DBMS – Session3 [Normalization]

Course Objective

- To enable the participants to understand Translation of ER-model into relation schema.
- To enable the participants to understand the Normalization concept
  - – 1NF, 2NF, 3NF and BCNF

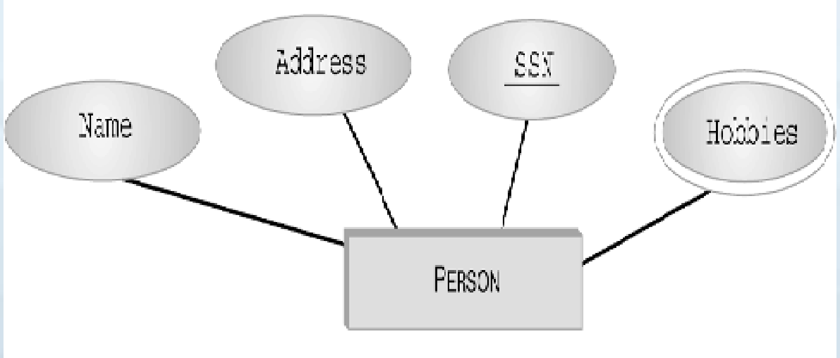
Session Objective

- ER Relational Mapping
- Normalization
- Need for Normalization
- First Normal Form (1NF)
- Second Normal Form (2NF)
- Third Normal Form (3NF)
- Boyce-Codd Normal Form (BCNF)

ER to Relational Mapping

The following step by step process is applied to on ER diagrams to derive relations

- **Step 1: Mapping of Regular/Strong entities**
  - Create relation (table) for each regular/strong entities
  - Create columns for all the simple/composite/single/stored attributes of this strong/regular entity
  - Create primary key (only one) from the key attribute(s)
  - Ignore derived attributes if any



Person (Name, Address, <u>SSN</u>)

Relation Name: Person

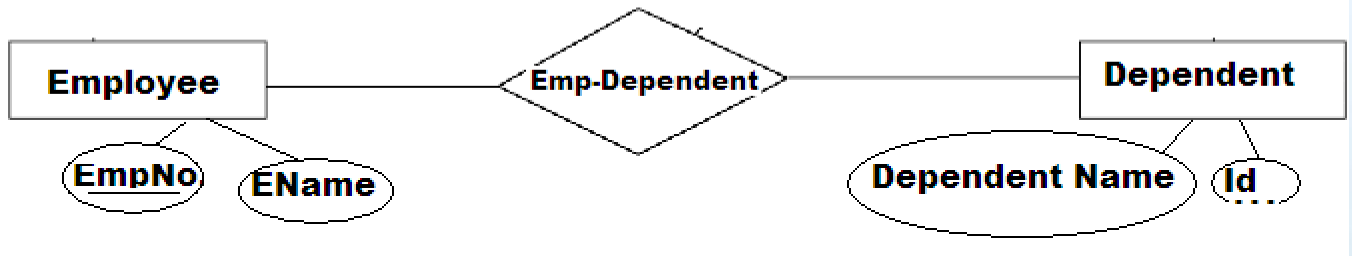
Columns: Name, Address, SSN

Primary Key: SSN

ER to Relational Mapping contd…

Step 2: Mapping of weak entities

- Create relation (table) for each weak entity
- Create columns for all simple/composite attributes
- Create a foreign key column by including primary key column of its strong entity
- Create primary key by combining foreign key column (which refers primary key column of its strong entity) and partial key column
  - Partial key is key attribute with respect to weak entity



Dependent (Dependent Name, …, EmpNo,Id)

Relation Name: Dependent

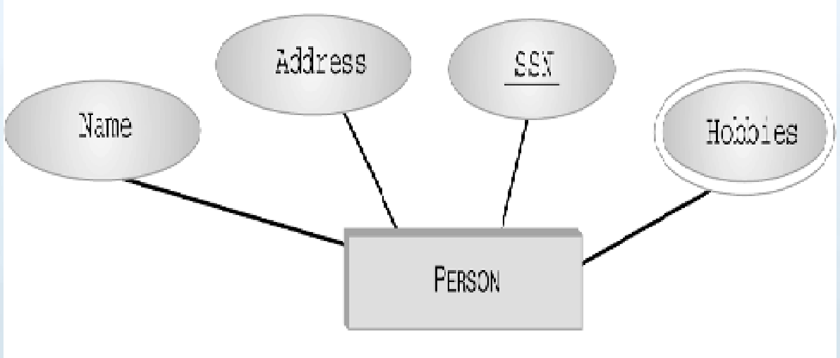
Columns: Dependent Name, Empno, id

Primary Key: Empno + Id Foreign key: EmpNo

ER to Relational Mapping

Step 3: Mapping of Multi-valued attributes

- Create relation (table) for multi-valued attributes
- create primary key by combining the primary key attribute of Entity and multi-valued attribute



Hobbies (SSN,Hobby)

Relation Name: Hobbies

Columns: SSN, Hobby

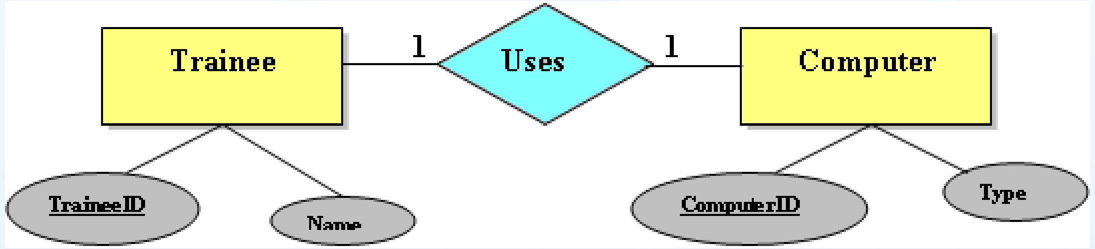
Primary Key: SSN + Hobby

Foreign key: SSN

ER to Relational Mapping contd…

Step 4: Mapping of Relationship types (Binary)– 1:1

- Include one attribute (as a foreign key) on the optional side



Trainee(TraineeId, Name, …)

Computer (ComptuerID, Type, …,Usedby)

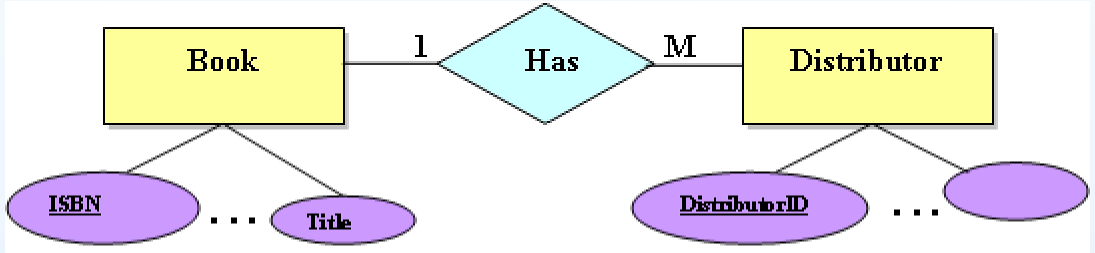
Primary Key: ComptuerId

Foreign key: Usedby

ER to Relational Mapping contd…

Step 5: Mapping of Relationship types (Binary)– 1: M

- Include a attribute (as a foreign key) for the relationship at the M side of the entity table



Book (ISBN, Title …)

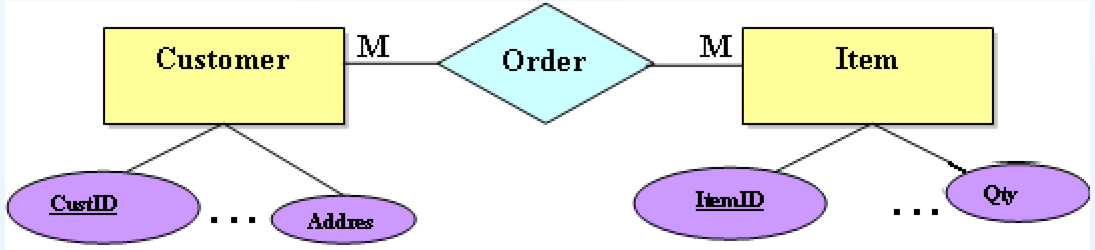
Distributor (DistributorID, Book…)

Primary Key (for Book): Empno Foreign key: Book

(for Distributor): DistributorID

ER to Relational Mapping contd…

- **Step 6: Mapping of Relationship types (Binary) – M: N**
  - Create a new relation for the relationship.
  - Create primary key attribute by combining the primary key attributes both the participating entities



Customer(CustID, Addrees …)

Item (ItemID, Qty, …)

Orders(CustID, ItemID, …)

Primary Key (for customer): CustID Foreign key (for Orders): CustID, ItemID

(for Item): ItemID

(for Orders): CustId+ItemID

ER to Relational Mapping contd…

Step 7: Mapping of Relationship types (unary)– 1:M and M:N

One-to-Many

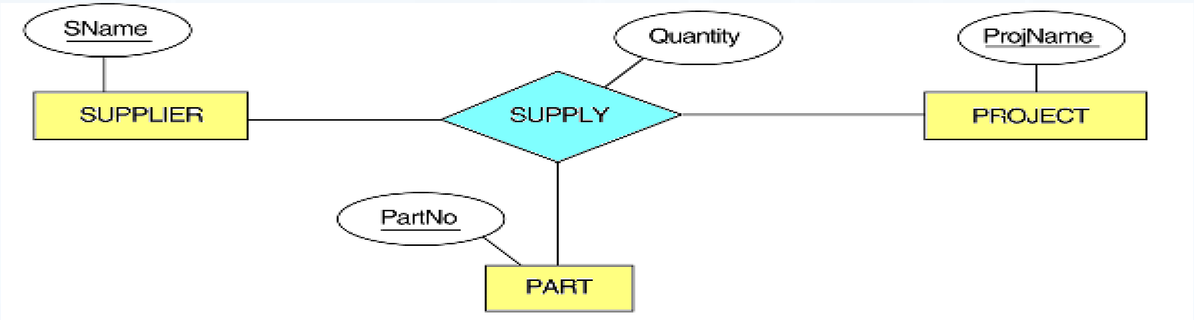
- Include an attribute(as a recursive foreign key) in the same relation

Many-to-Many

- Create a new relation
- Include a primary attribute by combing the primary key of participating entity and the relationship

ER to Relational Mapping contd…

- **Step 8: Mapping of Ternary Relationship types**
  - Create a new table for the relationship type
  - Include a primary key attribute by combing all the participating entities primary key



Supplier(SName, …)

Project (ProjName, …)

Part (PartNo,…)

Supply(SName, ProjName, PartNo,Quantity …)

Primary Key (for Supplier): SName Foreign key (for supply): SName, ProjName, PartNo

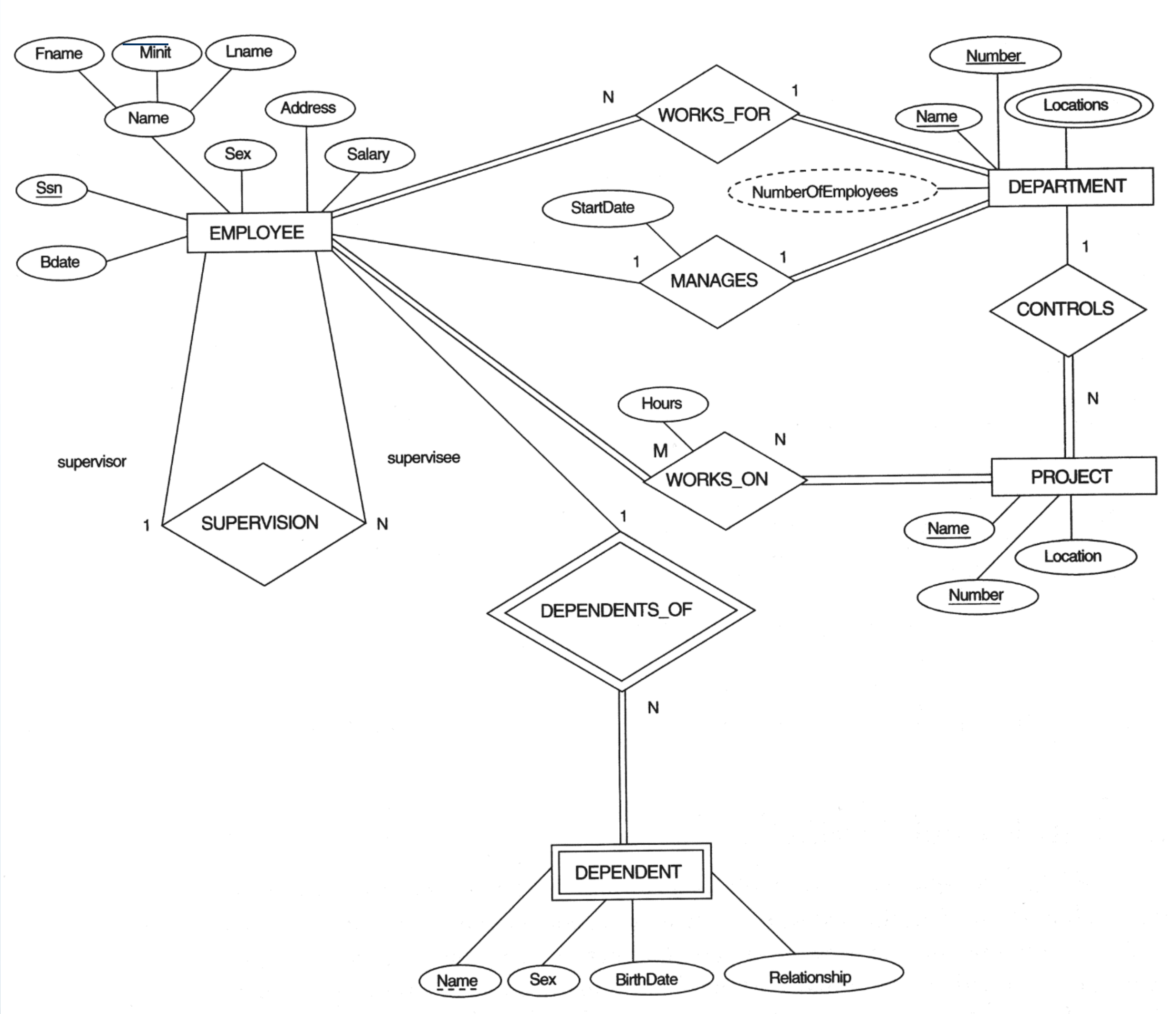
(for Project): ProjName

(for Part): PartNo

(for Supply):SName, ProjName, PartNo

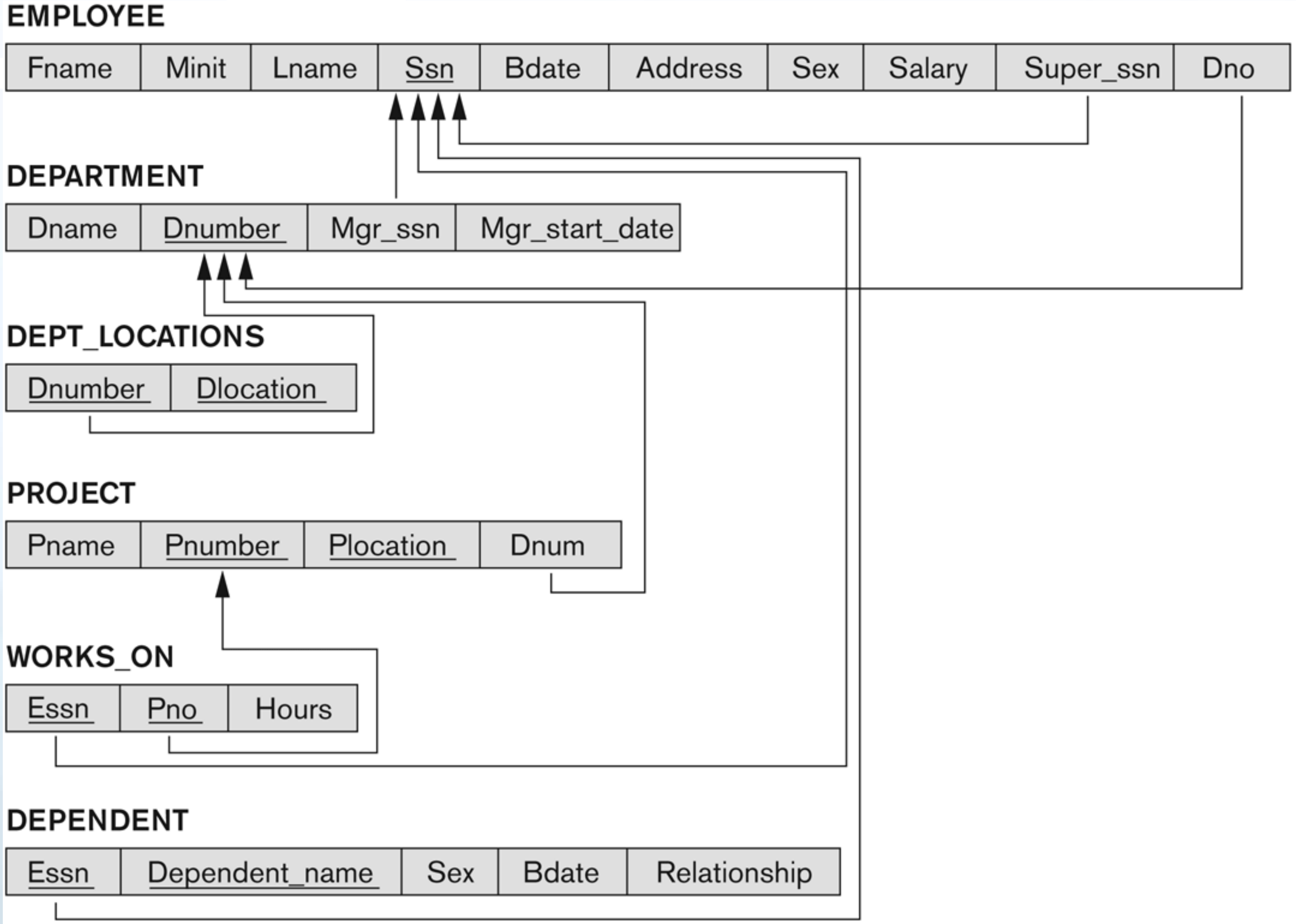
ER diagram for the COMPANY database

CASE STUDY



Mapping the COMPANY ER schema into a relational schema

CASE STUDY



EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno

DEPARTMENT

Dname	Dnumber	Mgr_ssn	Mgr_start_date

DEPT_LOCATIONS

Dnumber	Dlocation

PROJECT

Pname	Pnumber	Plocation	Dnum

WORKS_ON

Essn	Pno	Hours

DEPENDENT

Essn	Dependent_name	Sex	Bdate	Relationship

Courtesy: Fundamentals of DB system by Elmasri

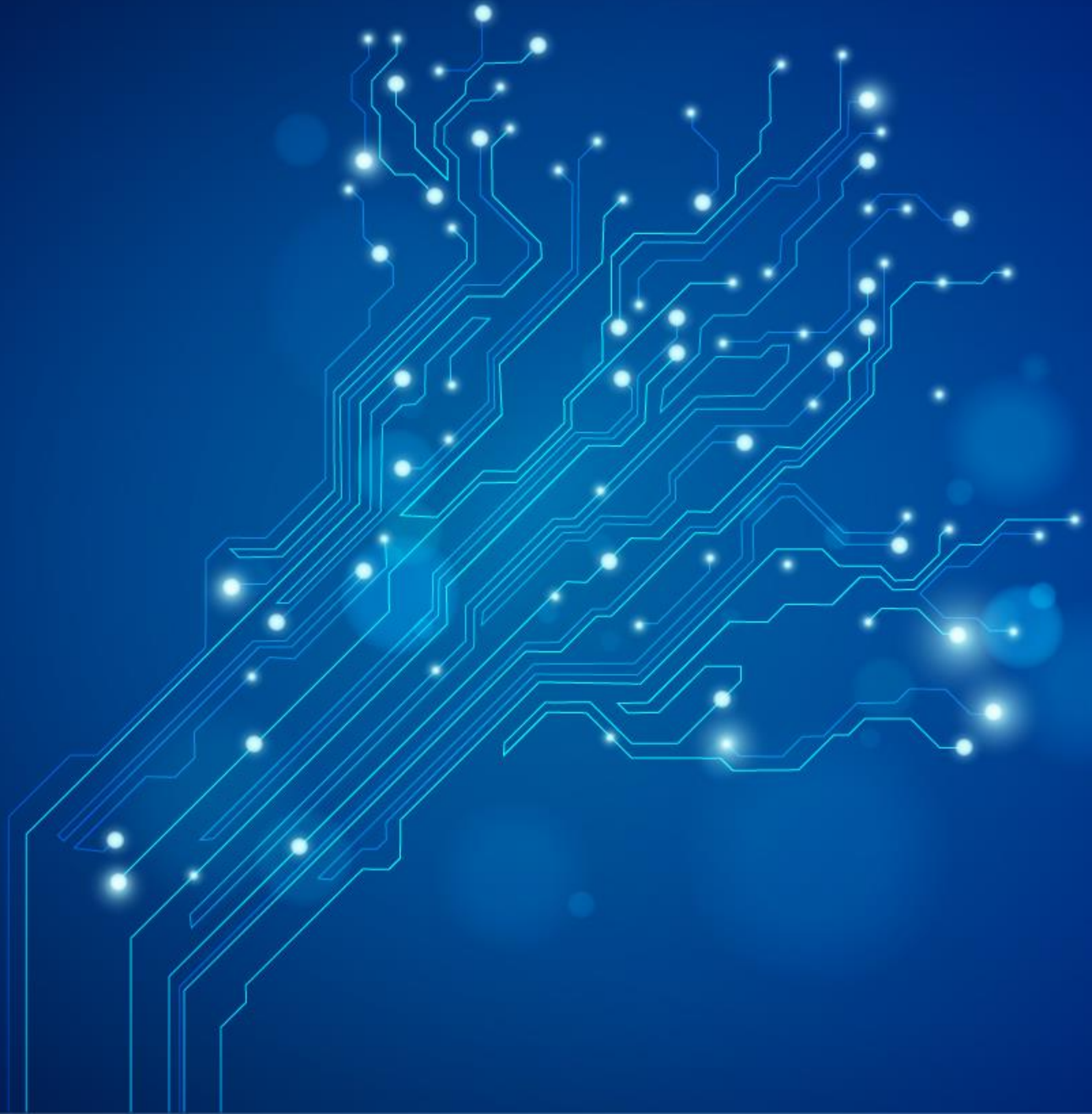
GAMIFICATION
$

Gamification

Objective:

To make the participants familiarize with Normalization through activity.

Activity on Normalization



Normalization

Normalization

- Process of decomposing relations with anomalies to produce smaller, ***well-structured*** relations

- Primarily a tool to validate and improve a logical design so that it satisfies certain constraints that **avoid unnecessary duplication of data**

- **A well-structured relations** is a relation, which contains minimal data redundancy and allows users to insert, delete, and update rows without causing data inconsistencies

Un Normalized Table

Student_id	Name	Branch	Hod	Office_tel	Subject	Marks	Teacher	Examname	Totalmarks
1	Amit	CSE	Mr. X	23347	SQL	40	Mr. Ram	Practicals	40
2	Shah	CSE	Mr. X	23347	C, JAVA	60,65	Mr.John, Mr. Kent	Theory exam	70
3	Shreya	CSE	Mr. X	23347	DS, OS	45,40	Mr. Sam, Mr.John	Sessionals	50
4	Dheena	CSE	Mr. X	23347	JAVA	60	Mr.Kent	Theory exam	70

Data Redundancy

STUDENTS TABLE

Student_id	Name	Branch	HOD	Office_tel
1	Amit	CSE	Mr. X	23347
2	Shah	CSE	Mr. X	23347
3	Shreya	CSE	Mr. X	23347
4	Dheena	CSE	Mr. X	23347

Unnecessary data repetition increases the size of the database.

Need for Normalization

- **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

Need for Normalization

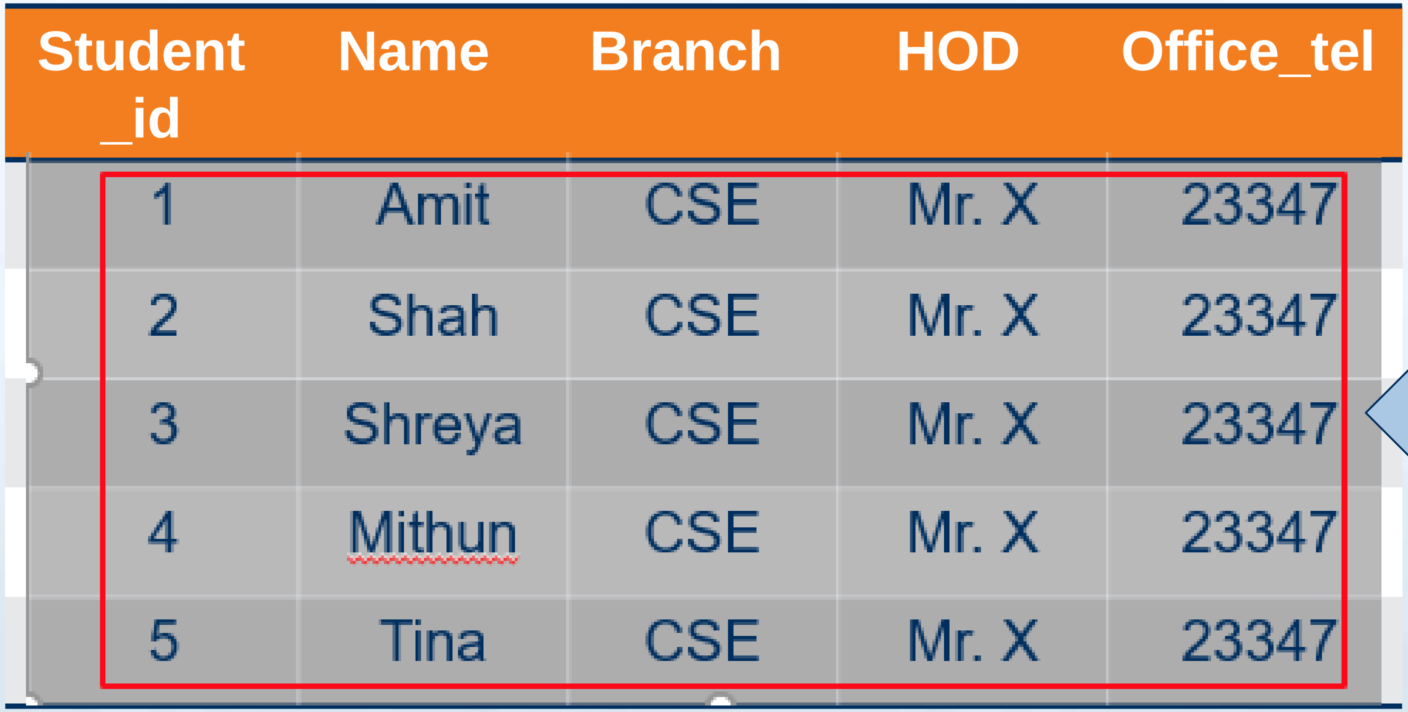
- **Insertion Anomaly**:
  - To insert redundant data for every new row is a data insertion problem or anomaly.

Student_id	Name	Branch	HOD	Office_tel
1	Amit	CSE	Mr. X	23347
2	Shah	CSE	Mr. X	23347
3	Shreya	CSE	Mr. X	23347
4	Dheena	CSE	Mr. X	23347
5	Tina	CSE	Mr. X	23347

To insert new student data repeating the same branch, hod, office_tel information.

Need for Normalization

- **Deletion Anomaly**:



Student _id	Name	Branch	HOD	Office_tel
1	Amit	CSE	Mr. X	23347
2	Shah	CSE	Mr. X	23347
3	Shreya	CSE	Mr. X	23347
4	Mithun	CSE	Mr. X	23347
5	Tina	CSE	Mr. X	23347

To delete the student information- branch, hod, office_tel information also gets deleted.

- Branch information deleted along with Student data.

Need for Normalization

- **Updation Anomaly**:

Student _id	Name	Branch	HOD	Office_tel
1	Amit	CSE	~~Mr. X~~ Mr. Y	23347
2	Shah	CSE	~~Mr. X~~ Mr. Y	23347
3	Shreya	CSE	~~Mr. X~~ Mr. Y	23347
4	Dheena	CSE	~~Mr. X~~ Mr. Y	23347
5	Tina	CSE	~~Mr. X~~ Mr. Y	23347

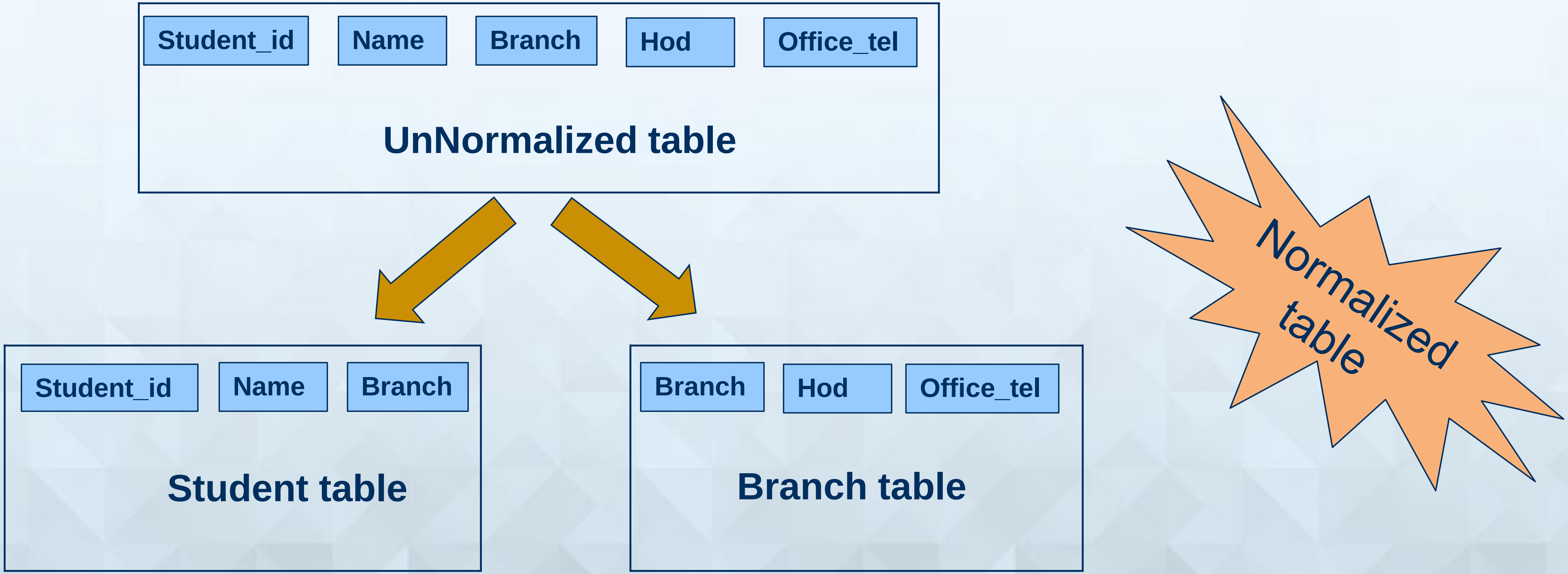
Mr. X leaves and Mr. Y joins as the new HOD for CSE. Its need to update whole student record.

Normalization

Data Redundancy:

- Repetition of data hence needs extra space.
- Leads to insertion, deletion and updation issues.

Normalization will solve these problem.



Student_id | Name | Branch | Hod | Office_tel

UnNormalized table

Student_id | Name | Branch

Student table

Branch | Hod | Office_tel

Branch table

Normalized table

Normalized table

STUDENTS TABLE

STUDENT_ID	NAME	BRANCH
1	Amit	CSE
2	Shah	CSE
3	Shreya	CSE

Only branch information gets repeated.

BRANCH TABLE

BRANCH	HOD	OFFICE_TEL
CSE	Mr. Y	23347

Now branch table is independent to update, insert and delete the information.

- Normalization is not eliminating redundancy. Its minimizing redundancy

1NF- First Normal Form

A table/relation have the following,

- No multi-valued attributes
- Every attribute value is atomic

How to make it in INF

1NF- First Normal Form

As per the rule of first normal form, an attribute (column) of a table cannot hold multiple values. It should hold only atomic values.

STUDENTS TABLE

STUDENT_ID	NAME	SUBJECT
1	Amit	SQL
2	Shah	C, JAVA
3	Shreya	DS, OS

It holds multiple values

Violation in 1NF

First Normal Form (1NF)

STUDENTS TABLE

STUDENT_ID	NAME	SUBJECT
1	Amit	SQL
2	Shah	C
2	Shah	JAVA
3	Shreya	DS
3	Shreya	OS

Functional Dependency (FD)

- Functional dependency describes relationship between attributes
- 2nd Normal Form and 3rd Normal Form are based on Functional dependency
- Functional dependency (FD) definition:
  - if A and B are attributes of relation R, B is functionally dependent on A (denoted A → B), if each value of A in R is associated with exactly one value of B in R

For example:

STUDENT_ID	NAME	BRANCH
1	Amit	CSE
2	Amit	CSE
3	Shreya	CSE

Primary key

NAME is functionally dependent on **Student_id**

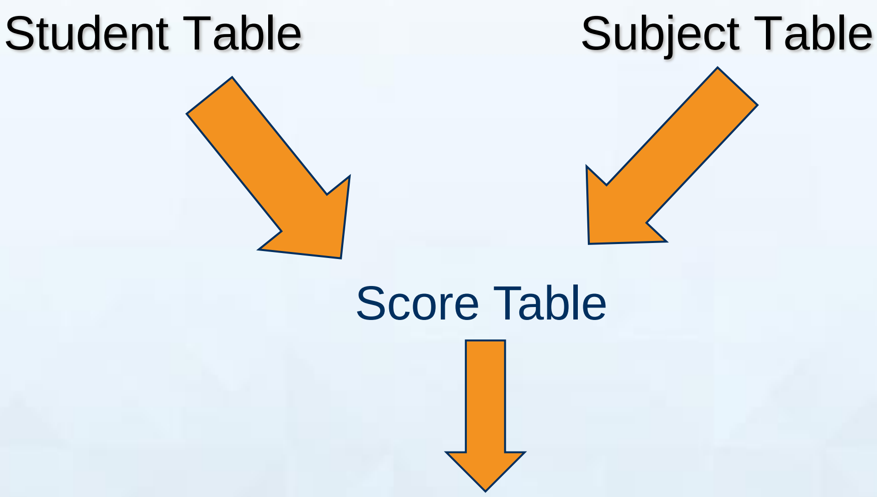
Here students name are same, As the student_id in this table will be unique, it can be used easily to fetch any data.

2NF- Second Normal Form

Rule 1- Be in 1NF

Rule 2- No partial dependencies in the table

For Example:



Student Table

Subject Table

Score Table

To save marks obtained by students in each subject.

2NF- Second Normal Form

SCORE TABLE

Score_id should be Primary key.
But
Student_id+Subject_id together makes a more meaningful primary key.
Student_id+Subject_id can uniquely identify any row of data in SCORE table

Teacher column only depends on Subject and not on Student. So its partial dependency.

Violation in 2NF

Score_id	Student_id	Subject_id	marks	teacher
1	1	1	90	Mr. Ram
2	1	2	85	Mr. John
3	2	1	80	Mr. Ram
4	2	2	75	Mr. John
5	2	3	82	Mr. Kentt

Primary key

Marks is functionally dependent on Student_id and Subject_id

2NF- Second Normal Form

Score Table

Score_id	Student_ id	Subject_ id	marks	teacher
1	1	1	90	Mr. Ram
2	1	2	85	Mr. John
3	2	1	80	Mr. Ram
4	2	2	75	Mr. John
5	2	3	82	Mr. Kentt

The partial dependent teacher column is removed from score table

Second Normal Form (2NF)

Score Table

Score_id	Student_ id	Subject_ id	marks
1	1	1	90
2	1	2	85
3	2	1	80
4	2	2	75
5	2	3	82

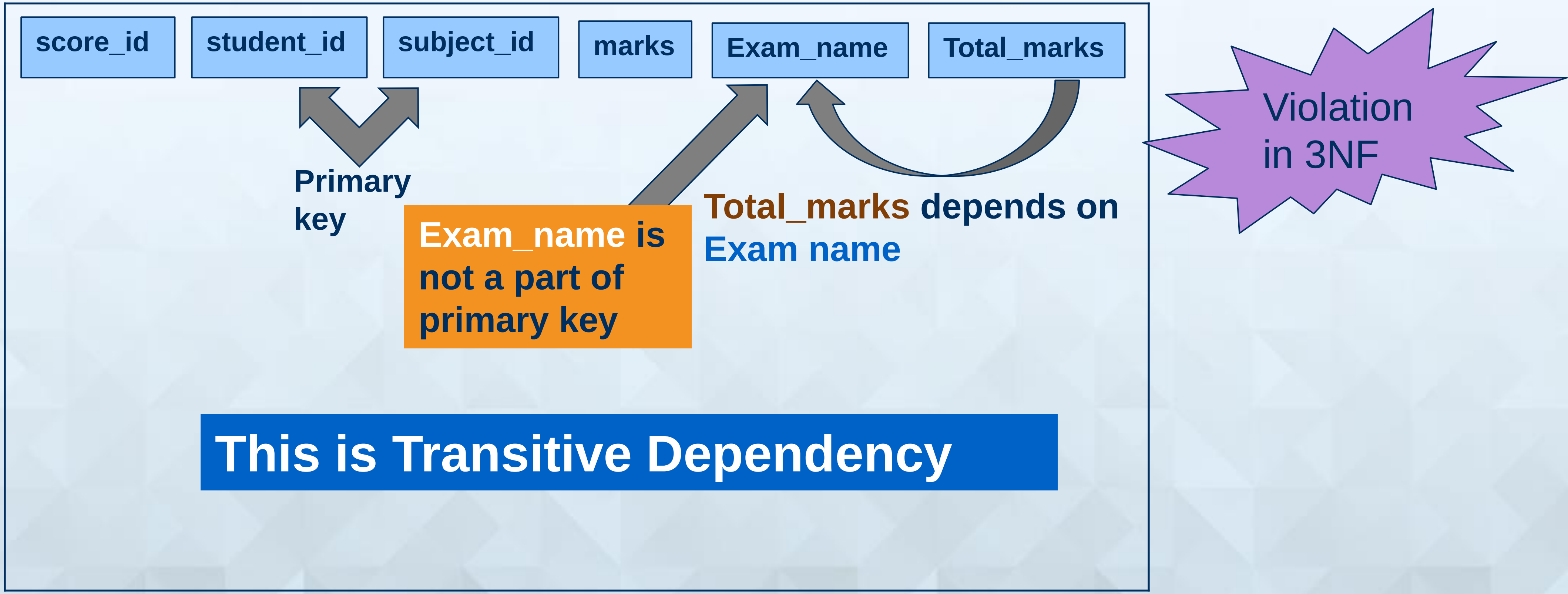
Teacher Table

teacher_id	Teacher
1	Mr. Ram
2	Mr. John
3	Mr. Kentt
4	Mr. James

3NF- Third Normal Form

- Rule 1- Be in 2NF
- Rule 2- Has no transitive functional dependencies

Score Table



score_id	student_id	subject_id	marks	Exam_name	Total_marks

Primary key

Exam_name is not a part of primary key

Total_marks depends on Exam name

Violation in 3NF

This is Transitive Dependency

3NF- Third Normal Form

Score Table

score_id	student_id	subject_id	marks	exam_name	total_marks
1	1	1	30	Practical	40
2	1	1	50	Theory	70

The transitive dependent **total_marks** column is removed from score table

Third Normal Form (3NF)

Score Table

score_id	student_id	subject_id	marks	exam_name
1	1	1	30	Practical
2	1	1	50	Theory

Exam Table

exam_name	total_marks
Practical	40
Theory	70

There are no transitive functional dependencies, and hence our table is in 3NF

Video on Third Normal Form

Objective:

To make the Trainee understand the concept of Third Normal Form.

Video Path:

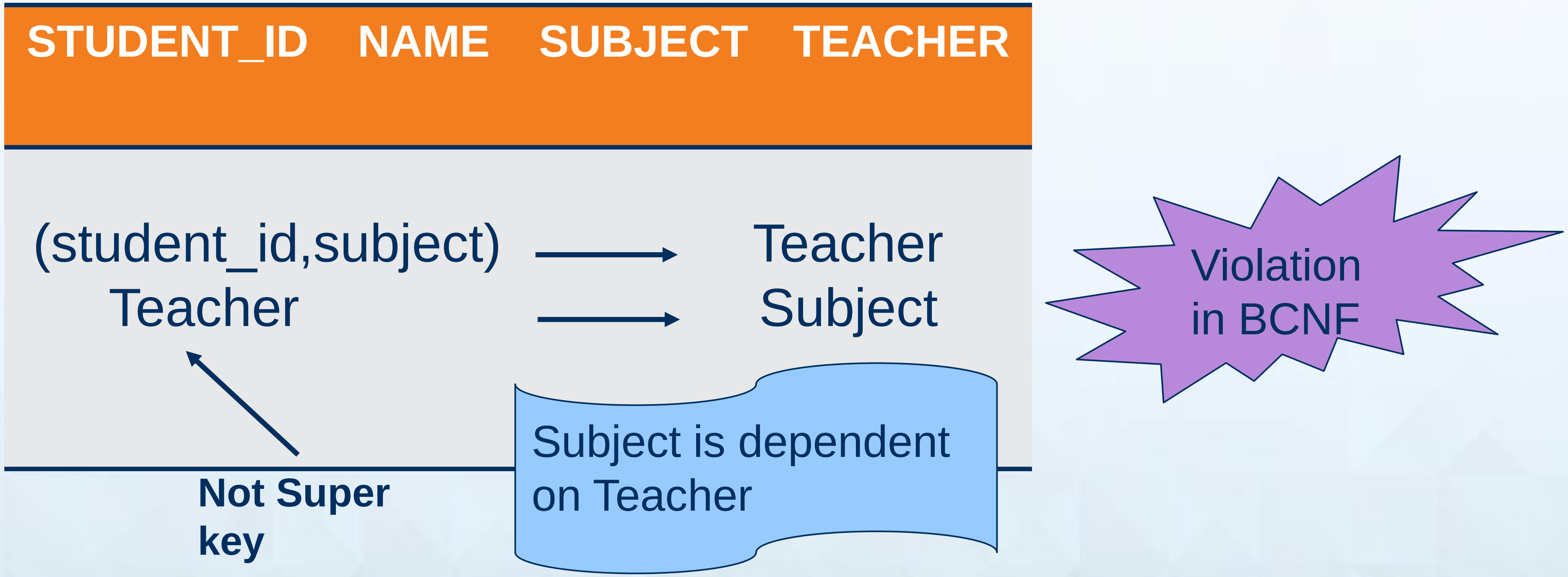
https://www.youtube.com/watch?v=GP_RcibUicQ

BCNF (Boyce Codd Normal Form)

- It should be in the 3rd Normal Form
- For any dependency A -> B, A should be a super key.
- For example,

STUDENT_ID	SUBJECT	TEACHER
1	SQL	Mr. Ram
2	C	Mr. John
2	JAVA	Mr. Ram
3	DS	Mr. John
3	OS	Mr. Kentt

BCNF (Boyce Codd Normal Form)



STUDENT_ID	NAME	SUBJECT	TEACHER

(student_id,subject) → Teacher

Teacher → Subject

Not Super key

Subject is dependent on Teacher

Violation in BCNF

- **This tables satisfies 3NF. but violation in BCNF.**
- **Subject is dependent on teacher, but teacher is not super key. So the table is not satisfying BCNF.**

BCNF (Boyce Codd Normal Form)

Student Table

STUDENT_ID	SUBJECT	TEACHER
1	SQL	Mr. Ram
2	C	Mr. John
2	JAVA	Mr. Ram
3	DS	Mr. John

Boyce codd Normal Form (BCNF)

Student Table

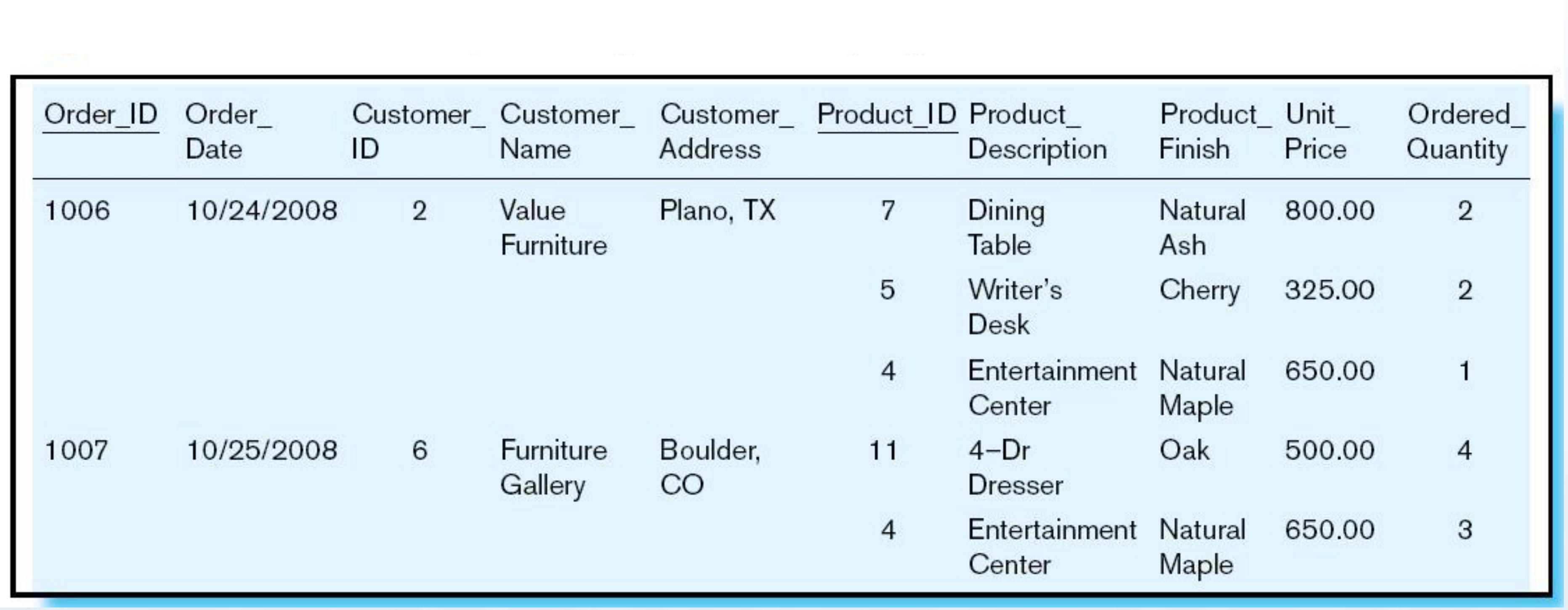
student_id	teacher_id
1	1
2	2

Teacher Table

Teacher_id	Teacher_name	Subject
1	Mr. Ram	SQL
2	Mr. John	C

Case Study on Normalization

Consider the relation INVOICE,

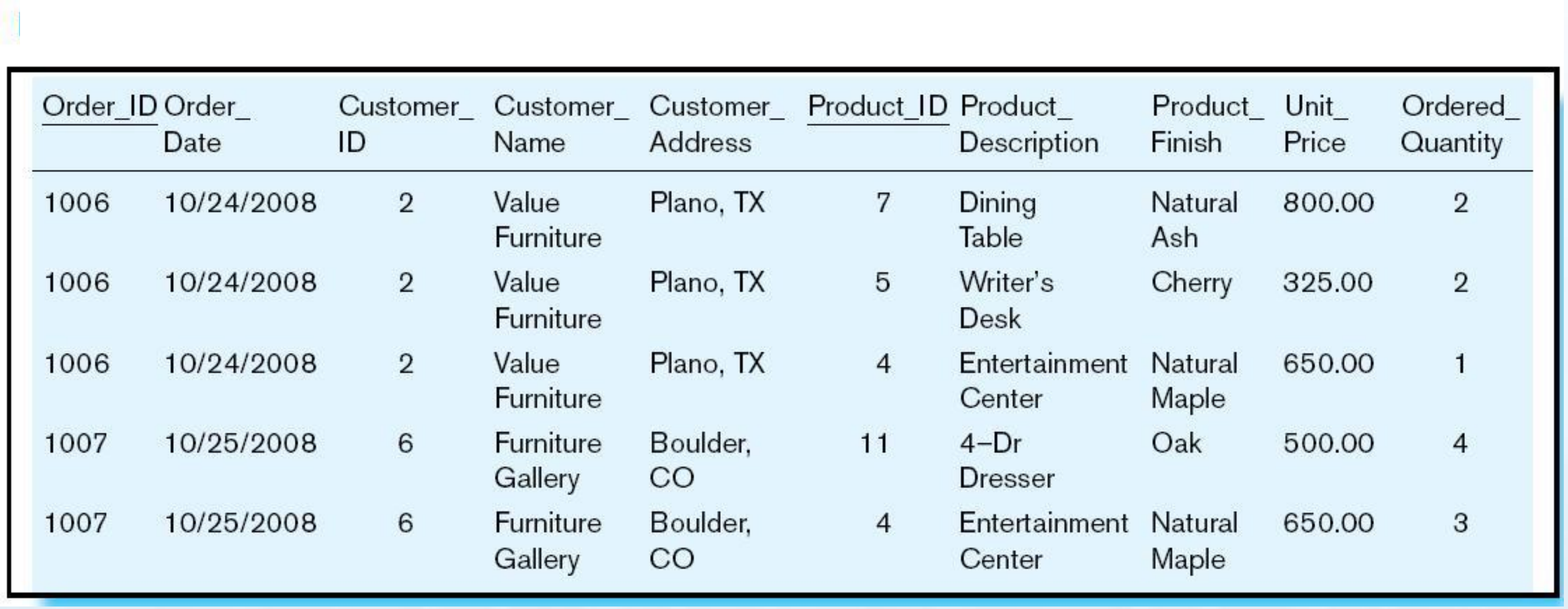


Order_ID	Order_Date	Customer_ID	Customer_Name	Customer_Address	Product_ID	Product_Description	Product_Finish	Unit_Price	Ordered_Quantity
1006	10/24/2008	2	Value Furniture	Plano, TX	7	Dining Table	Natural Ash	800.00	2
					5	Writer's Desk	Cherry	325.00	2
					4	Entertainment Center	Natural Maple	650.00	1
1007	10/25/2008	6	Furniture Gallery	Boulder, CO	11	4–Dr Dresser	Oak	500.00	4
					4	Entertainment Center	Natural Maple	650.00	3

- The above relation is in Un-normalized state.

Case Study on Normalization contd…

- After 1NF, (multiple values are present, which violates 1NF)



Order_ID	Order_ Date	Customer_ ID	Customer_ Name	Customer_ Address	Product_ID	Product_ Description	Product_ Finish	Unit_ Price	Ordered_ Quantity
1006	10/24/2008	2	Value Furniture	Plano, TX	7	Dining Table	Natural Ash	800.00	2
1006	10/24/2008	2	Value Furniture	Plano, TX	5	Writer's Desk	Cherry	325.00	2
1006	10/24/2008	2	Value Furniture	Plano, TX	4	Entertainment Center	Natural Maple	650.00	1
1007	10/25/2008	6	Furniture Gallery	Boulder, CO	11	4–Dr Dresser	Oak	500.00	4
1007	10/25/2008	6	Furniture Gallery	Boulder, CO	4	Entertainment Center	Natural Maple	650.00	3

- The above table in 1NF, because no multi-value attributes.

Case Study on Normalization contd…

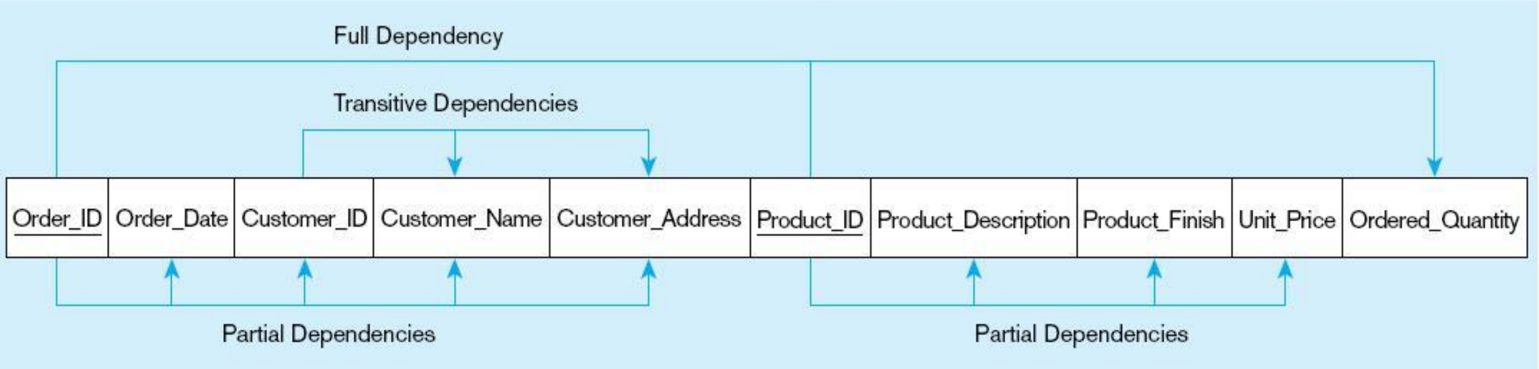
- Based on the information (table) details, list all the FDs

Order_ID ➔ Order_Date, Customer_ID, Customer_Name, Customer_Address

Customer_ID ➔ Customer_Name, Customer_Address

Product_ID ➔ Product_Description, Product_Finish, Unit_Price

Order_ID, Product_ID ➔ Order_Quantity

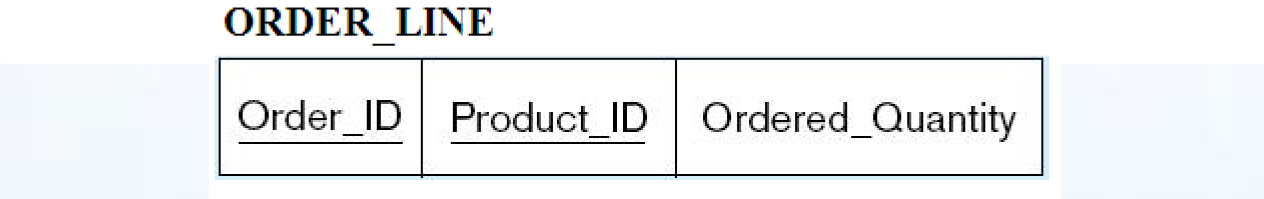


Order_ID	Order_Date	Customer_ID	Customer_Name	Customer_Address	Product_ID	Product_Description	Product_Finish	Unit_Price	Ordered_Quantity

- The FDs, which violates 2NF and 3NF are mentioned

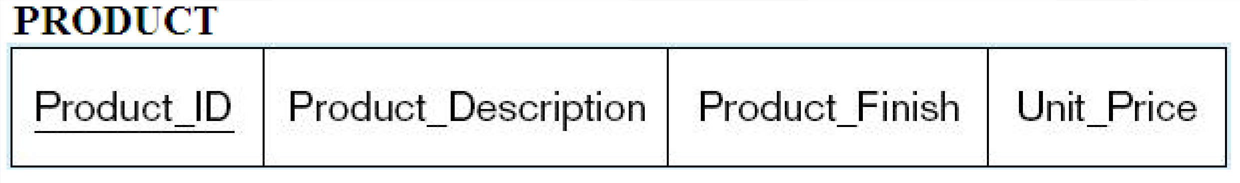
Case Study on Normalization contd…

After removing the partial and transitive dependencies,



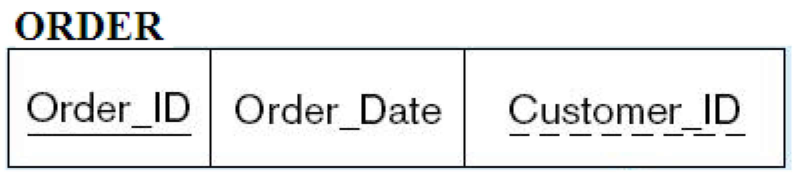
ORDER_LINE

Order_ID	Product_ID	Ordered_Quantity



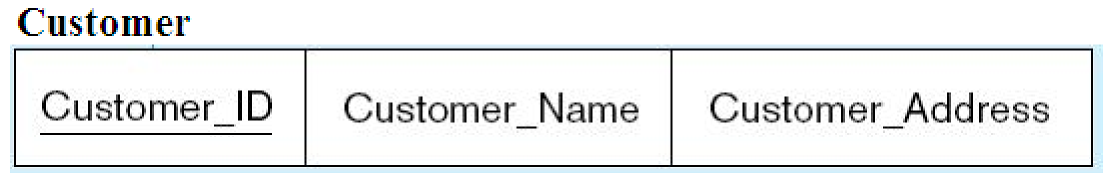
PRODUCT

Product_ID	Product_Description	Product_Finish	Unit_Price



ORDER

Order_ID	Order_Date	Customer_ID



Customer

Customer_ID	Customer_Name	Customer_Address

The above relations are in 3NF

Additional Resources

Objective:

To make the participants to get more practice with additional Example.

Normalaization -
dditional Resourc

HEXAWARE

Innovative Services

Passionate Employees

Delighted Customers

Thank you

www.hexaware.com