CM1603 - Database Systems

Week 07| Data Normalization

Dileeka Alwis – Senior Lecturer Grade II / Level Coordinator,

Department of Computing, IIT

ROBERT GORDON UNIVERSITY ABERDEEN
TEF Gold
INFORMATICS INSTITUTE OF TECHNOLOGY

Learning Outcomes

- Covers LO1 for Module - Describe and evaluate underlying theory and principles of relational database management systems (RDBMS).

- Covers LO2 for Module – Analyses and apply database design and modelling methods for a given business case study

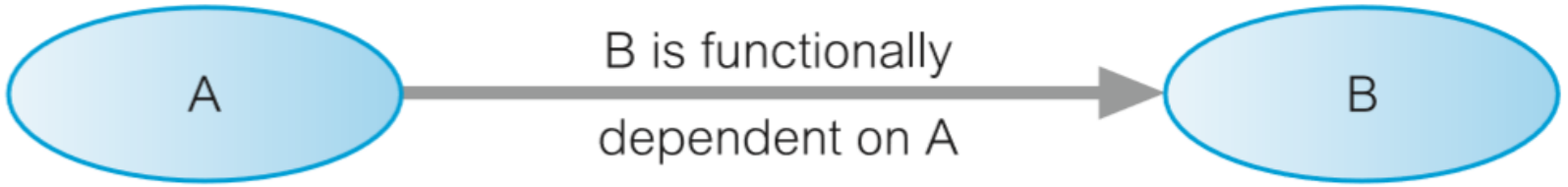
- On completion of this lecture, students are expected to be able to:
  - Identify issues related with relations
  - Know normalization steps
  - Normalize relational schema

Lesson Outline

- Functional Dependency
  - Full Functional Dependency vs. Partial Dependency
- Introduction to Data Normalization
- Data Anomalies
  - Insert Anomaly
  - Update Anomaly
  - Delete Anomaly
- Data Normalization Forms
  - 0NF, 1NF, 2NF, 3NF

Functional Dependency

- It is a relationship between attributes, where the value of attribute ‘A’ determines the value for attribute ‘B’.

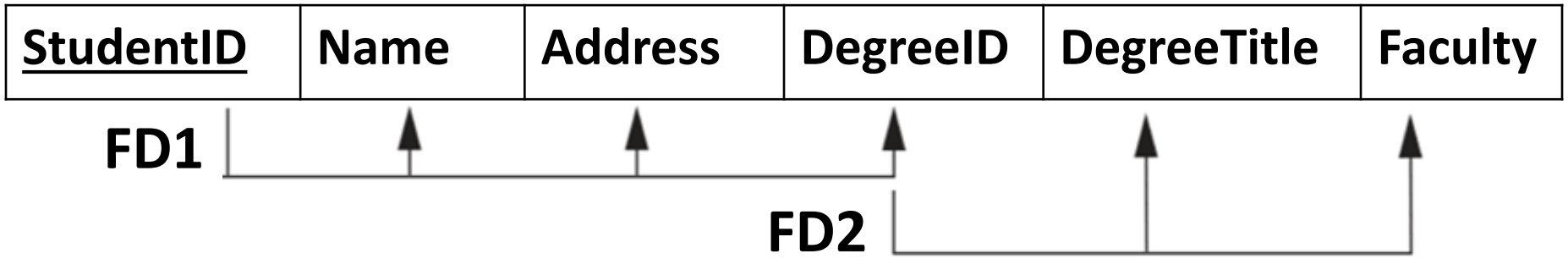


‘B’ is functionally depend on ‘A’.

‘A’ functionally determines ‘B’.

‘A’ is called as the **Determinant**.

Example



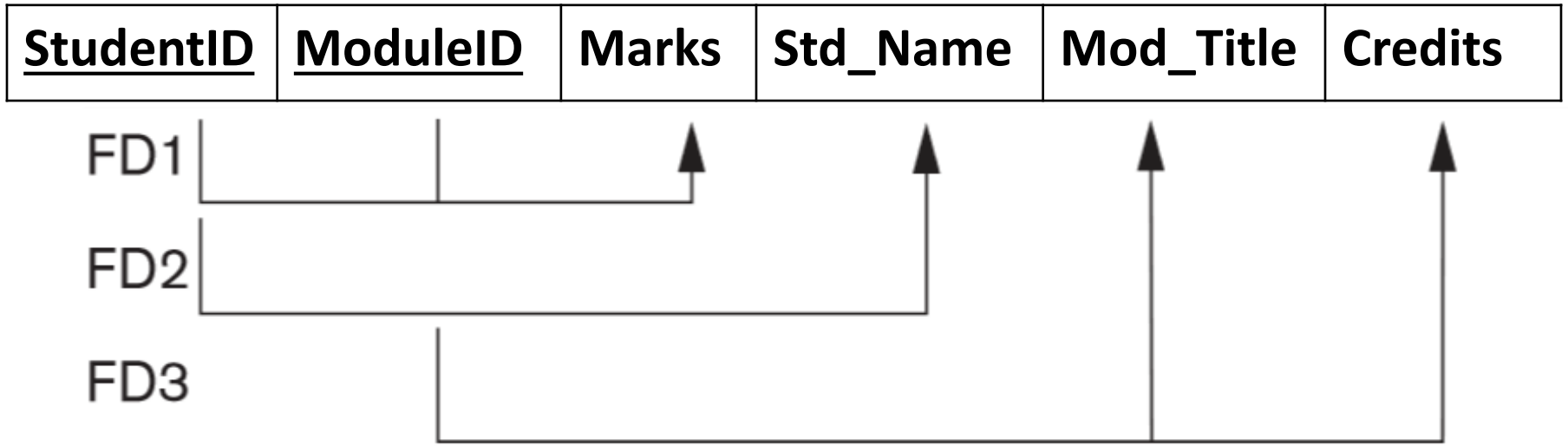
- FD1

  StudentID → {Name, Addres, DegreeID}

- FD2

  DegreeID → {DegreeTitle, Faculty}

Example



- FD1 {StudentID, ModuleID} → Marks
- FD2 StudentID → Std_Name
- FD3 ModuleID → {Mod_Title, Credits}

Fully Functional Dependency

- ‘B’ is fully functionally dependent on ‘A’,
  if it is functionally dependent on ‘A’
  but not any part of ‘A’.

Eg:

StudentId, ModuleCode ⟶ Marks

SupplierID, ProductID ⟶ Quantity

Full Dependency & Partial Dependency

- {StudentID, ModuleID} → Marks
  - Full functional dependency
- StudentID → Std_Name
  - Partial dependency
- ModuleID → {Mod_Title, Credits}
  - Partial dependency

Transitive Dependency

- If x, y and z are attributes, and if

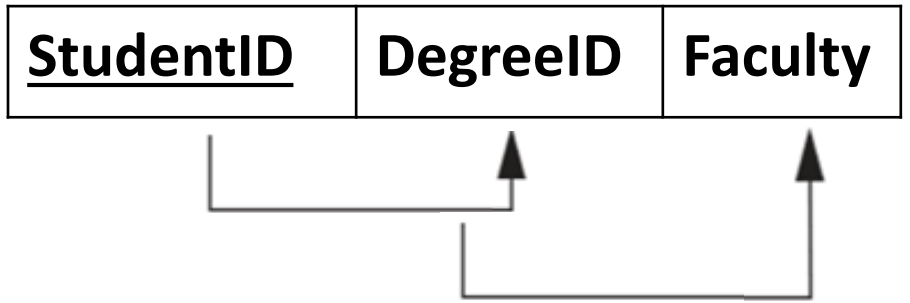
$$x \longrightarrow y$$

$$y \longrightarrow z$$

- Then ‘z’ is transitively dependent on x.

$$x \longrightarrow z$$

Example



- StudentID → DegreeID
- DegreeID → Faculty

- Therefore StudentID → Faculty

  Transitive dependency

Data Normalization

- Normalisation is a data modelling technique used to decompose unsatisfactory relations (tables that are not according to the relational model) into **a set of well-structured relations**.

- Work through a series of stages, called normal forms:
  - First normal form (1NF), second normal form (2NF), third normal form (3NF), etc.

- It helps to
  - eliminate data redundancy.
  - organize data efficiently and maintain relationships between relations.
  - perform different data operations (insert/update/delete) without producing anomalies, errors or inconsistencies.

Insert Anomaly

- An insertion anomaly occurs when to add a new row into a table requires duplication of data that already exists.

- Eg: Cannot record a new sales office until a new salesperson is added to the table because EmployeeID is the primary key of the table.

EmployeeID	SalesPerson	SalesOffice	OfficeNumber	Customer1	Customer2	Customer3
1003	Mary Smith	Chicago	312-555-1212	Ford	GM	
1004	John Hunt	New York	212-555-1212	Dell	HP	Apple
1005	Martin Hap	Chicago	312-555-1212	Boeing		
???	???	Atlanta	312-555-1212			

Update Anomaly

- An update anomaly occurs when the same data is stored repeatedly in one table, and hence any updates to the data requires multiple changes.

- Eg: If the office number changes, then multiple updates need to be made.

EmployeeID	SalesPerson	SalesOffice	OfficeNumber	Customer1	Customer2	Customer3
1003	Mary Smith	Chicago	312-555-1212	Ford	GM	
1004	John Hunt	New York	212-555-1212	Dell	HP	Apple
1005	Martin Hap	Chicago	312-555-1212	Boeing		

Delete Anomaly

- A deletion anomaly occurs when the deletion of a single piece of data results in a loss of valid data on the same row.

- Eg: Deletion of a Sales Office will cause a deletion of Employee and Customer data as well.

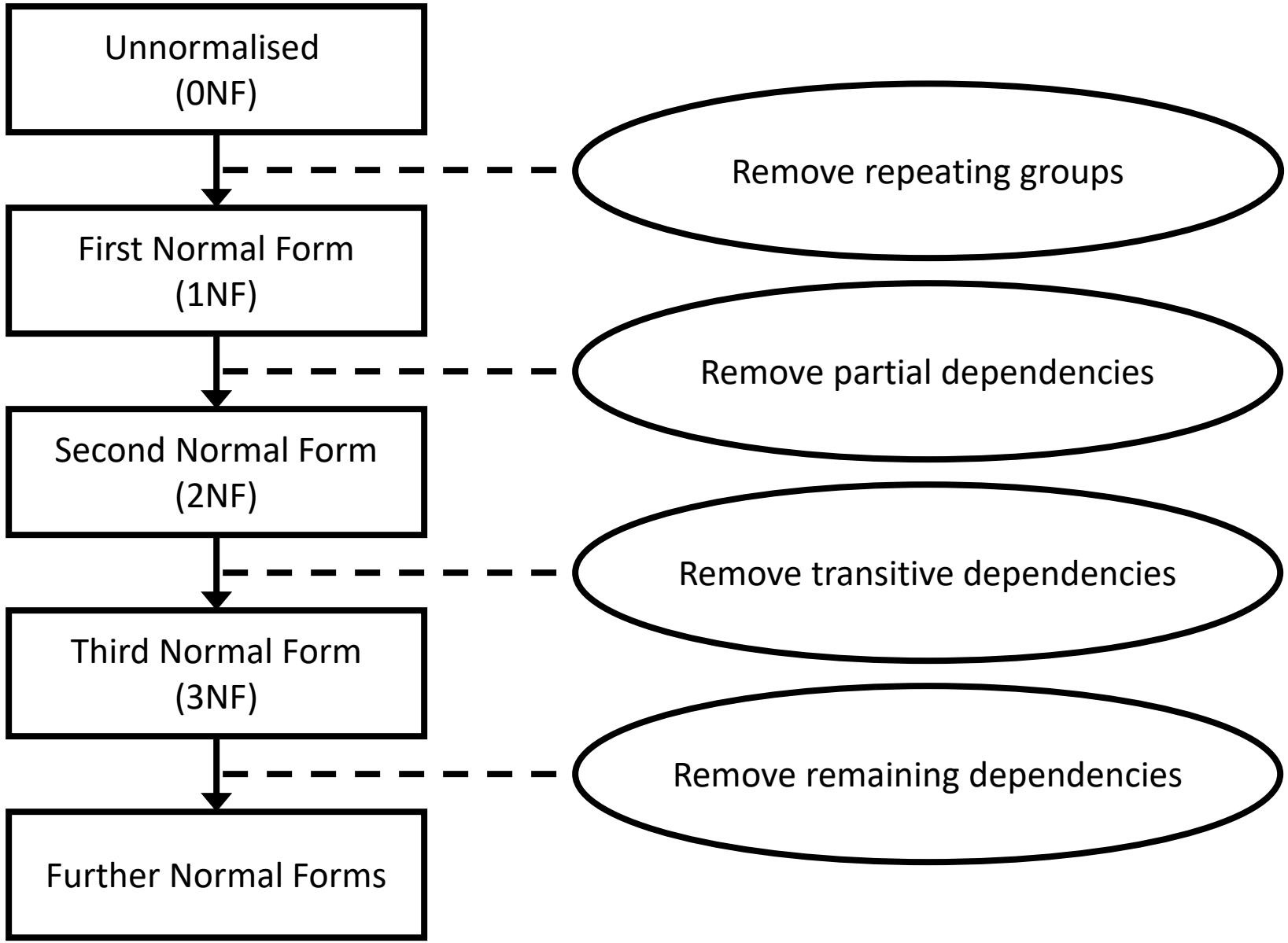
EmployeeID	SalesPerson	SalesOffice	OfficeNumber	Customer1	Customer2	Customer3
1003	Mary Smith	Chicago	312-555-1212	Ford	GM	
~~1004~~	~~John Hunt~~	~~New York~~	~~212-555-1212~~	~~Dell~~	~~HP~~	~~Apple~~
1005	Martin Hap	Chicago	312-555-1212	Boeing		

Normalization Forms

- First Normal Form (1NF)
- Second Normal Form (2NF)
- Third Normal Form (3NF)

- For a correctly drawn ER/EER diagram and a correct mapping according to the relational model, the final relations are in the **3NF**.

Stages of Normalisation



First Normal Form (1NF)

- A relation R is in first normal form (1NF) if
  - domains of all attributes in the relation are atomic (simple & indivisible).
  - avoid multi valued & composite attributes.
  - remove repeating groups into a new relation.
- The attributes must be atomic and single valued.
- If a particular relation is not in 1NF, must decompose it into relevant relations and add the PK of the original relation.

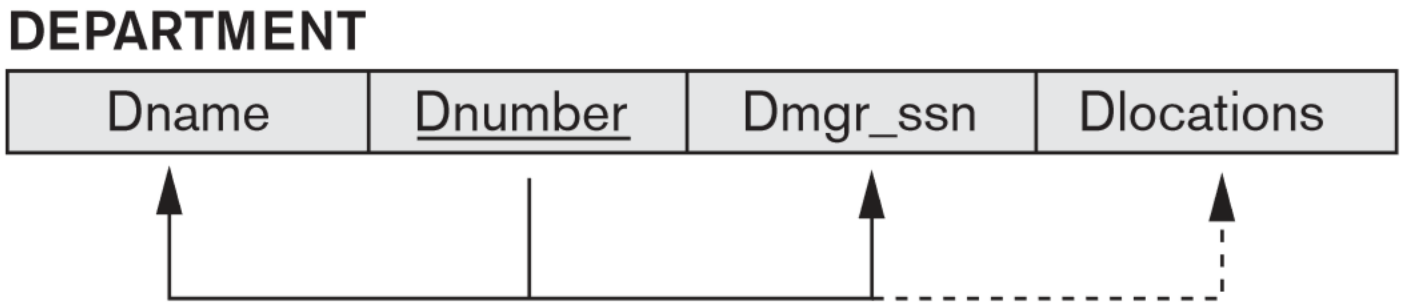
Steps from 0NF to 1NF:

- Remove the outermost repeating group and create a new relation.
- Add a copy of the PK of the original relation into the new relation.
- Name the new relation.
- Determine the PK of the new relation.
- Repeat steps until no more repeating groups.

Example 1

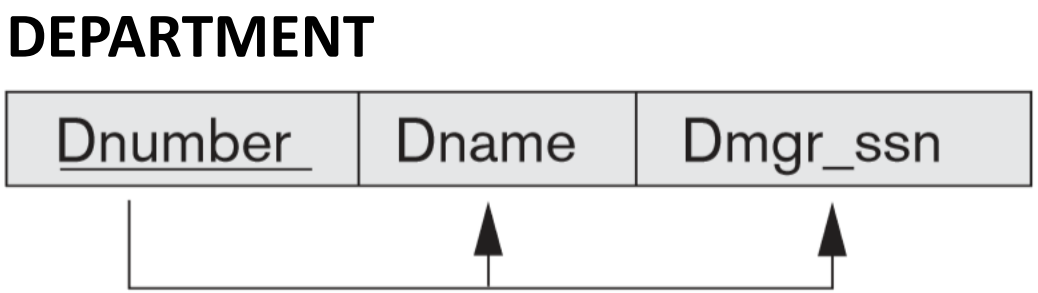
Dname	Dnumber	Dmgr_ssn	Dlocations
Research	5	333445555	{Bellaire, Sugarland, Houston}
Administration	4	987654321	{Stafford}
Headquarters	1	888665555	{Houston}

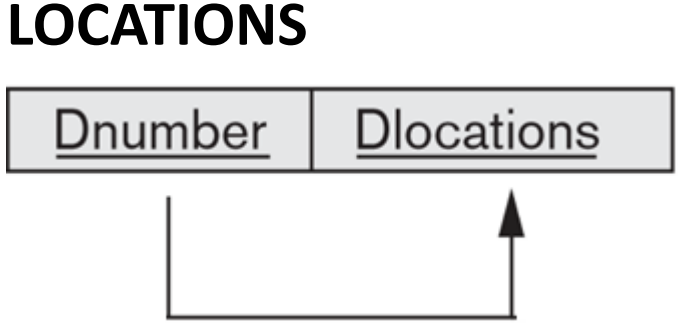
Dname	Dnumber	Dmgr_ssn	Dlocation
Research	5	333445555	Bellaire
Research	5	333445555	Sugarland
Research	5	333445555	Houston
Administration	4	987654321	Stafford
Headquarters	1	888665555	Houston



- Relation DEPARTMENT is **not** in 1NF.

- According to 1NF:





Second Normal Form (2NF)

- Based on the concept of fully functional dependency.
- A relation that is in 1NF and every non-primary key attribute is fully functionally dependent on the primary key.
- 2NF involves the removal of partial dependencies.
- If a partial dependency exists, remove the partially dependent attribute(s) from the relation by placing them in a new relation along with a copy of their determinant.
- The determinant will be the PK of the new relation.

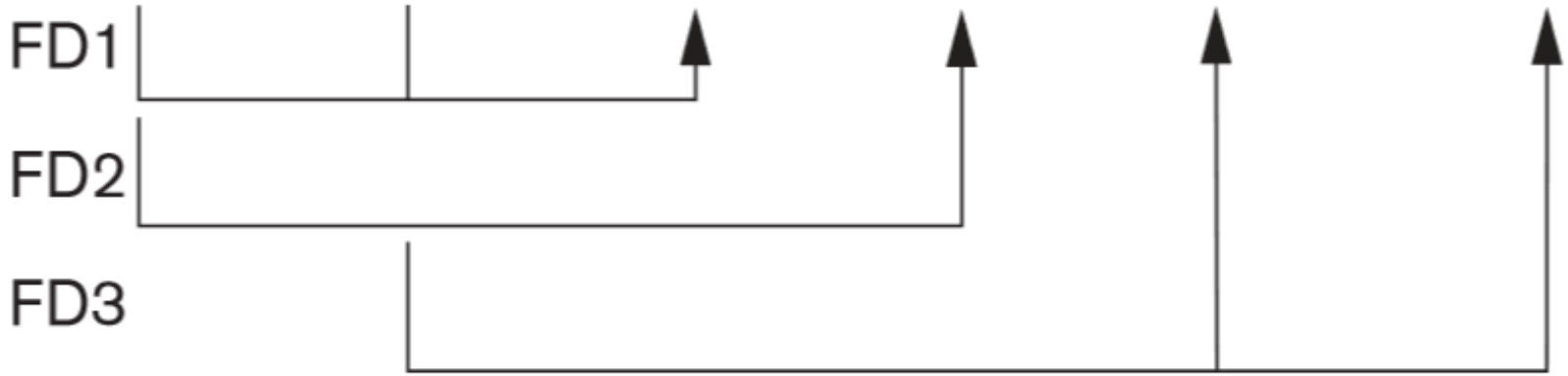
Steps from 1NF to 2NF:

- Remove the offending attributes that are only partially functionally dependent on the composite key and place them in a new relation.
- Add to this relation a copy of the attribute(s) which are the determinants of these offending attributes. These will automatically become the primary key of this new relation.
- Name the new relation.
- Rename the original relation.

Example

Student_Module

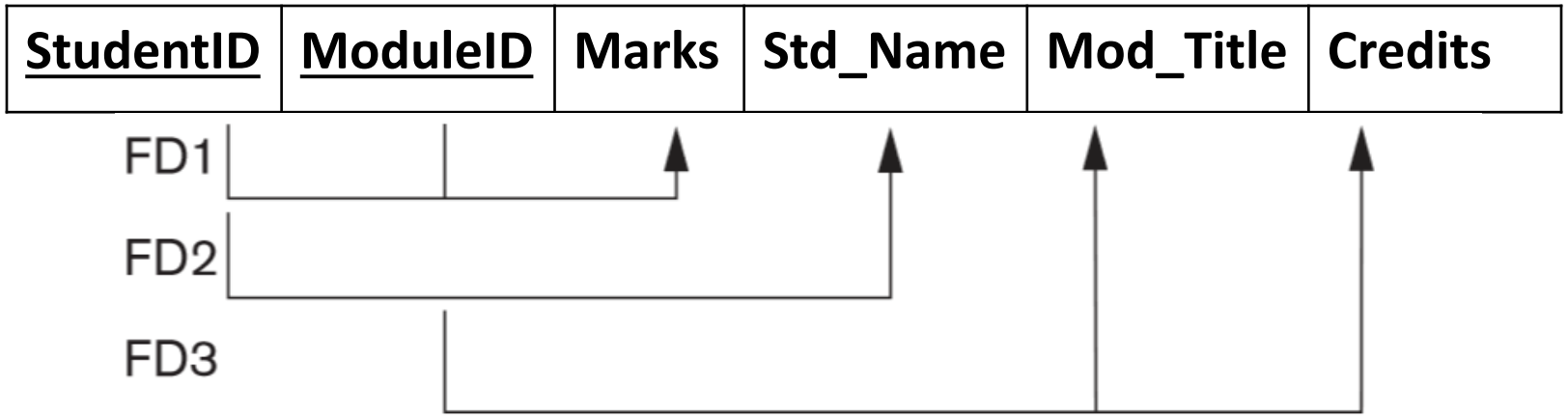
StudentID	ModuleID	Marks	Std_Name	Mod_Title	Credits



- This relation is in 1NF but is not in 2NF.
- The functional dependencies FD2 and FD3 partially dependent on the primary key {StudentID, ModuleID}

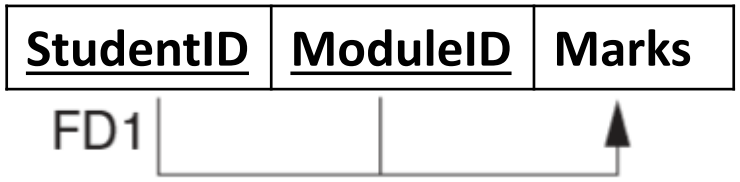
According to 1NF

Student_Module



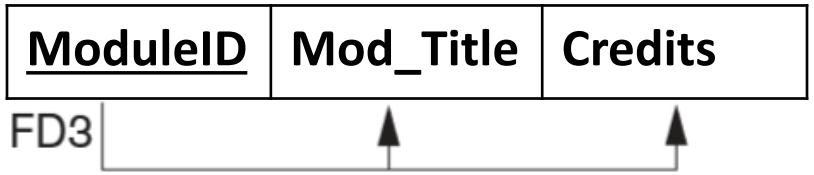
According to 2NF

Marks



Student

Module



Third Normal Form (3NF)

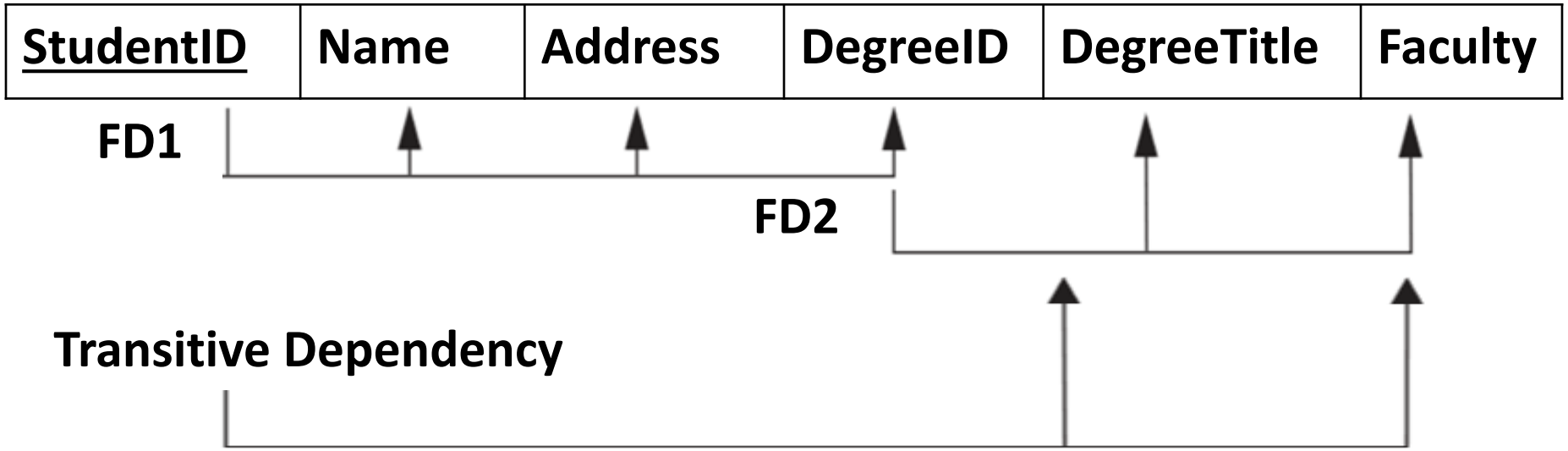
- 3NF is based on the concept of transitive dependency.
- A relational schema is in 3NF only if it is in 2NF and every non key attribute is non transitively dependent on the PK.
- If a transitive dependency exists, remove the transitively dependent attribute(s) from the relation by placing the attribute(s) in a new relation along with a copy of the determinant.
- The determinant will be the PK of the new relation.

Steps from 2NF to 3NF:

- Remove the offending attributes that are transitively dependent on non-key attribute(s) and place them in a new relation.
- Add to this relation a copy of the attribute(s) which are the determinants of these offending attributes. These will automatically become the primary key of this new relation.
- Name the new relation.
- Rename the original relation.

Example

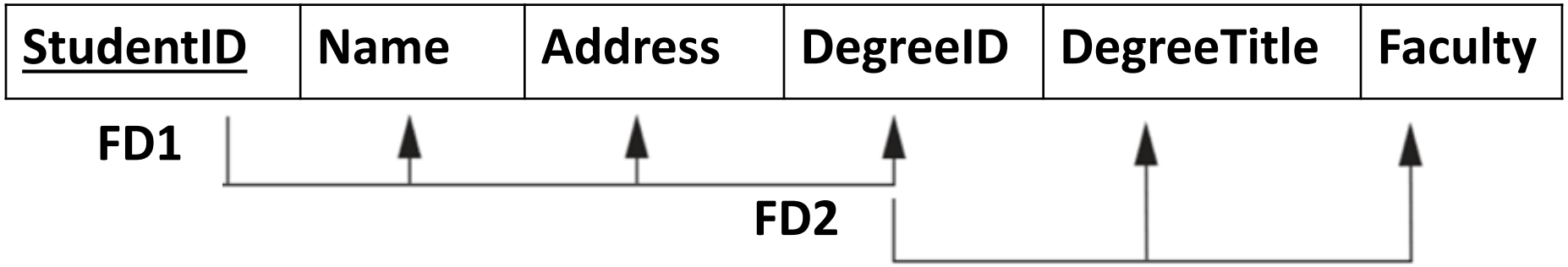
Student_Degree



- This relation is in 2NF but is not in 2NF.
- The functional dependency FD2 is transitively dependent on {StudentID}

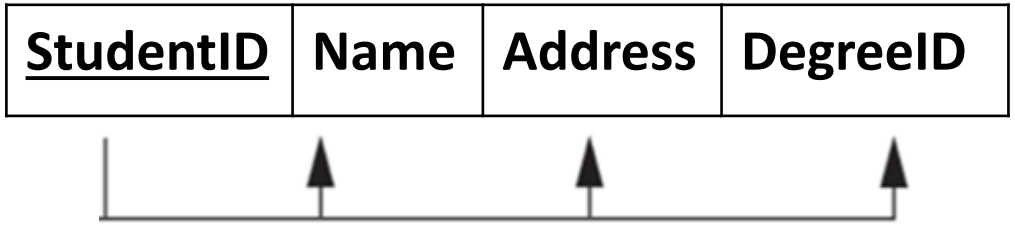
According to 2NF

Student_Degree

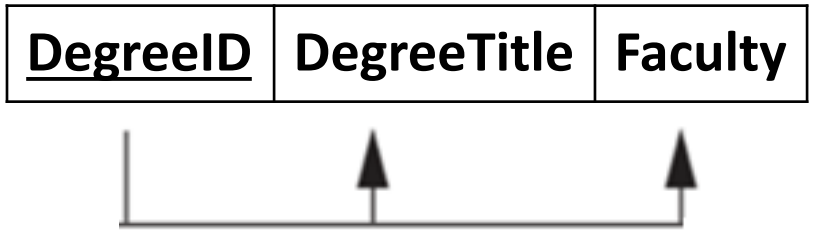


According to 3NF

Student



Degree



Exercise

Sales Invoice

InvoiceNo : 254186
OrderDate: 29/6/09

CustomerNo: 78901
CustomerName: Fred Bloggs
Customer PhoneNo: 9370 6111
Customer Address: 3 Uphill Rise, Ferndale, WA 6303

ItemNo	Description	Qty	UnitPrice	Subtotal
9898	Bearing, Ball	25	$2.50	$62.50
9999	Bearing, Roller	10	$5.00	$50.00
8888	Seal, shaft	10	$3.00	$30.00
777	Glasses, Safety	10	$10.00	$100.00
1555	Punch, 5mm	1	$4.00	$4.00

Total: $246.50

0NF:

R1 = (InvoiceNo, OrderDate, CustomerNo, Name, Phone, Address, {ItemNo, Description, Qty, UnitPrice})

0NF to 1NF

0NF

R1 = (InvoiceNo, OrderDate, CustomerNo, Name, Phone, Address, {ItemNo, Description, Qty, UnitPrice})

1NF

R11 = (InvoiceNo, OrderDate, CustomerNo, Name, Phone, Address)

R12 = (InvoiceNo, ItemNo, Description, Qty, UnitPrice)

1NF to 2NF

1NF

R11 = (InvoiceNo, OrderDate, CustomerNo, Name, Phone, Address)

R12 = (InvoiceNo, ItemNo, Description, Qty, UnitPrice)

2NF

R11 = (InvoiceNo, OrderDate, CustomerNo, Name, Phone, Address)

R121 = (InvoiceNo, ItemNo, Qty)

R122 = (ItemNo, Description, UnitPrice)

2NF to 3NF

2NF

R11 = (InvoiceNo, OrderDate, CustomerNo, Name, Phone, Address)

R121 = (InvoiceNo, ItemNo, Qty)

R122 = (ItemNo, Description, UnitPrice)

3NF

R111 = (InvoiceNo, OrderDate, CustomerNo)

R112 = (CustomerNo, Name, Phone, Address)

R121 = (InvoiceNo, ItemNo, Qty)

R122 = (ItemNo, Description, UnitPrice)

Final Relational Schema

- Invoice (InvoiceNo, OrderDate, CustomerNo)
- Customer (CustomerNo, Name, Phone, Address)
- Invoice_Details (InvoiceNo, ItemNo, Qty)
- Items (ItemNo, Description,UnitPrice)

Thank you

Contact: dileeka.a@iit.ac.lk

ROBERT GORDON UNIVERSITY ABERDEEN
TEF Gold
INFORMATICS INSTITUTE OF TECHNOLOGY