DATABASE DESIGN CONCEPTS

ASSIGNMENT

MARCH 2017

Sydney Twigg
M8C3XRSN8

CONTENTS

QUESTION 1

QUESTION 1.1

A *database* is an electronic filing system that stores data in an organised way to facilitate quick retrieval of information. A database is a collection of related data (Connoly & Begg, 2015), a database is comprised of information that is grouped together in order to easily access, manage and update the information (Rouse, 2017). Within a database, data is organised into tables, which are further organised into rows and columns - the information in the database is indexed to allow users to find relevant information (Rouse, 2017). A database is made of many tables - each table stores data related to one topic - the tables within in the database are made up of fields containing data. A database is managed using database management system (DBMS) software.

A *database management system (DBMS)* is the software that manages and controls access to the database (Connoly & Begg, 2015). A DBMS provides users with an organized way to create, retrieve, update and manage data within a database (Rouse, 2017). A DBMS acts as an interface between a database and the end users - allowing end users to create, read, update and delete data in a database - as well as ensuring that data is organised, easily accessible and in order to ensure data integrity within the database (Rouse, 2017). A DBMS also allows programmers to limit what data the end user can view and edit, as well as can hide the physical location of the database on the disk from end users which ensures data integrity and security (Rouse, 2017).

QUESTION 1.2

The costs involved in implementing a database and database management system (DBMS) are as follows:

The cost of the DBMS: This cost includes the cost of purchasing the software, the cost of installation and cost of maintenance. The cost of purchase can be estimated at R12800 ($1000) per month based on the cost of enterprise level DBMS's (Champagne, 2016). The DBMS will require yearly maintenance as well as the cost of installation and implementation which will require a qualified specialist (Apoorva College, 2011).

Hardware costs: The DBMS and database may require additional storage space in order to meet the storage requirements (Connoly & Begg, 2015). A dedicated computer may be required to run the DBMS, which will require up-to-date and high end hardware in order for the DBMS to run at optimal performance (Connoly & Begg, 2015). The database requires backup and recovery options, in order to implement these, there must be additional storage space available for backup copies of the data to restore the database if any damage occurs (Apoorva College, 2011).

Conversion costs: These costs include converting the existing applications to run on the new DBMS and hardware, as well as the cost of training staff to use the new DBMS (Connoly & Begg, 2015). Specialist staff may need to be hired to assist in the design, installation, implementation, and management of the DBMS, which brings additional costs on top of training current staff (Apoorva College, 2011) (Connoly & Begg, 2015).

QUESTION 1.3

The Systems Development Life Cycle is a methodology used in the development of information systems. The standard SDLC consists of six main stages used to develop an information system - however, these six stages can be adapted to meet the needs of specific system development - such as a database system development life cycle as shown below.

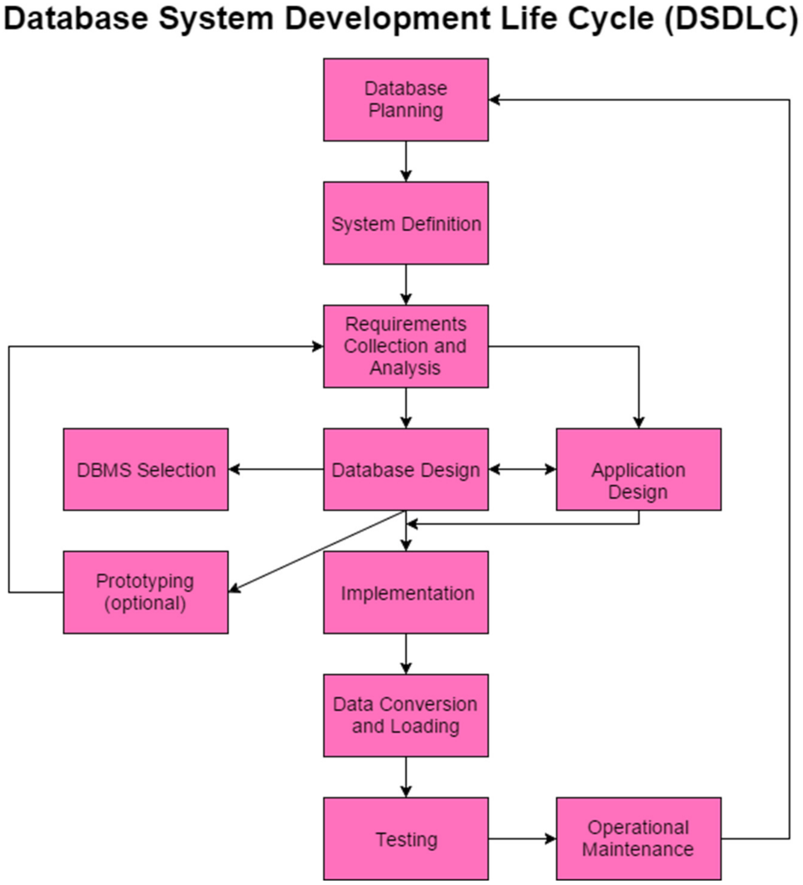


Diagram showing the Database System Development Life Cycle.

The Database System Development Life Cycle (DSDLC) can be broken down into the following stages:

Database planning: The main activity of database planning is planning how the stages of the life cycle can be achieved in the most effective and efficient way (Connoly & Begg, 2015). In this stage, the following must be defined: the aims, purpose, supported tasks and resources of the database system (Auer, 2006). Database planning should include the development of standards to govern data collection, documentation required, how design and implementation will be carried out and how the format is specified (Connoly & Begg, 2015).

System definition: The main activity of system definition is specifying the scope and boundaries of the database system (Connoly & Begg, 2015). The systems definition also describes the major views of the user - what is required of the database system from the perspective of a particular user (such as managers or supervisors within the organisation) (Connoly & Begg, 2015). The system definition should include: how the database links with other systems within the organisation, what the system will do and who the users are (Auer, 2006). The outcome of the system definition is a project plan for the database.

Requirements collection and analysis: The main activity of this stage is the collection and analysis of the requirements for the database system (Connoly & Begg, 2015). In this stage, information is gathered using fact finding techniques (user interviews, questionnaires, documentation investigation, use cases, etc.) and then the information is analysed to determine the requirements for the database system (Connoly & Begg, 2015). The outcome of this stage is a set of user requirements for the database system.

Database design: This is divided into three main phases: conceptual, logical and physical design.

> *Conceptual design* is the phase in which the model of the data to be used independent of all physical considerations is constructed - this model is based on the requirements specification of the system (Auer, 2006) (Connoly & Begg, 2015).
>
> *Logical design* is the phase in which the model of the data to be used is based off a specific data model but is independent of a DBMS is constructed - this model is based on the relational data model for the database (Auer, 2006) (Connoly & Begg, 2015).
>
> *Physical design* is the phase in which the description of the implementation of the database on secondary storage is created - this describes the base relations, file organisations, indexes, integrity constraints and security measures of the database (Auer, 2006) (Connoly & Begg, 2015).

Database management system selection: This stage defines the criteria for a DBMS system to support the database system, and evaluating and selecting products based on the criteria (Auer, 2006).

Application design: This stage involves designing the user interface and programs that interact with the database, these are based on the user requirements specification and produce a system design (Connoly & Begg, 2015).

Prototyping: In this stage, a working model of the database system is designed and built to allow the users to identify the features of the system that work well and the features that require more work, and to give feedback on improvements and features that need to be added (Auer, 2006) (Connoly & Begg, 2015).

Implementation: During this stage, the database and applications are built and programmed, all components outlined in the system design are built and integrated into a complete system (Auer, 2006). The outcome of this phase is a complete system made of the created database and related applications.

Data conversion and loading: This stage involves the transference of data from the previous system onto the newly implemented database system and converting existing applications within the business to work with the database system (Connoly & Begg, 2015).

Testing: The database is run in order to find any errors or issues in this stage, this should be conducted before the system goes live in the business. This is done by implementing planned test strategies and using realistic data and scenarios (Connoly & Begg, 2015). The testing phase ensures that all user requirements are met to the specified standard.

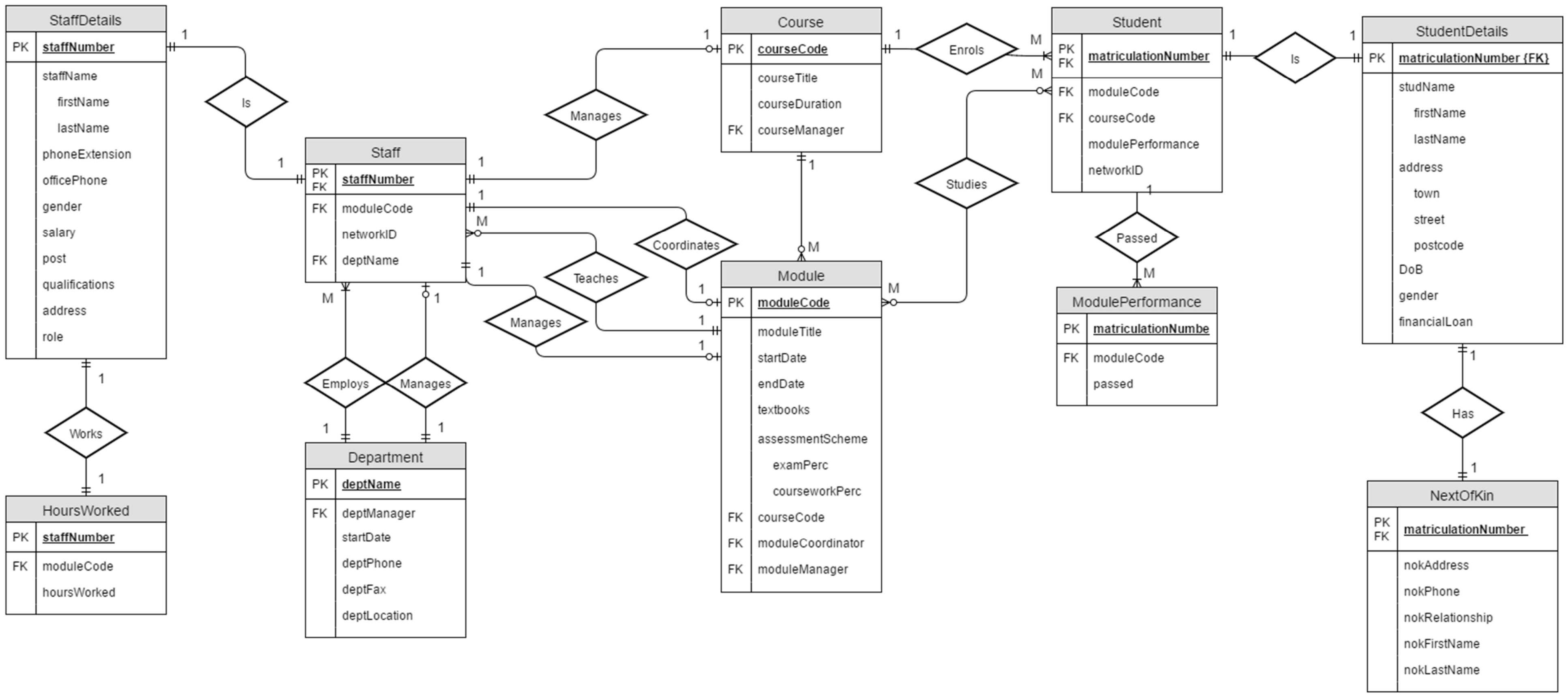
Operational maintenance: This stage takes place after the database system has been installed, tested and implemented - this stage involves monitoring system performance, and maintaining and upgrading the system when required (Connoly & Begg, 2015). The outcome of this stage is an updated system or a request for system modification.

The database systems development lifecycle should be followed when developing a database system to ensure that the system is efficiently designed, well documented and reliable.

QUESTION 2

QUESTION 2.1

Tithuto College Database Entity Relationship Diagram



Screenshot showing the Entity Relationship Diagram of Tithuto College's database showing all entities, attributes, primary keys, foreign keys, relationships, and cardinalities. Created using Draw.io.

Sources used: (SmartDraw, 2017), (Connoly & Begg, 2015)

QUESTION 2.2

MODULE

Column Name	Data Type (Length)	Key	Required	Default Value	Remarks
moduleCode	Number	Primary Key	Yes	None	Pre-defined unique code per module
moduleTitle	Text (100)	No	Yes	None	Unique
startDate	Date	No	Yes	None	Format: dd-mm-yyyy
endDate	Date	No	Yes	None	Format: dd-mm-yyyy
textbooks	Text (300)	No	Yes	None	
assessmentScheme	Text (50)	No	Yes	None	Composite attribute
examPerc	Number	No	Yes	None	Percentage
courseworkPerc	Number	No	Yes	None	Percentage
courseCode	Number	Foreign Key	Yes	None	References *Course* table - uses courseCode as the foreign key.
moduleCoordinator	Number	Foreign Key	Yes	None	References *Staff* table - uses staffNumber as the foreign key.
moduleManager	Number	Foreign Key	Yes	None	References *Staff* table - uses staffNumber as the foreign key.

STUDENT

Column Name	Data Type (Length)	Key	Required	Default Value	Remarks
matriculationNumber	Number	Primary Key, Foreign Key	Yes	None	Unique number assigned when a student enrols in a course. Foreign key: references *StudentDetails* table.
moduleCode	Number	Foreign Key	Yes	None	References *Module* table - uses moduleCode as the foreign key.
courseCode	Number	Foreign Key	Yes	None	References *Course* table - uses courseCode as the foreign key.
modulePerformance	Boolean	No	Yes	False	Linked to *ModulePerformance* table. If all field values are true in *ModulePerformance* table, indicating all modules are passed, the course will be passed.
networkID	Number	No	Yes	No	Unique

COURSE

Column Name	Data Type (Length)	Key	Required	Default Value	Remarks
courseCode	Number	Primary Key	Yes	None	Primary Key - predefined unique code per course
courseTitle	Text (100)	No	Yes	None	Unique
courseDuration	Number	No	Yes	None	Measured in years
courseManager	Number	Foreign Key	Yes	None	References *Staff* table - uses staffNumber as the foreign key.

STUDENT DETAILS

Column Name	Data Type (Length)	Key	Required	Default Value	Remarks
matriculationNumber	Number	Primary Key	Yes	None	Unique number assigned when student enrols in a course
studName	Text (50)	No	Yes	None	Composite attribute
firstName	Text (25)	No	Yes	None	
lastName	Text (25)	No	Yes	None	
address	Text (200)	No	Yes	None	Composite attribute
town	Text (25)	No	Yes	None	
street	Text (150)	No	Yes	None	
postcode	Text (25)	No	Yes	None	
DoB	Date	No	Yes	None	Format: dd-mm-yyyy
gender	Text (10)	No	Yes	None	Data validation: only “male”, “female”, and “other” allowed.
financialLoan	Number	No	No	None	

-

NEXT OF KIN

Column Name	Data Type (Length)	Key	Required	Default Value	Remarks
matriculationNumber	Number	Primary Key, Foreign Key	Yes	None	Unique number assigned when a student enrols in a course. Foreign key: references *StudentDetails* table.
nokAddress	Text (200)	No	Yes	None	
nokPhone	Number	No	Yes	None	Input mask: (###) ###-####
nokRelationship	Text (25)	No	Yes	None	
nokFirstName	Text (25)	No	Yes	None	
nokLastName	Text (25)	No	Yes	None	

DEPARTMENT

Column Name	Data Type (Length)	Key	Required	Default Value	Remarks
deptName	Text (50)	Primary Key	Yes	None	Unique name assigned to each department within the college
deptManager	Number	Foreign Key	Yes	None	References *Staff* table - uses staffNumber as the foreign key.
startDate	Date	No	Yes	None	Format: dd-mm-yyyy
deptPhone	Number	No	Yes	None	Input mask: (###) ###-####
deptFax	Number	No	Yes	None	Input mask: (###) ###-####
deptLocation	Text (25)	No	Yes	None	

STAFF DETAILS

Column Name	Data Type (Length)	Key	Required	Default Value	Remarks
staffNumber	Number	Primary Key	Yes	None	Unique number assigned to each staff member.
staffName	Text (50)	No	Yes	None	Composite attribute
firstName	Text (25)	No	Yes	None	
lastName	Text (25)	No	Yes	None	
phoneExtension	Number	No	Yes	None	Input mask: ####
officePhone	Number	No	Yes	None	Input mask: (###) ###-####
gender	Text (10)	No	Yes	None	Data validation: only “male”, “female”, and “other” allowed.
salary	Currency	No	Yes	None	Currency used: ZAR (R)
post	Text (30)	No	Yes	None	
qualifications	Text (100)	No	Yes	None	
address	Text (200)	No	Yes	None	
role	Text (50)	No	No	None	

STAFF

Column Name	Data Type (Length)	Key	Required	Default Value	Remarks
staffNumber	Number	Primary Key, Foreign Key	Yes	None	Unique number assigned to each staff member. Foreign Key: references *StaffDetails* table.
moduleCode	Number	Foreign Key	Yes	None	References *Module* table - uses moduleCode as the foreign key.
networkID	Number	No	Yes	No	Unique
deptName	Text (50)	Foreign Key	Yes	None	References *Department* table - uses deptName as foreign key.

HOURS WORKED

Column Name	Data Type (Length)	Key	Required	Default Value	Remarks
staffNumber	Number	Primary Key, Foreign Key	Yes	None	Unique number assigned to each staff member. Foreign Key: references *StaffDetails* table.
hoursWorked	Number	No	Yes	No	Measured in hours.
moduleCode	Number	Foreign Key	Yes	None	References *Module* table - uses moduleCode as the foreign key.

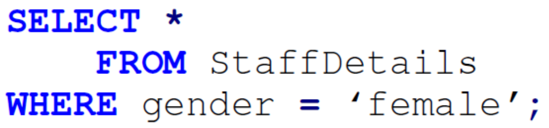
MODULE PERFORMANCE

Column Name	Data Type (Length)	Key	Required	Default Value	Remarks
matriculationNumber	Number	Primary Key, Foreign Key	Yes	None	Unique number assigned when a student enrols in a course. Foreign key: references *StudentDetails* table.
moduleCode	Number	Foreign Key	Yes	None	References *Module* table - uses moduleCode as the foreign key.
passed	Boolean	No	Yes	False	

QUESTION 3

QUESTION 3.1

List all the details of staff member who are female

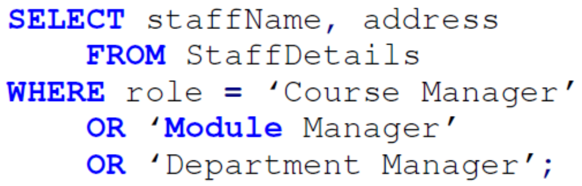


```
SELECT *
    FROM StaffDetails
WHERE gender = ‘female’;
```

(TutorialsPoint, 2017).

QUESTION 3.2

List the names and addresses of staff members who are Managers

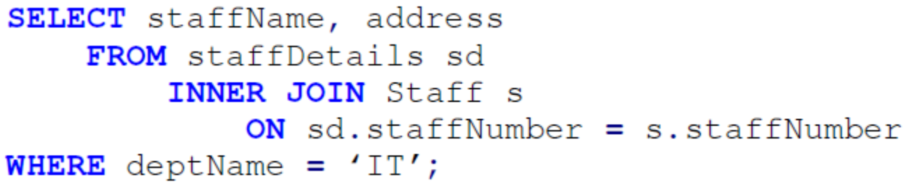


```
SELECT staffName, address
    FROM StaffDetails
WHERE role = ‘Course Manager’
    OR ‘Module Manager’
    OR ‘Department Manager’;
```

(TutorialsPoint, 2017).

QUESTION 3.3

Produce a list of the names and addresses of all staff members who work for the ‘IT’ department.

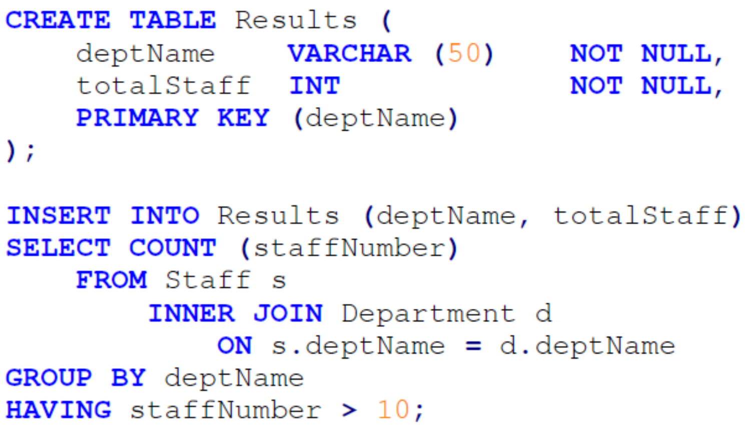


```
SELECT staffName, address
    FROM staffDetails sd
        INNER JOIN Staff s
            ON sd.staffNumber = s.staffNumber
WHERE deptName = ‘IT’;
```

(TutorialsPoint, 2017) (SQL Join, 2016).

QUESTION 3.4

List the total number of staff members in each department for those departments with more than 10 staff members. Create an appropriate heading for the columns of the results table.



```
CREATE TABLE Results (
    deptName     VARCHAR (50)     NOT NULL,
    totalStaff   INT              NOT NULL,
    PRIMARY KEY (deptName)
);

INSERT INTO Results (deptName, totalStaff)
SELECT COUNT (staffNumber)
    FROM Staff s
        INNER JOIN Department d
            ON s.deptName = d.deptName
GROUP BY deptName
HAVING staffNumber > 10;
```

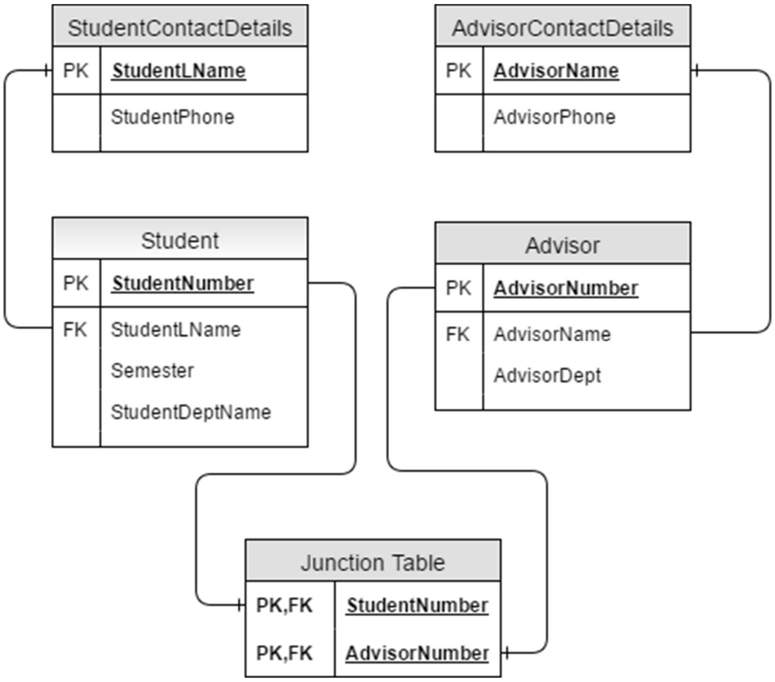
(SQL Join, 2016) (TutorialsPoint, 2017) (Alexander, 2014)

QUESTION 4

UN-NORMALISED FORM (UNF)

StudentNumber	StudentLName	StudentPhone	StudentDeptName	AdvisorName	AdvisorPhone	AdvisorDept	Semester
TN252	Khoza	596-579	IT	Toela	687-981	Commerce	1
TN350	Pillay	567-987	IT	Caddy	687-985	Arts	1
TN452	Sibiya	458-698	Commerce	Sini	687-853	Humanities	2
TN557	Marco	489-967	Arts	Khumalo	687-892	IT	1
TN254	Tiger	557-964	Humanities	Naido	687-756	Science	1

THIRD NORMAL FORM (3NF)



3NF table relationships

Student

StudentNumber (PK)	StudentLName (FK)	Semester	StudentDeptName
TN252	Khoza	1	IT
TN350	Pillay	1	IT
TN452	Sibiya	2	Commerce
TN557	Marco	1	Arts
TN254	Tiger	1	Humanities

StudentContactDetails

StudentLName (PK)	StudentPhone
Khoza	596-579
Pillay	567-987
Sibiya	458-698
Marco	489-967
Tiger	557-964

Advisor

AdvisorNumber (PK)	AdvisorName (FK)	AdvisorDept
LN111	Toela	Commerce
LN112	Caddy	Arts
LN113	Sini	Humanities
LN114	Khumalo	IT
LN115	Naido	Science

AdvisorContactDetails

AdvisorName (PK)	AdvisorPhone
Toela	687-981
Caddy	687-985
Sini	687-853
Khumalo	687-892
Naido	687-756

JunctionTable

StudentNumber (PK, FK)	AdvisorNumber (PK, FK)
TN252	LN111
TN350	LN112
TN452	LN113
TN557	LN114
TN254	LN115

Sources used: (channel5567, 2015) (Ghug, 2011).

REFERENCES

Alexander, 2014. *SQL select from inner join where count greater than.* [Online]
Available at: http://stackoverflow.com/a/22128255
[Accessed March 2017].

Apoorva College, 2011. *Costs and Risks of Database Approach.* [Online]
Available at: http://www.smartclass.co/2011/02/costs-and-risks-of-database-approach.html
[Accessed March 2017].

Auer, L., 2006. *Designing Databases.* [Online]
Available at:
http://www2.amk.fi/digma.fi/www.amk.fi/opintojaksot/0303011/1142845462205/1142847774995/1142849037295/1143037341377.html
[Accessed March 2017].

Champagne, J., 2016. *How Much Does Database Management Software Cost?.* [Online]
Available at: http://blog.capterra.com/how-much-does-database-management-software-cost/
[Accessed March 2017].

channel5567, 2015. *Normalization - 1NF, 2NF, 3NF and 4NF [Video file].* [Online]
Available at: https://www.youtube.com/watch?v=UrYLYV7WSHM
[Accessed March 2017].

Connoly, T. & Begg, C., 2015. *Database Systems: A Practical Approach to Design, Implementation, and Management.* 6th ed. Harlow: Pearson.

Ghug, D., 2011. *Database Normalization - Explained with Examples [Video File].* [Online]
Available at: https://www.youtube.com/watch?v=U-F_fRJ_YTQ&t=3s
[Accessed March 2017].

Rouse, M., 2017. *What is a database management system?.* [Online]
Available at: http://searchsqlserver.techtarget.com/definition/database-management-system
[Accessed March 2017].

Rouse, M., 2017. *What is a database?.* [Online]
Available at: http://searchsqlserver.techtarget.com/definition/database
[Accessed March 2017].

SmartDraw, 2017. *Entity Relationship Diagram.* [Online]
Available at: https://www.smartdraw.com/entity-relationship-diagram/
[Accessed March 2017].

SQL Join, 2016. *SQL Joins Explained.* [Online]
Available at: http://www.sql-join.com/sql-join-types
[Accessed March 2017].

TutorialsPoint, 2017. *SQL Tutorial.* [Online]
Available at: http://www.tutorialspoint.com/sql/sql_tutorial.pdf
[Accessed March 2017].