Official (Open)

**Objectives of this practical**

* Normalize database tables to 1NF, 2NF and 3NF

Practical 3 – Normalization

Contents

[Normalization 3](#_bookmark0)

[Question 1 3](#_bookmark1)

[Question 2 4](#_bookmark2)

## Normalization

# Question 1

You are given the following student\_course\_module table containing data as shown:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **adm\_no** | **stud\_name** | **crse\_cd** | **crse\_name** | **mod\_cd** | **mod\_name** | **mark** |
| A001 | Janice | DIT | Diploma in Information Technology | SC | Secure Coding | 83 |
| DBS | Database Systems | 72 |
| FOC | Fundamentals of  Computing | 78 |
| BED | Back-end Development | 87 |
| A002 | Anita | DAAA | Diploma in AI and Analytics | DL | Deep Learning | 87 |
| DENG | Data Engineering | 83 |
| FOC | Fundamentals of Computing | 78 |
| BED | Back-end Development | 87 |
| … | … | … | … | … | … | … |

1. Is the student\_course\_module table in 0NF? Justify your answer.

Yes. The table has multi-valued cells

1. Write the table in the relational heading format.

Student\_course\_module(adm\_no, stud\_name, crse\_cd, crse\_name, {mod\_cd, mod\_name, mark})

1. What is a first normal form (1NF) table? Transform the table, if it is not already in the 1NF, into the first normal form. Present your 1NF table in
   1. a table form as shown above

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **adm\_no** | **stud\_name** | **crse\_cd** | **crse\_name** | **mod\_cd** | **mod\_name** | **mark** |
| A001 | Janice | DIT | Diploma in Information Technology | SC | Secure Coding | 83 |
| A001 | Janice | DIT | Diploma in Information Technology | DBS | Database Systems | 72 |
| A001 | Janice | DIT | Diploma in Information Technology | FOC | Fundamentals of  Computing | 78 |
| A001 | Janice | DIT | Diploma in Information Technology | BED | Back-end Development | 87 |
| A002 | Anita | DAAA | Diploma in AI and Analytics | DL | Deep Learning | 87 |
| A002 | Anita | DAAA | Diploma in AI and Analytics | DENG | Data Engineering | 83 |
| A002 | Anita | DAAA | Diploma in AI and Analytics | FOC | Fundamentals of Computing | 78 |
| A002 | Anita | DAAA | Diploma in AI and Analytics | BED | Back-end Development | 87 |
| … | … | … | … | … | … | … |

* 1. relational heading format

student\_course\_module(adm\_no, stud\_name, crse\_cd, crse\_name, mod\_cd, mod\_name, mark)

PK – adm\_no, mod\_cd

1. Using the 1NF table student\_course\_module, explain, what is insert, update and delete anomaly.

Insert anomaly – trying to insert a row when one of the value is empty.

Update anomaly – Requires update in several rows

Delete anomaly – Loss of information when rows are deleted from the relation

1. Transform the 1NF relation of student\_course\_module into a set of 2NF relations.

|  |  |  |
| --- | --- | --- |
| Functional Dependencies | Relations (Relational Header Format) | Primary Key |
| Adm\_no -> stud\_name | Student (adm\_no, stud\_name) | Adm\_no |
| Crse\_cd -> crse\_name, mod\_cd, mod\_name | Module (crse\_cd, crse\_name, mod\_cd, mod\_name) | Crse\_cd |
| Adm\_no, crse\_cd -> mark | Student\_course\_module (adm\_no, crse\_cd, mark) | Adm\_no, crse\_cd |

1. What is a 2NF table?

Each table have its own functional dependencies

1. Transform the set of 2NF relations of into a set of 3NF relations.

|  |  |  |
| --- | --- | --- |
| Relations (Relational Header Format) | Transitive Dependency? | Transformation |
| Adm\_no -> stud\_name | No |  |
| Crse\_cd -> crse\_name, mod\_cd, mod\_name | Yes  Mod\_cd -> mod\_name | New relation:  Module (mod\_cd, mod\_name)  Existing relation:  Course (crse\_cd, crse\_name, mod\_cd)  Mod\_cd become foreign key |
| Adm\_no, crse\_cd -> mark | No |  |

# Question 2

The following table stores the project charges of a software house:

project\_charges

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **p\_no** | **p\_name** | **e\_no** | **e\_name** | **job\_type** | **man\_day\_ rate** | **man\_day\_ billed** | **total\_ charge** |
| 102 | VesselSoft | 565 | Tan | PM | 1000 | 5 | 5,000 |
| 798 | Lim | PL | 800 | 20 | 16,000 |
| 885 | Gay | SE | 400 | 50 | 20,000 |
| 201 | Soft Machine | 565 | Tan | PM | 1000 | 4 | 4,000 |
| 698 | Lin | PL | 800 | 10 | 8,000 |
| 888 | Sia | SE | 400 | 100 | 40,000 |
| 555 | Chan | Prog | 200 | 100 | 20,000 |

Legend

|  |  |
| --- | --- |
| p\_no | Project Number which uniquely identifies a project |
| p\_name | Project Name |
| e\_no | Employee Number which uniquely identifies an employee |
| e\_name | Employee Name |
| Job\_type | Job designation held by an employee |
| man\_day\_rate | Rate charged per day for a specific job type |
| man\_day\_billed | Number of days to be billed for an employee working in a project |
| total\_charge | Total amount charged for an employee in a project |

* Each employee can only hold one job\_type.
* The man\_day\_rate is dependent on the job\_type
* The number of days an employee worked on a project is recorded in the man\_day\_billed column.

1. The following is an *incorrect* first normal form (1NF) for the above project\_charges table:

project\_charges (**p\_no**, p\_name, {**e\_no, e\_name**, job\_type, man\_day\_rate, man\_day\_billed, total\_charges})

Primary key is given as: p\_no, e\_no, e\_name

Explain the error(s) in the given 1NF table, and write the corrected 1NF table.

There shouldn’t be multi values in a cell , e\_name is not suppose to be a primary key.

project\_charges (**p\_no**, p\_name, **e\_no,** e\_name, job\_type, man\_day\_rate, man\_day\_billed, total\_charge)

pk – p\_no, e\_no

1. Derive the second normal form relation(s) from the corrected first normal form relation.

|  |  |  |
| --- | --- | --- |
| Functional Dependencies | Relations (Relational Header Format) | Primary Key |
| p\_no -> p\_name | project (p\_no, p\_name) | p\_no |
| E\_no -> e\_name, job\_type, man\_day\_rate | employee (e\_no, e\_name, job\_type, man\_day\_rate) | E\_no |
| p\_no, e\_no -> man\_daybilled, total\_charge | Student\_course\_module (adm\_no, crse\_cd, mark) | P\_no, e\_no |

1. State if the following statement is True or False:

Deletion anomalies cannot exist in second normal form tables.

False

1. Derive the third normal form relations from the second normal form relations in (c).

|  |  |  |
| --- | --- | --- |
| Relations (Relational Header Format) | Transitive Dependency? | Transformation |
| P\_no -> p\_name | No |  |
| E\_no -> e\_name, job\_type, man\_day\_rate | Yes  Job\_type-> man\_day\_rate | New relation:  job (job\_type, man\_day\_rate)  Existing relation:  employee(e\_no, e\_name, job\_type)  job\_type become foreign key |
| p\_no, e\_no -> man\_daybilled, total\_charge | No |  |