****

**BN002/BN013/BN104/BN120/BN997**

**Assessment 2: Database Design**

**Submitted by: Joseph Tierney, B00092923**

**19-December-2016**

**Declaration**

I herby certify that this material, which I now submit for assessment on the programme of study leading to the award of Ordinary Degree in Computing in the Institute of Technology Blanchardstown, is entirely my own work except where otherwise stated.

Author: Joseph Tierney Dated: 19/12/2016

**Table of Contents**

**Section 1 Normalisation 3**

Task 1 3

Task 2 3-4

Task 3 4-5

**Section 2 Database Design 6**

Task 1 6

Task 2 6

Task 3 7

# Section 1 Normalisation

**Task 1** – Normalisation is used for breaking down tables to remove redundant data. Normalisation is a multi-step process that puts data into tabular form by removing duplicated data from the relational tables. The steps taken during the normalisation process include first normal form, second normal form and third normal form.

First normal form deals with the repeating groups in a relational model. It is during this step of normalisation that the repeating groups are removed from the table and a separate table is created for each set of the related data. Second normal form involves the process of removing partial dependencies. Third normal form deals with the removal of transitive dependencies.

**Task 2** –

**Relational Model:**

Receipt(ReceiptID(PK), ReceiptDate, StaffID, StaffName, ProductName, QuantityOrdered, ProductPrice)

**First Normal Form:**

ProductName, QuantityOrdered, and ProductPrice are repeating fields.

Receipt(ReceiptID(PK), ReceiptDate, StaffID, StaffName)

Receipt\_Product(ReceiptID(PK,FK), ProductID(PK), ProductName, QuantityOrdered, ProductPrice)

**Second Normal Form:**

ProductName and ProductPrice are only dependent on ProductID

Receipt(ReceiptID(PK), ReceiptDate, StaffID, StaffName)

Receipt\_Product(ReceiptID(PK,FK), ProductID(PK, FK), QuantityOrdered)

Product(ProductID(PK), ProductName, ProductPrice)

**Third Normal Form:**

StaffName is functionally dependent on another attribute that is not the PK in the Receipt table.

Receipt(ReceiptID(PK), ReceiptDate, StaffID(FK))

Receipt\_Product(ReceiptID(PK,FK), ProductID(PK,FK), QuantityOrdered)

Product(ProductID(PK), ProductName, ProductPrice)

Staff(StaffID(PK), StaffName)

**Task 3** – CREATE TABLE staff(

StaffID INT NOT NULL,

StaffName VARCHAR(25),

PRIMARY KEY (StaffID)

);

CREATE TABLE product(

ProductID INT NOT NULL,

ProductName VARCHAR(50),

ProductPrice INT NOT NULL,

PRIMARY KEY(ProductID)

);

CREATE TABLE receipt(

RecepitID INT NOT NULL,

RecepitDate Date,

StaffID INT NOT NULL,

PRIMARY KEY(RecepitID),

CONSTRAINT receipt\_StaffID\_FK FOREIGN KEY(StaffID) REFERENCES staff(StaffID)

);

CREATE TABLE receipt\_product(

RecepitID INT NOT NULL,

ProductID INT NOT NULL,

QuantityOrdered INT,

PRIMARY KEY(RecepitID, ProductID),

CONSTRAINT recepit\_product\_RecepitID\_FK FOREIGN KEY(RecepitID) REFERENCES recepit(RecepitID),

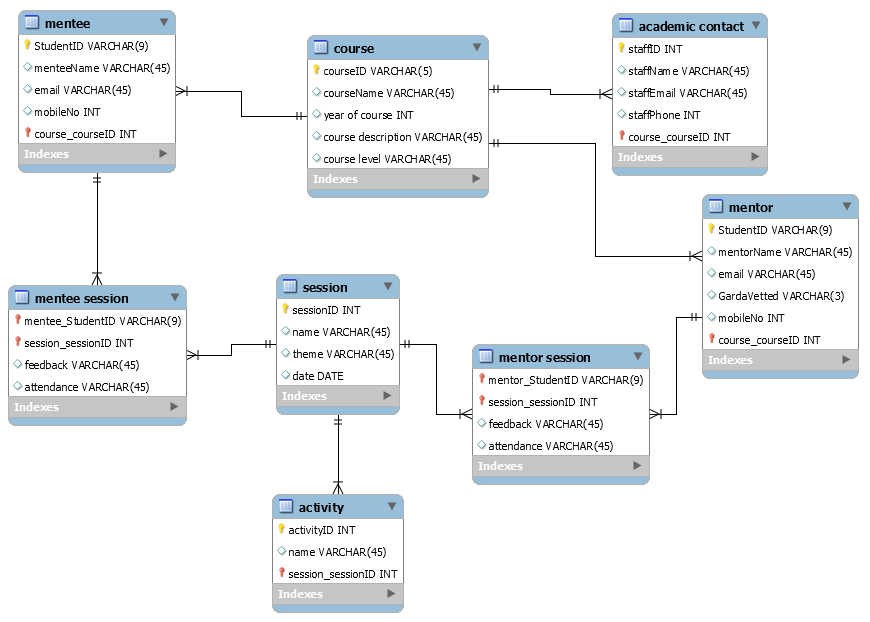
CONSTRAINT recepit\_product\_ProductID\_FK FOREIGN KEY(ProductID) REFERENCES product(ProductID)

);

Created the Staff table first due to it being a foreign key in the Receipt table, this was done to avoid errors. The order for Product and Receipt tables was done in any order, as there were no effects on the other tables until the Recepit\_Product table – this was made last due to the foreign keys coming from the product table and the receipt table.

# Section 2 Database Design

## Task 1 –



Beforehand, we assume that feedback and attendance are part of the session table. In the final ERD, these are moved to the mentee session and mentor session tables, as they are assumed to be repeating fields and are part of the many to many relationships.

**Task 2 –**

**Relational Model:**

mentee(StudentID(PK), menteeName, email, mobileNo, courseID, courseName, year of course)

mentor(StudentID(PK), mentorName, email, GardaVetted, mobileNo, courseID, courseName, year of course, course description, course level)

academic contact(staffID(PK), staffName, staffEmail, staffPhone)

session(sessionID(PK), name, theme, date, activity, feedback, attendance)

## Task 3 –

**First Normal Form:**

Session feedback and attendance are repeating fields.

session(sessionID(PK), name, theme, date, activity)

mentee\_session(studentID(PK,FK), sessionID(PK), feedback, attendance)

mentor\_session(studentID(PK,FK), sessionID(PK), feedback, attendance)

**Second Normal Form:**

activity is partially dependant on sessionID

activity(activityID(PK), name, sessionID(FK))

session(sessionID(PK), name, theme, date)

**Third Normal Form**

courseName and year of course are functionally dependent on courseID

course(courseID(PK), courseName, year of course, course description, course level)

mentee(StudentID(PK), menteeName, email, mobileNo, courseID(FK))

mentor(StudentID(PK), mentorName, email, GardaVetted, mobileNo, courseID(FK))

**Final Tables**

session(sessionID(PK), name, theme, date)

mentee\_session(studentID(PK,FK), sessionID(PK), feedback, attendance)

mentor\_session(studentID(PK,FK), sessionID(PK), feedback, attendance)

activity(activityID(PK), name, sessionID(FK))

course(courseID(PK), courseName, year of course)

mentee(StudentID(PK), menteeName, email, mobileNo, courseID(FK))

mentor(StudentID(PK), mentorName, email, GardaVetted, mobileNo, courseID(FK))

academic contact(staffID(PK), staffName, staffEmail, staffPhone, courseID(FK))