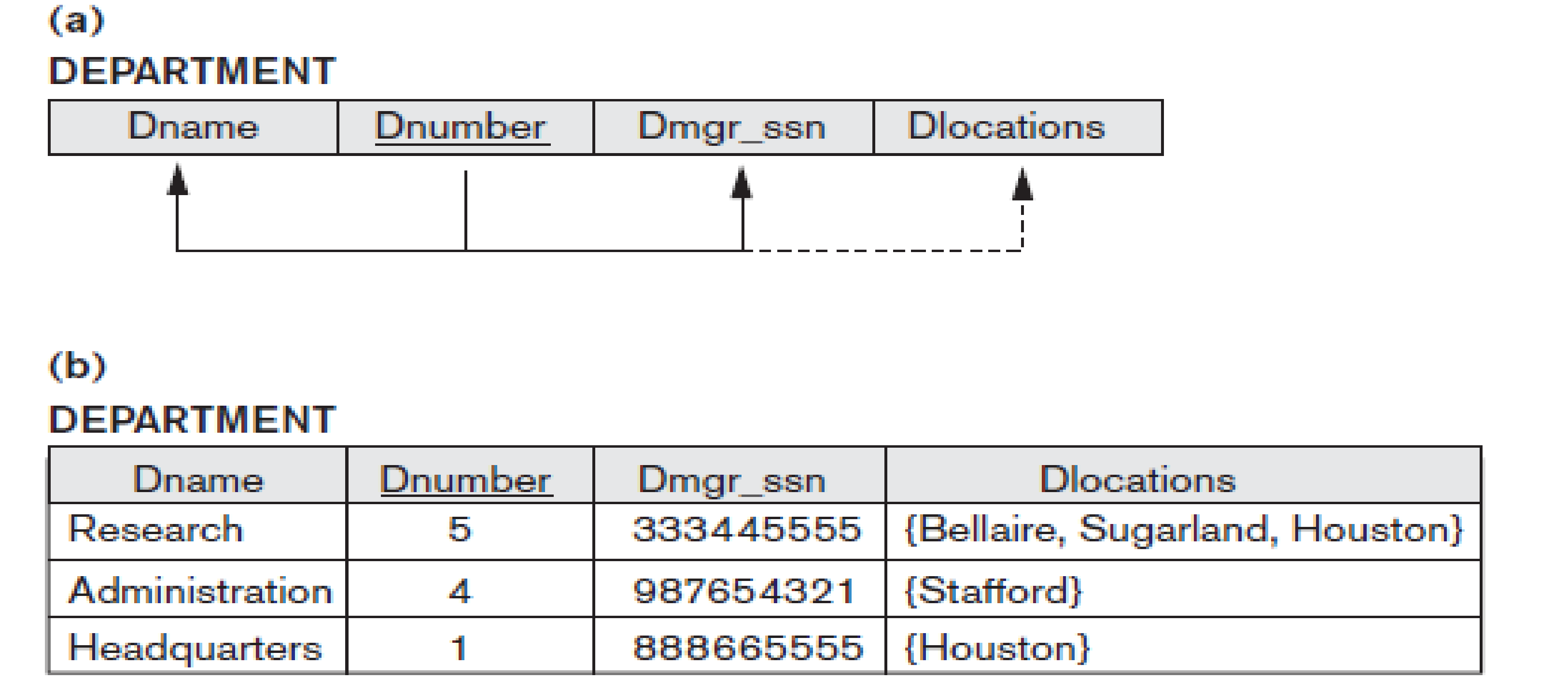
Table with multivalued attributes
Remove multivalued attributes
First normal form
Remove partial dependencies
Second normal form
Remove transitive dependencies
Third normal form

Addison-Wesley
is an imprint of
PEARSON

First Normal Form

- For a relation to be in 1st normal form, redundant groups or multivalued attributes should be removed
- To change to 1NF:
  - Remove nested relation attributes into a new relation
  - Propagate the primary key into it

Removing multivalued attributes



(a)

DEPARTMENT

Dname	<u>Dnumber</u>	Dmgr_ssn	Dlocations

(b)

DEPARTMENT

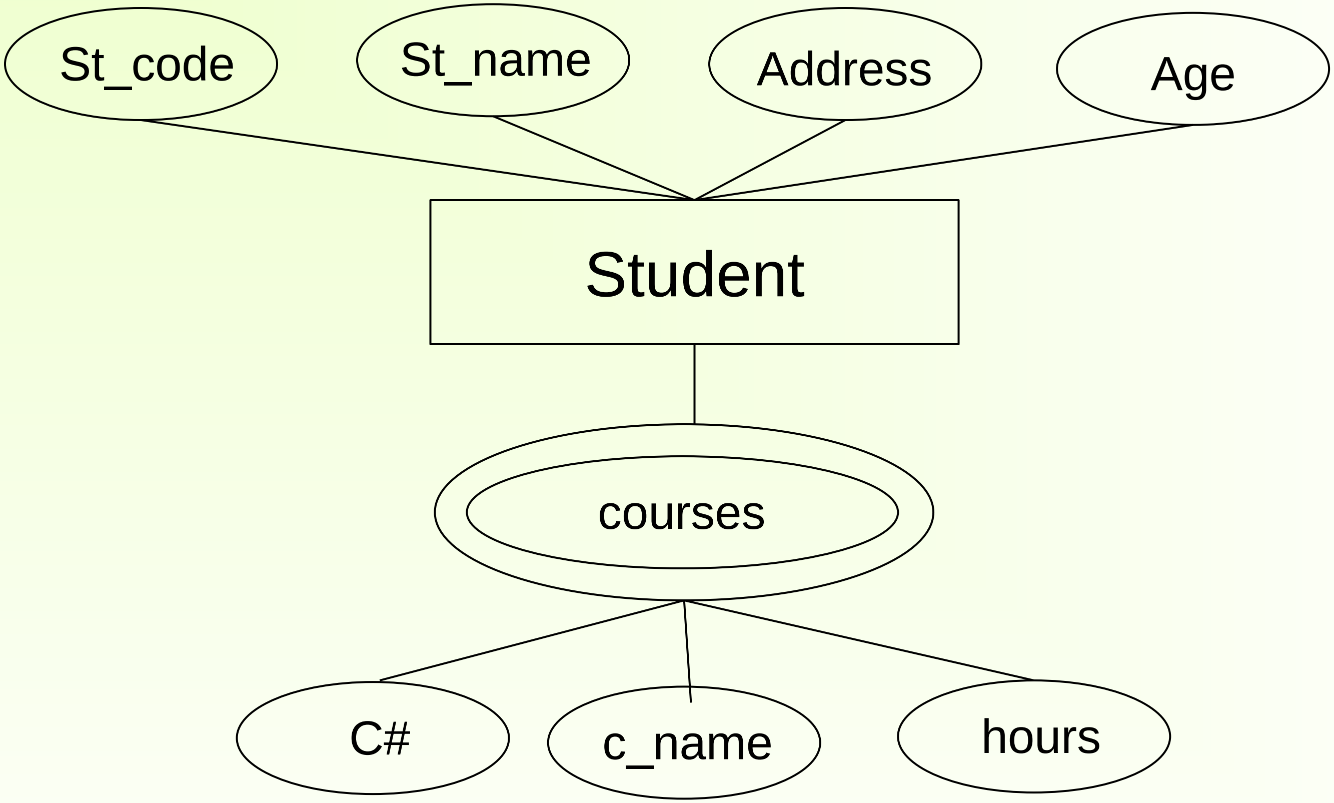
Dname	Dnumber	Dmgr_ssn	Dlocations
Research	5	333445555	{Bellaire, Sugarland, Houston}
Administration	4	987654321	{Stafford}
Headquarters	1	888665555	{Houston}

New relation to be in 1st normal form

Dnumber	Dlocations

Addison-Wesley
is an imprint of
PEARSON

Removing Relation within Relation or Repeating Groups

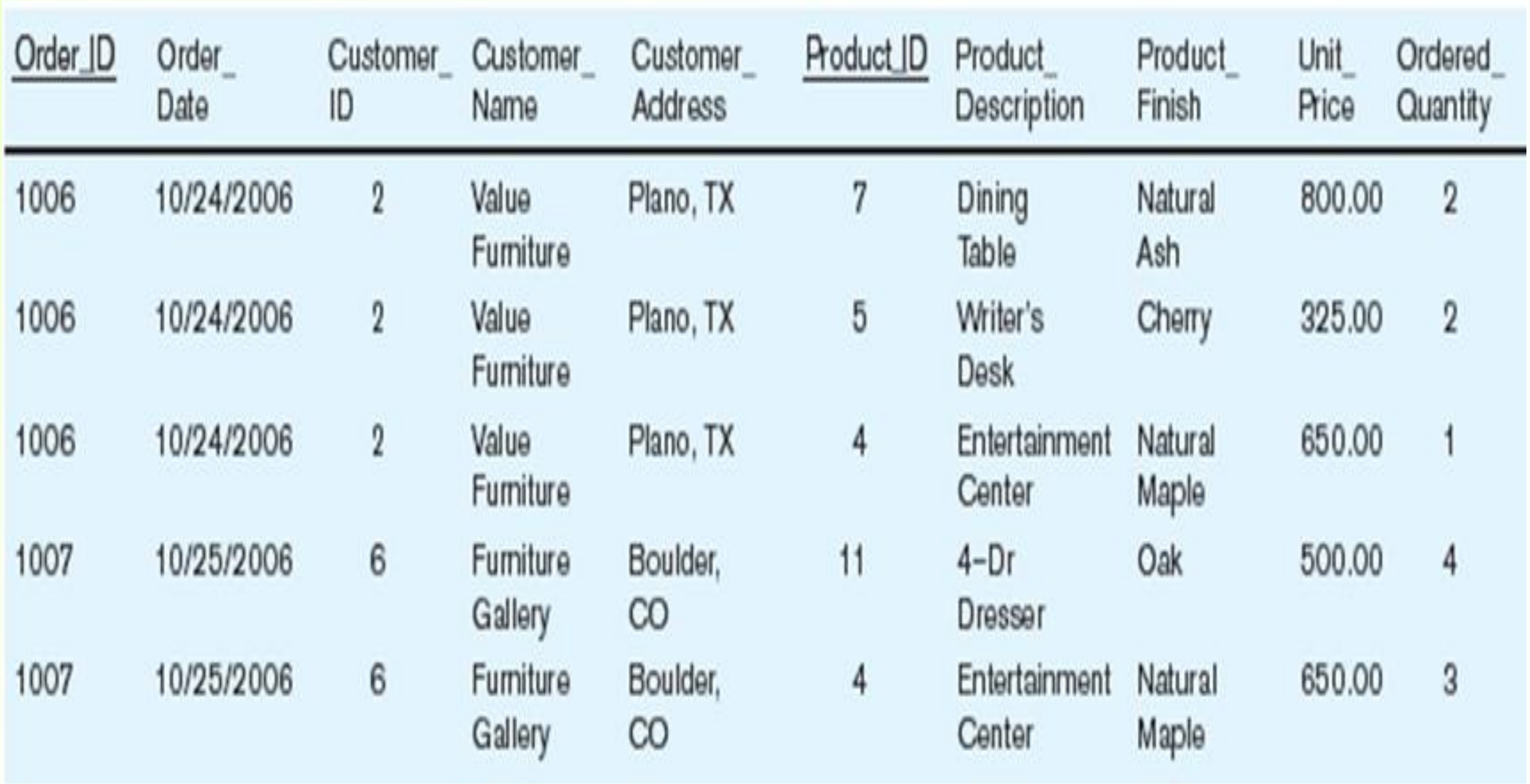


Addison-Wesley
is an imprint of
PEARSON

Example Repeating Groups

St_code	St_name	Address	Age	C#	C_name	Hours
20111	Karim	AlRawda	18			
				DS34	Data Str.	3
				OR23	Ope. Rese	3
				DB12	Database	3
20112	Mona	Alfayhaa	17			
				N22	Network	2
				OR23	Ope. Rese	3
				Db12	Database	3
				DS34	Data Str.	3

Example Repeating Groups



Order_ID	Order_Date	Customer_ID	Customer_Name	Customer_Address	Product_ID	Product_Description	Product_Finish	Unit_Price	Ordered_Quantity
1006	10/24/2006	2	Value Furniture	Plano, TX	7	Dining Table	Natural Ash	800.00	2
1006	10/24/2006	2	Value Furniture	Plano, TX	5	Writer's Desk	Cherry	325.00	2
1006	10/24/2006	2	Value Furniture	Plano, TX	4	Entertainment Center	Natural Maple	650.00	1
1007	10/25/2006	6	Furniture Gallery	Boulder, CO	11	4-Dr Dresser	Oak	500.00	4
1007	10/25/2006	6	Furniture Gallery	Boulder, CO	4	Entertainment Center	Natural Maple	650.00	3

Addison-Wesley
is an imprint of
PEARSON

To Be in First Normal Form

The previous data to be in 1st normal form, split the repeating group in another table with the primary key of the relation to be as foreign key in the new relation

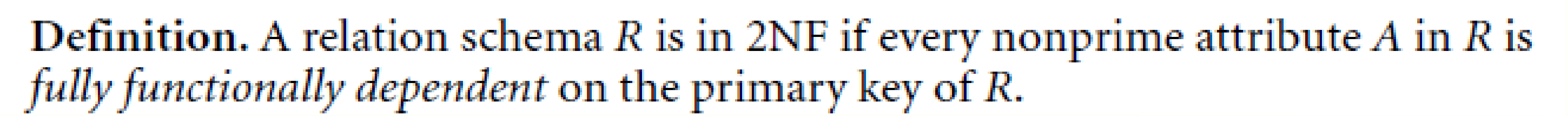
St_code	St_name	Address	Age

St_code	C#	C_name	Hours

Addison-Wesley
is an imprint of
PEARSON

Second Normal Form

- Based on concept of full functional dependency
  - Versus **partial dependency-** Remove partial dependency



Definition. A relation schema *R* is in 2NF if every nonprime attribute *A* in *R* is *fully functionally dependent* on the primary key of *R*.

- Second normalize into a number of 2NF relations
  - Nonprime attributes are associated only with part of primary key on which they are fully functionally dependent

Second Normal Form (2)

SSN	P#	Hours	Ename	Pname	Plocation

- Ename is functionally dependent on SSN only, no need for P#
- Pname and Plocation are functionally dependent on P#
- So this relation is not in the second normal form
- Solution:

SSN	P#	Hours

SSN	Ename

P#	Pname	Plocation

Third Normal Form

- To be in 3rd normal form you should remove transitive dependency
- Non key attribute is dependent on non key attribute

Third Normal Form (2)

SSN	Ename	Address	Dept_code	Dept_name	Mgr_SSN

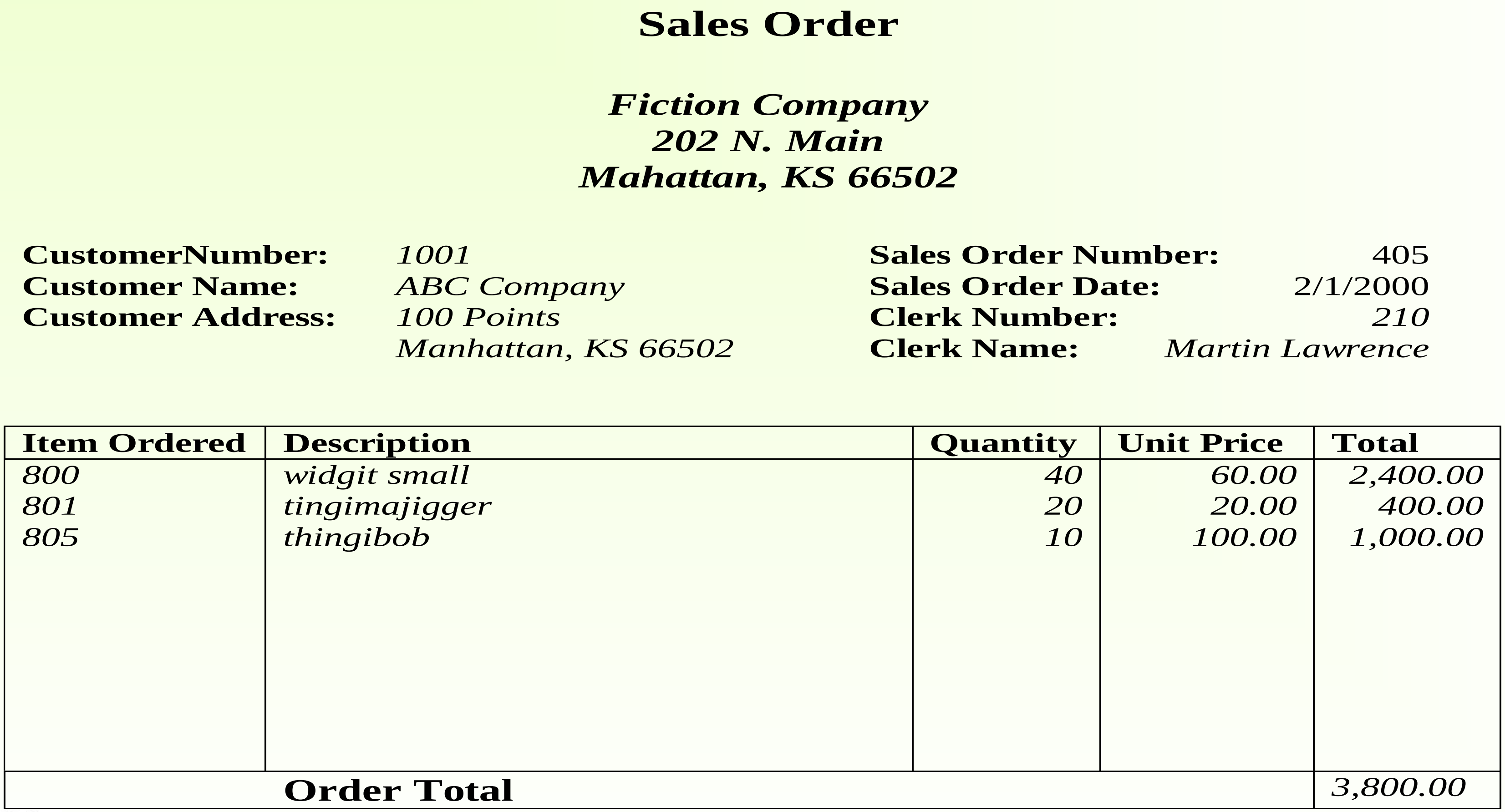
- Dept_name and mgr_SSN are functionally dependent on dept_code which is non key attribute
- Solution:

SSN	Ename	Address	*Dept_code*

Dept_code	Dept_name	Mgr_SSN

Exercises

Consider the following report, suppose sales order number with item ordered is the primary key, normalize this report to reach better design



Sales Order

Fiction Company
202 N. Main
Mahattan, KS 66502

CustomerNumber:	*1001*	**Sales Order Number:**	405
Customer Name:	*ABC Company*	**Sales Order Date:**	2/1/2000
Customer Address:	*100 Points* *Manhattan, KS 66502*	**Clerk Number:**	*210*
		Clerk Name:	*Martin Lawrence*

Item Ordered	Description	Quantity	Unit Price	Total
800	*widgit small*	*40*	*60.00*	*2,400.00*
801	*tingimajigger*	*20*	*20.00*	*400.00*
805	*thingibob*	*10*	*100.00*	*1,000.00*
	Order Total			*3,800.00*

Relation

R=salesOrderNo, salesOrderDate, custNo, custName, address clerkNo, clerkName, {itemsOrdered, description, quantity, unitPrice}

To be in 1st normal form, remove repeating group:

R1= salesOrderNo, salesOrderDate, custNo, custName, address clerkNo, clerkName

R2= salesOrderNo, itemsOrdered, description, quantity, unitPrice

2nd Normal Form

R1= salesOrderNo, salesOrderDate, custNo, custName, address clerkNo, clerkName

R2.1= salesOrderNo, itemsOrdered, quantity

R2.2= itemsOrdered, description, unitPrice

3rd Normal Form

R1.1= salesOrderNo, salesOrderDate, custNo, clerkNo

R1.2= custNo, custName, address

R1.3= clerkNo, clerkName

R2.1= salesOrderNo, itemsOrdered, quantity

R2.2= itemsOrdered, description, unitPrice

Exercises

- Normalize the following schemas into 3rd normal form:
- BRANCH (Branch#, Branch_Addr, {ISBN, Title, Author, Publisher, Num_copies})

1st Normal Form:

R1: Branch#, Branch_addr

R2: Branch#, ISBN, title, author, publisher, num_copies

2nd Normal Form:

R1: Branch#, Branch_addr
R2.1: Branch#, ISBN, num_copies
R2.2: ISBN, title, author, publisher

3rd Normal Form:

No change in the previous relations.

Exercises

- Project code, project title, project manager, project budget {employeeNo, employeename,completed_hour, departmentNo, department_name, rate_per_hour}
- ***Note:*** rate per hour for each employee is fixed regardless of the project. Completed hour means the number of hours employee accomplished in this project

- **1st Normal Form:**

R1: Pcode, ptitle, pmgr, pbudget

R2: Pcode, empNo, empName, c_hours, dNo, dName, rate

- **2nd Normal Form:**

R1: Pcode, ptitle, pmgr, pbudget

R2.1: Pcode, empNo, c_hours

R2.2: empNo, empName, dNo, dName, rate

- **3rd Normal Form:**

R1: Pcode, ptitle, pmgr, pbudget

R2.1: Pcode, empNo, c_hours

R2.2.1: empNo, empName, dNo, rate

R2.2.2: dNo, dName

Exercises

Al Salam Hospital - Doctor's report   Date: 10/5/2004

Doctor Id. : A121   Doctor Name : Dr. Ahmed

Department Id : A   Department Name : Internal Diseases

P#	Pat-name	Address	Given Treatments			
			Item#	Description	Quantity	Unit Price
10	Saleh	Maadi	A01	Aspirin	10	1.5
			A03	Panadol	6	3.5
			B01	Vitamin C	12	4.0

Relation

R=docId, docname, deptId, deptName,{ patient(p#,pname,address){given treatment (item#, description, quantity, unit_price)}}

First Normal Form

- R1= docId, docname, deptId, deptName
- R2= docId, p#, pname, address
- R3=docId, p#, item#, description, qantity, unit_price

Second Normal Form

- R1= docId, docname, deptId, deptName
- R2.1= docId, p#
- R2.2= p#, pname, address
- R3=docId, p#, item#, qantity
- R3.2= item#, description, unit_price

Addison-Wesley
is an imprint of
PEARSON

Third Normal Form

- R1.1= docId, docname, deptId
- R1.2=deptId, deptName
- R2.2= p#, pname, address
- R3=docId, p#, item#, qantity
- R3.2= item#, description, unit_price