SCHOOL OF EEECS

CSC1023 Databases

Project Assignment

Submission Date: 5pm, 1st Apr 2022

NOTE:

This project assignment constitutes 30% of the total marks for this module.

The total mark for this project assignment is 100.

Answer **ALL** questions.

Software required: **MySQL**, **phpMyAdmin**

This project is an **individual** piece of work. All materials that you submit should be **your own work**. Collusion and plagiarism are academic offences and should be avoided.

Database Implementation for INTO University Partnerships

Project Objectives

This project is designed to assess the students in their ability to demonstrate competency and understanding in:

- Creation of RDBMS using MySQL from a provided schema
- Implementing constraints to ensure referential integrity
- Using SQL for creation of tables
- Using phpMyAdmin for importing data into the tables
- Using SQL for extraction of data to answer specific queries

Project Deliverables

1. A script in the name of ***schema.sql*** that contains SQL statements to create relational tables from the supplied data dictionary (See Appendix 1).

2. A script in the name of ***insert.sql*** that contains SQL statements to import the provided data into the relational tables (Data available to download from Canvas).

3. A script in the name of ***query.sql*** that contains SQL statements for the ten queries outlined in Procedure 4.

4. These three scripts should be **zipped** together and uploaded onto Canvas in the following format **XXXXXXX.zip** (where XXXXXXX is your student number) by the submission due date stated on the cover page of this coursework.

Project Outline

INTO University Partnerships is a pathway education provider focused on the provision of foundation courses for international students with 30 leading universities in the US, the UK and Australia. **You have been asked to implement and query a database for INTO University Partnerships based on the provided schema and queries.**

Procedure

1. Examine the ER diagram shown in **Figure 1** and study the following relationship cardinality of the database:

- Staff have one and only one department. A staff member must be linked to one Department and a Department must have at least one staff
- Staff have zero or more subordinate staff that they manage and staff may have zero or one supervisor.
- Staff may be involved in zero or more activities and each activity may have zero or more budgets. One budget is linked to one activity (or not)
- Staff have one and only one campus as a base and each campus must host at least one staff.
- A campus has at least one and probably more than one classroom and each classroom will belong to only one campus.
- Staff teach zero or more modules but each module must have one member of staff to teach on it.
- A classroom may be assigned to one module and a module may be assigned zero or more classrooms as a teaching space.
- A Student must affiliate to one Department and may enrol in zero or more Modules.
- A module can enrol zero or more students. Each student may enrol in zero or more modules.

Entity Relationship Diagram

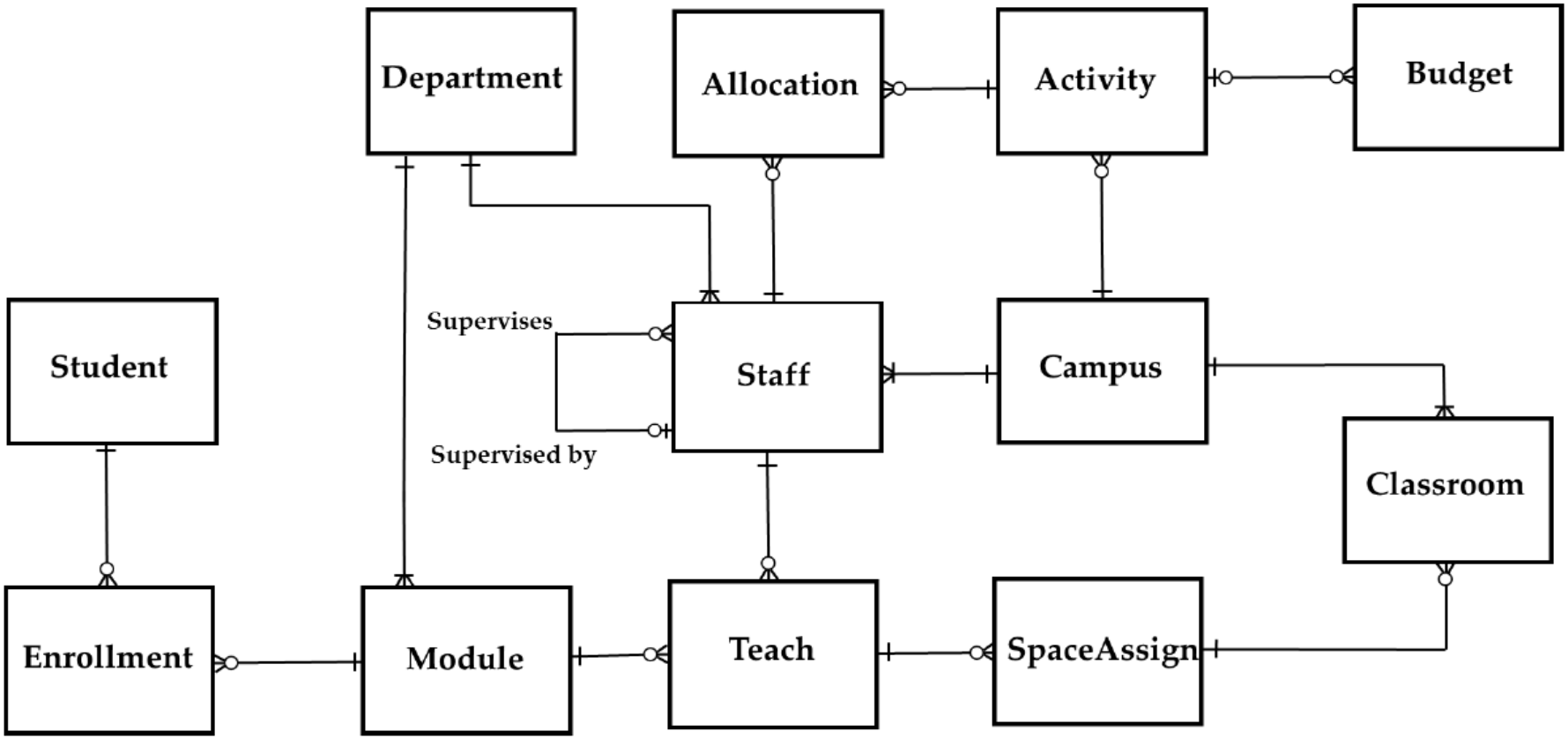

Figure 1. Entity Relationship Diagram (ERD)

Procedure

2. Create tables according to the data dictionary in **Appendix**.

3. Download the Sample data from CSC1023 CANVAS page and import the data into your created tables.

4. Exploiting the database, use your knowledge of SQL to find out the answers to the following queries:

a) What is the average budget for the Coding Competition and the Welcome Party Activities? Present your result in a column entitled 'Average'.

b) Write an SQL query to return the details (i.e. StaffID, the name and campus address) of the staff members whose contract are due to finish in the year 2019.

c) Update the campus table with additional campus information:

Address = "122, Pineapple Avenue, New York"
Campus Manager = "Michael Carlos"
Country = "USA"
Status = "Open Soon"

d) Find the list of staff (to include their salutation/title, first name, last name) who work in the EEECS department.

e) Get an alphabetical list of the top 10 staff members.

f) Perform a count of the number of activities of each staff member, and output the first names and last names of staff members who were allocated with activities. Do not count staff members who were not allocated to any activity.

g) Produce a list of the students (names, addresses and scores) who are taking the module 'Java Programming'. Show your result ordered by the score, from highest to lowest.

h) What is the total budget that has been approved by Lisa Amy? Show the result in a column called "Lisa's Spend".

i) Find all the staff whose last name begins with 'C' and are also working in the London Campus.

j) Write an SQL query to output the details of the student who had the second highest score for the academic year 2020. Your work should allow for the output screen to show the student's name, the module taken, and whether the module was taken in semester 1, semester 2 or in the summer.

NOTE

This project is an **individual** piece of work. All materials that you submit should be **your own work**. Collusion and plagiarism are academic offences and should be avoided.

Appendix - Data Dictionary

The following are twelve data dictionaries, each data dictionary contains the data structure of a table that needs to be created.

Database:	INTO	**Table Name:**	**Module**	**Key Field(s):**	ModuleID

Related to:

Table Name	**Foreign Key**	**Table Name**	**Foreign Key**
Department	DeptID		

General description:

Module details for module offered by INTO

Fields:

Field Name	**R**	**I**	**U**	**Data Type**	**Length**	**Format Rule(s)**	**Default Value**	**Description**	**Example Data**
ModuleID	Y	Y	Y	BIGINT				AUTO_INCREMENT ModuleID primary key	109801
ModuleName	Y	Y	N	VARCHAR	255			Module name	"Big Data"
DeptID	Y	N	N	BIGINT				DeptID for hosting department FK DeptID on Department.DeptID	129801
Programme	N	N	N	VARCHAR	255			Degree programme associated	"Bachelor of CS"
TMode	N	N	N	VARCHAR	255			Teaching mode	"Online", "Face to Face"
Date_of_Firstoffer	N	N	N	DATE				Date of first time offer	1998-11-25

Key: R = Required (Y/N), I = Indexed (Y/N), U = Unique (Y/N)

Database:	INTO	**Table Name:**	Department	**Key Field(s):**	DeptID

Related to:

Table Name	**Foreign Key**	**Table Name**	**Foreign Key**

General description:

Table of INTO department information.

Fields:

Field Name	**R**	**I**	**U**	**Data Type**	**Length**	**Format Rule(s)**	**Default Value**	**Description**	**Example Data**
DeptID	Y	Y	Y	BIGINT				AUTO_INCREMENT Department ID, primary key	12389
DeptName	Y	N	N	VARCHAR	255			Department Name	“EEECS”
Faculty	Y	N	N	VARCHAR	255			Faculty that the department attaches	“Science”
HODName	Y	N	Y	VARCHAR	255			Department head name	“Jimmy Maire”
Address	Y	N	Y	VARCHAR	255			Department address	“10 Milan Road”

Key: R = Required (Y/N), I = Indexed (Y/N), U = Unique (Y/N)

Database:	INTO	Table Name:	Enrolment	Key Field(s):	EID

Table Name	Foreign Key	Table Name	Foreign Key
Student	StudentID	Module	ModuleID

Collection of all the possible enrolment of all students.

Fields:

Field Name	R	I	U	Data Type	Length	Format Rule(s)	Default Value	Description	Example Data
EID	Y	Y	Y	BIGINT				AUTO_INCREMENT ID for the Enrolment, PRIMARY KEY for Grade	12345
StudentID	Y	Y	N	BIGINT				Student ID FK on Student.StudentID	12898
ModuleID	Y	Y	N	BIGINT				Module ID FK on Module. ModuleID	28980
Score	Y	N	N	INT			1	Module score for student	95
Semester	Y	Y	N	VARCHAR	255			Teaching semester	"S1", "S2" or "Summer"
Year	Y	Y	N	BIGINT				Academic year	2018

Key: R = Required (Y/N), I = Indexed (Y/N), U = Unique (Y/N)

Database:	INTO	Table Name:	Student	Key Field(s):	StudentID

Table Name	Foreign Key	Table Name	Foreign Key

Table of student details

Fields:

Field Name	R	I	U	Data Type	Length	Format Rule(s)	Default Value	Description	Example Data
StudentID	Y	Y	Y	BIGINT				AUTO_INCREMENT primary key	2213027
StudentName	Y	N	N	VARCHAR	255			Student's name	"Jason Wood"
DoB	Y	N	N	DATE				Day of Birth	2000-01-25
Address	Y	N	N	VARCHAR	255			Student post address	"11 Meeks Street"
StudyType	Y	N	N	VARCHAR	255			Type of Study	"Part time"
Date_of_Start	Y	N	N	DATE				Day of Starting	2018-09-25

Key: R = Required (Y/N), I = Indexed (Y/N), U = Unique (Y/N)

Database:	INTO	Table Name:	Teach	Key Field(s):	TID

Related to:

Table Name	Foreign Key	Table Name	Foreign Key
Module	ModuleID	Staff	StaffID
SpaceAssign	SpaID		

General description:

Table Teach to link Module and Staff

Fields:

Field Name	R	I	U	Data Type	Length	Format Rule(s)	Default Value	Description	Example Data
TID	Y	Y	Y	BIGINT				Teaching Task ID AUTO_INCREMENT ID, PRIMARY KEY for Teach	123
ModuleID	Y	Y	N	BIGINT				Module ID FK on Module.ModuleID	12345
StaffID	Y	Y	N	BIGINT				Staff ID FK on Staff StaffID	45678
SpaID	Y	N	N	BIGINT				SpaID FK on SpaceAssign SpaID	12045
Semester	Y	Y	N	VARCHAR	255			Teaching semester	"S1", "S2" or "Summer"
Year	Y	Y	N	BIGINT				Academic year	2018

Key: R = Required (Y/N), I = Indexed (Y/N), U = Unique (Y/N)

Database:	INTO	Table Name:	Staff	Key Field(s):	StaffID

Related to:

Table Name	Foreign Key	Table Name	Foreign Key
Department	DeptID	Campus	CaID

General description:

Staff table containing staff details

Fields:

Field Name	R	I	U	Data Type	Length	Format Rule(s)	Default Value	Description	Example Data
StaffID	Y	Y	Y	BIGINT				AUTO_INCREMENT Staff ID, PRIMARY KEY	20901
Title	N	N	N	VARCHAR	32			Title of employee	"Rt. Hon. Professor"
FirstName	Y	Y	N	VARCHAR	255			First Name	"Bob"
LastName	Y	Y	N	VARCHAR	255			Last Name	"Smith"
DeptID	**Y**	Y	N	BIGINT				FK Department ID	12345
CaID	Y	N	N	INT				FK Staff on Campus CaID	1
Joined	N	N	N	DATE				Start Date	2019-02-10
LeftD	N	N	N	DATE				Contract Finish Date (ex-employees)	2020-06-30
Current	Y	Y	N	BOOLEAN			1	Current employee flag	1
Salary	Y	N	N	BIGINT				Annual salary amount	35,000
ContractType	Y	N	N	VARCHAR	32			Contract Type	"Part time"
SupervisorID	N	Y	N	BIGINT				Supervisor Identification	19901

Key: R = Required (Y/N), I = Indexed (Y/N), U = Unique (Y/N)

Important note: Supervisor here is a foreign key within Staff pointing back to Staff (i.e. a Staff instance can manage another Staff instance). You must make sure it's a non-required (NULLable) field otherwise no data can be inserted here.

Database:	INTO	Table Name:	Allocation	Key Field(s):	StaffID AcID

Related to:

Table Name	Foreign Key	Table Name	Foreign Key
Staff	StaffID	Activity	AcID

General description:

Link table to show which staffs are assigned to which activities

Fields:

Field Name	R	I	U	Data Type	Length	Format Rule(s)	Default Value	Description	Example Data
StaffID	Y	Y	N	BIGINT				Staff ID FK on Staff StaffID	456
AcID	Y	Y	N	BIGINT				AcID FK on Activity.AcID	789

Key: R = Required (Y/N), I = Indexed (Y/N), U = Unique (Y/N)

Database:	INTO	Table Name:	Activity	Key Field(s):	AcID

Related to:

Table Name	Foreign Key	Table Name	Foreign Key
Campus	CaID	Budget	BuID

General description:

Table of projects NIU staffs may be working on

Fields:

Field Name	R	I	U	Data Type	Length	Format Rule(s)	Default Value	Description	Example Data
AcID	Y	Y	Y	BIGINT				AUTO_INCREMENT Activity ID, PRIMARY KEY	1
Title	Y	Y	N	VARCHAR	255			Activity Title	"Teaching and Learning Workshop"
CaID	N	N	N	INT				Campus ID FK on Campus CaID	2
BuID	N	N	N	BIGINT				BuID FK on Budget BuID	1034
Status	Y	N	N	VARCHAR	255			Activity status	"On going", "Start"
Started	Y	N	N	DATE				Date activity started	2011-06-10
Ended	N	N	N	DATE				Date activity ended (or NULL)	2013-12-01
Internal	Y	N	N	BOOLEAN			0	Internal project flag	1

Key: R = Required (Y/N), I = Indexed (Y/N), U = Unique (Y/N)

Database:	INTO	Table Name:	Campus	Key Field(s):	CaID

Related to:

Table Name	Foreign Key	Table Name	Foreign Key

General description:

Table of campus INTO staffs may be based on

Fields:

Field Name	R	I	U	Data Type	Length	Format Rule(s)	Default Value	Description	Example Data
CaID	Y	Y	Y	INT				AUTO_INCREMENT Campus ID, PRIMARY KEY	1
Address	N	N	N	VARCHAR	255			Campus Address	"333 Grant Plam Tree Road, Sydney"
GmName	N	N	N	VARCHAR	255			General manager name	"Jeff Tan"
Country	N	N	N	VARCHAR	255				"Australia"
Status	Y	N	N	VARCHAR	255			Campus status	"Open", "Closed", "Reopening"

Key: R = Required (Y/N), I = Indexed (Y/N), U = Unique (Y/N)

Database:	INTO	Table Name:	Budget	Key Field(s):	BuID

Related to:

Table Name	Foreign Key	Table Name	Foreign Key

General description:

Table of budget plan for activity

Fields:

Field Name	R	I	U	Data Type	Length	Format Rule(s)	Default Value	Description	Example Data
BuID	Y	Y	Y	BIGINT				AUTO_INCREMENT Budget ID, PRIMARY KEY	1
Amount	N	N	N	BIGINT				Budget amount	5000
Approver	N	N	N	VARCHAR	255			Approver name	"Jeff Tan"
Payee	N	N	N	VARCHAR	255			Department who pays	"EEECS"
Status	Y	N	N	VARCHAR	255			Budget status	"Open", "Completed"

Key: R = Required (Y/N), I = Indexed (Y/N), U = Unique (Y/N)

Database:	INTO	Table Name:	Classroom	Key Field(s):	RmID

Related to:

Table Name	Foreign Key	Table Name	Foreign Key
Campus	CaID		

General description:

Table of classroom details

Fields:

Field Name	R	I	U	Data Type	Length	Format Rule(s)	Default Value	Description	Example Data
RmID	Y	Y	Y	BIGINT				AUTO_INCREMENT Classroom ID, PRIMARY KEY	156
Capacity	N	N	N	BIGINT				Class size hosted	50
CaID	N	N	N	INT				Campus ID FK on Campus CaID	2
Location	N	N	N	VARCHAR	255			Location of classroom	"Rm102, CS Tower"
Type	Y	N	N	VARCHAR	255			Classroom type	"General purpose", "Active learning classroom (ACL)", "Mutimedia classroom"
Status	Y	N	N	VARCHAR	255			Classroom status	"Open", "Under Repair"

Key: R = Required (Y/N), I = Indexed (Y/N), U = Unique (Y/N)

Database:	INTO	Table Name:	SpaceAssign	Key Field(s):	SpaID

Related to:

Table Name	Foreign Key	Table Name	Foreign Key
Classroom	RmID	Teach	TID

General description:

Link table to show which space are assigned to which teaching

Fields:

Field Name	R	I	U	Data Type	Length	Format Rule(s)	Default Value	Description	Example Data
SpaID	Y	N	Y	BIGINT				AUTO_INCREMENT Space ID, PRIMARY KEY	1678
RmID	Y	Y	N	BIGINT				Room ID FK on Classroom RmID	456
TID	Y	Y	N	BIGINT				Teach ID FK on Teach.TID	789
Manager	Y	N	N	VARCHAR	255			Manager name for the assignment	"Larry Karen", "Paul Compton"
Approved	Y	N	N	VARCHAR	255			Approving status	"Yes", "No", "Pending"

Key: R = Required (Y/N), I = Indexed (Y/N), U = Unique (Y/N)