



1814ict/2814ict/7003ict/1011ICT:  
Data Management/  
Database Design/  
Applied Computing

# Weekly Workshop/ Lab 2.3 Activities

Course Convenor: AProf. Henry Nguyen  
Course developed by: Dr Mohammad Awrangjeb; AProf John Wang; Dr Zhe Wang

---

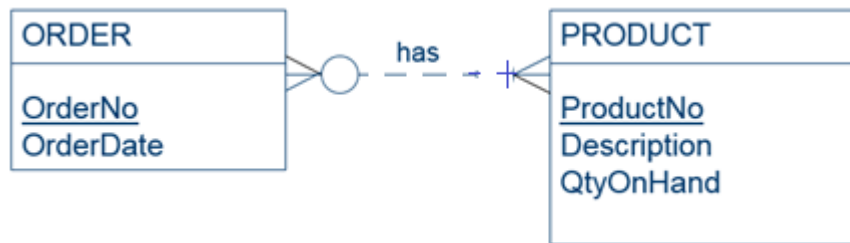
**Activity Set 1: Solve M:N relationships** [Individual Activity][15 min]

---

**Note: Remember FIVE (5) Steps to develop an ERD from Topic 2.2 lecture slides.**

**Steps:**

1. Insert a **Bridging Entity**
  2. Insert **PKs from parent entities** into new entity as **FKs**
  3. Add in **any new attributes** required
  4. **Connectivity** – Crow's foot points to the new entity
  5. Two FKs in new entity can be a composite **primary key**, or enter a new attribute to the **primary key**
- 

**1. Solve the following M:N relationship**

- a) Draw the ERD for the solution.
- b) Briefly discuss if the bridging entity in your diagram has a strong or weak relationship with the original two entities.
- c) What additional attribute can you enter in the bridging entity?

---

**Activity Set 2: Develop of an appropriate ERD** [Individual Activity][15 min]

---

**Note: Remember SIX (6) Steps to develop an ERD from Week 2 lecture**

**FOLLOW for PARTS 2 & 3 below**

**Step 1:** Find all possible **entities** [look for *nouns*].

**Step 2:** Draw entities with **attributes**, add primary keys.

**Step 3:** Find & show **relationships** [find *verbs*] one by one between entities.

**Step 4:** Find & show **connectivity** one by one between entities.

**Step 5:** Find & show **participation** one by one between entities.

**Step 6:** **Insert Foreign keys** and **Revise** above Steps 1 to 5 to ensure your ERD is complete.

---

*Note: The ERD solution to any case study may vary depending on the **assumptions** you may make. Your assumptions should not contradict any facts already given in the case study. Please write your assumptions in the ERD that you submit into your learning journals.*

## **2. Develop an ERD for the following business rules**

- Acme Pty Ltd is made up of a number of departments that manage none or more projects. Each project belongs to a department.
  - Each department employs a number of employees. Each employee works in one department.
  - A team of employees work on each project.
  - Some employees may not be assigned to any project yet and a new project may not have any team assigned to it.
  - One of the team members supervises the other team members on the project.
- 
- a) Identify the entities and make up attributes.
  - b) Draw a logical ERD.

### Activity Set 3: Continue developing a database [Individual Activity] [30 min]

#### Requirement:

- You will need the database (MyUniversity) you created in the last week. If you forgot to export it, you need to use 'MyUniversity.sql' supplied with this file in Week 4.

**Reminder:** Last week you created 'MyUniversity' database based on the following ER Diagram and an incomplete database schema that you needed to complete:

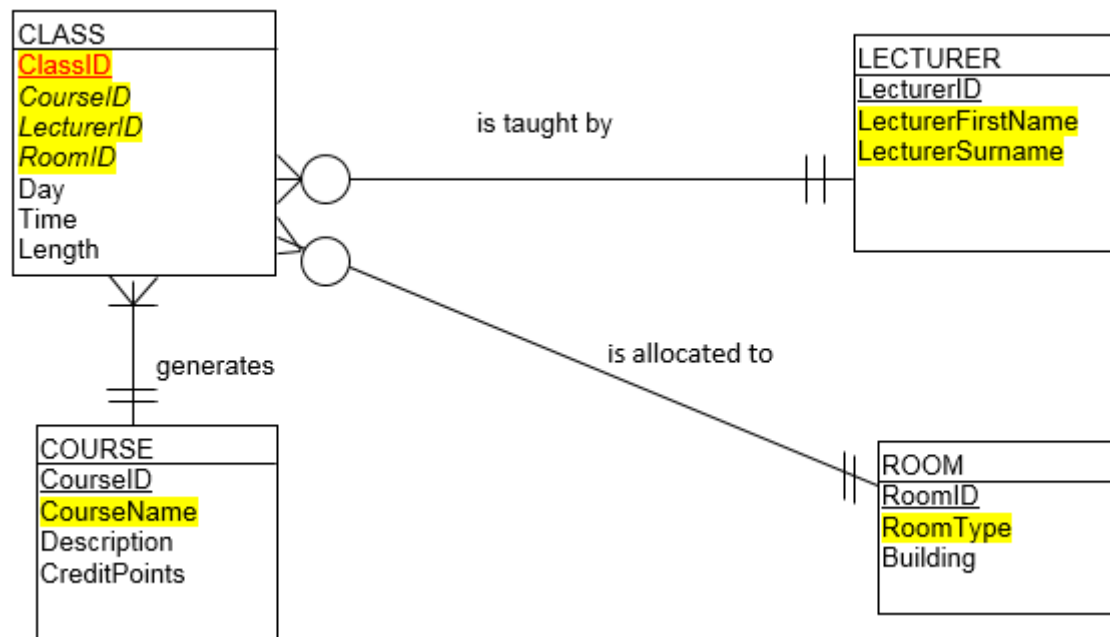


Figure 1 (ERD from Week 3)

#### Activities:

It is very important to aim for designing the database as good as possible to reflect all specifications before its implementations. This week you are going to experience some difficulties and challenges with changing the structure of some tables and altering some data; to appreciate the importance of the initial database design before its implementation!

Note that some changes to the last week ERD are highlighted in Figure 1 above. Check these changes in the ERD shown in Figure 2 below.

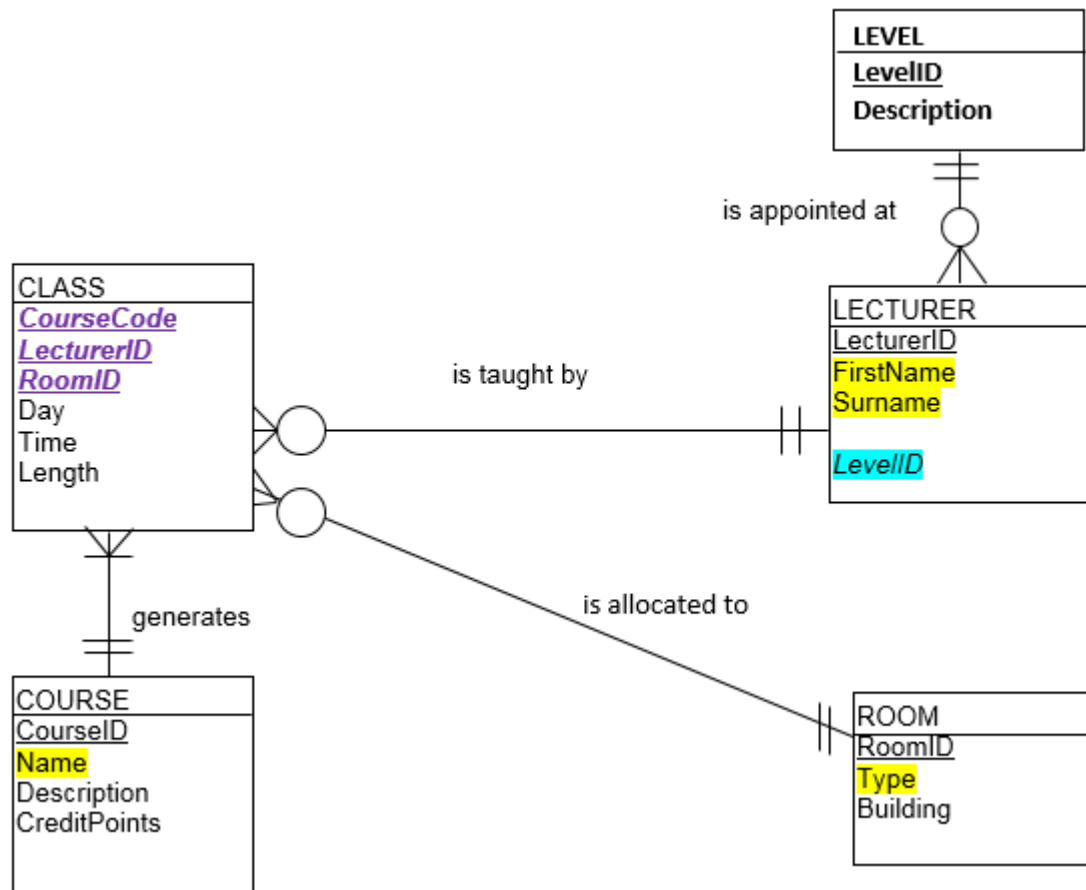


Figure 2 (changed ERD from Figure 1)

### Relational database Schema (Updated according to Figure 2):

Entity	Field Name	Datatype	Length	
ROOM	RoomID	VARCHAR	4	PRIMARY KEY
	Type	TEXT	30	
	Building	VARCHAR	4	
LECTURER	LecturerID	INT ( AUTO)		PRIMARY KEY
	FirstName	TEXT	30	
	Surname	TEXT	30	
	LevelID	INT		FOREIGN KEY REFERENCES LEVEL (LevelID)
COURSE	CourseID	VARCHAR	8	PRIMARY KEY
	Name	TEXT	30	
	Description	TEXT	100	
	CreditPoints	INT		
CLASS	CourseID	VARCHAR	8	PRIMARY KEY, FOREIGN KEY REFERENCES COURSE (CourseID)
	LecturerID	INT		PRIMARY KEY, FOREIGN KEY REFERENCES LECTURER(LecturerID)

	<b>RoomID</b>	<b>VARCHAR</b>	<b>4</b>	<b>PRIMARY KEY, FOREIGN KEY REFERENCES ROOM (RoomID)</b>
	Day	CHAR	4	
	Time	TIME	3	
	Length	INT		
<b>LEVEL</b>	<b>LevelID</b>	<b>INT</b>		<b>PRIMARY KEY</b>
	<b>Description</b>	<b>TEXT</b>	<b>30</b>	

- 1) Import your 'MyUniversity' database from Week 3 Workshop/Lab.

NOTE:

- i) If you forgot to export your database last week, import the provided 'MyUniversity.sql'.
- ii) If you get an error message when importing your file from last week, it may be that you have only exported your tables. In this case: you need to create a database first, call it "MyUniversity", click on the name of the database 'MyUniversity', and then import the database.

- 2) Alter the tables in *MyUniversity* from last week so that they now implement the ER Diagram and the database Schema on the following page (including the changes of the field names). To do this use the instructions from the previous lab and the following:

**a) For updating a field properties**

- i) Click on the name of the database (**MyUniversity**).
- ii) Click on the name of the **table** that includes the field you want to change.
- iii) Click on the **Structure** tab at the top of the table listing.
- iv) Tick the **check box beside the field** (row of the table) that you like to make the change.
- v) Click on the **Change** beside 'With selected' title.
- vi) Make the desired changes for the field properties and click 'Save'.

*Note: You can update yellow highlighted attributes in Figure 2 by following the above steps.*

**b) For inserting a new field into an existing table**

Complete the following steps for each new field in each of the tables.

- i) Click on the name of the database (*MyUniversity*).
- ii) Click on the name of the **table** where you want to add a field.
- iii) Click on the **Structure** tab at the top of the table listing.
- iv) At the end of the field listing indicate how many new fields you like to add and where you like to position it ('At End of Table', 'At Beginning of Table' or After a particular field selected from the drop down list).

*See highlighted part in the screen shot below for Lecturer table:*

#	Name	Type	Collation	Attributes	Null	Default	Comments
<input type="checkbox"/> 1	LecturerID	int(11)			No	None	
<input type="checkbox"/> 2	FirstName	text	latin1_swedish_ci		No	None	
<input type="checkbox"/> 3	Surname	text	latin1_swedish_ci		No	None	

☐ Check all    With selected:

1 column(s) after Surname

- v) Click on the **Go** button.
- vi) Add the information for the new field (following the new Database Schema given above).
- vii) Click 'Save' to create the new attribute.

*Note: You can add LevelID in Lecturer table (highlighted in cyan colour in Figure 2) by following the above steps.*

**c) For dropping an existing field from a table**

Complete the following steps for each field that you would like to drop from each of the tables.

- i) Click on the name of the database (**MyUniversity**).
- ii) Click on the name of the **table** that includes the field you want to drop.
- iii) Click on the **Structure** tab at the top of the table listing.
- iv) If the field you want to drop is part of the primary key or a foreign key:
  - (1) Click on the [Relation view](#).
  - (2) Remove the relationship for the field that you would like to Drop (select itself from the option list) and save
- v) Tick the **check box beside the field** that you would like to Drop (remove) from the table.
- vi) Click on the **Drop** beside the 'With selected': title.
- vii) Click **Yes** to confirm the drop.

*Note: You can drop (the primary key) ClassID in Class table (highlighted in red colour text in Figure 1) by following Steps i)-iii) & v)-vii).*

**d) Add a new primary key in an existing table**

- i) To assign more than one fields as the primary keys, tick the check boxes beside those fields and click the 'Primary' icon located at 'With selected' options.

*Note: You can add the composite primary key to Class table (highlighted in purple colour bold text in Figure 2) by following the above step. See a screen shot below:*

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
<input checked="" type="checkbox"/>	1 CourseID	varchar(8)	latin1_swedish_ci		No	None		
<input checked="" type="checkbox"/>	2 LecturerID	int(11)			No	None		
<input checked="" type="checkbox"/>	3 RoomID	varchar(4)	latin1_swedish_ci		No	None		
<input type="checkbox"/>	4 Day	char(4)	latin1_swedish_ci		No	None		
<input type="checkbox"/>	5 Time	time(3)			No	None		
<input type="checkbox"/>	6 Length	int(11)			No	None		

**e) For creating a new table**

Follow the instruction from the last week to create the new table: **Level** with properties indicated in the Database Schema.

*Note: You can add the new table Level shown in Figure 2 above. For data types and length follow the database schema given above.*

**f) For updating the relationships between the tables**

**Note:**

- You may need to refer to the instructions from Week 3 to update the relationships between the tables.
- To make a relationship between two tables make sure any data presenting the foreign key already exist as the primary key of the related table. If not, before creating a relationship between the two tables, create a record in the table with the primary key that matches the foreign key.

**For example,** to define the relationship between Lecturer and Level tables,

- Define **LevelID** in **Lecturer** table as an index.



Table structure									
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	LecturerID	int(11)		No	None		AUTO_INCREMENT	<a href="#">Change</a>
<input type="checkbox"/>	2	FirstName	text	latin1_swedish_ci	No	None			<a href="#">Change</a>
<input type="checkbox"/>	3	Surname	text	latin1_swedish_ci	No	None			<a href="#">Change</a>
<input checked="" type="checkbox"/>	4	LevelID	int(11)		No	None			<a href="#">Change</a>

[Check all](#)
 With selected: [Browse](#) [Change](#) [Drop](#) [Primary](#) [Unique](#) [Index](#)

- **Insert data** into Level table, as there may be some records in Lecturer table with LevelID columns having a 0 value.

*See new data entered into Level table below:*

LevelID	Description
0	No Level, for external staff
1	Starting level for an internal staff

- Go to **Relation view** and relate it to the primary key LevelID of Level table.

Table structure	<b>Relation view</b>
-----------------	----------------------

---

Foreign key constraints

Actions	Constraint properties	Column	Foreign key constraint (InnoDB)
		Database	Table
Constraint name	ON DELETE RESTRICT ON UPDATE RESTRICT	LevelID	level LevelID
		+ Add column	

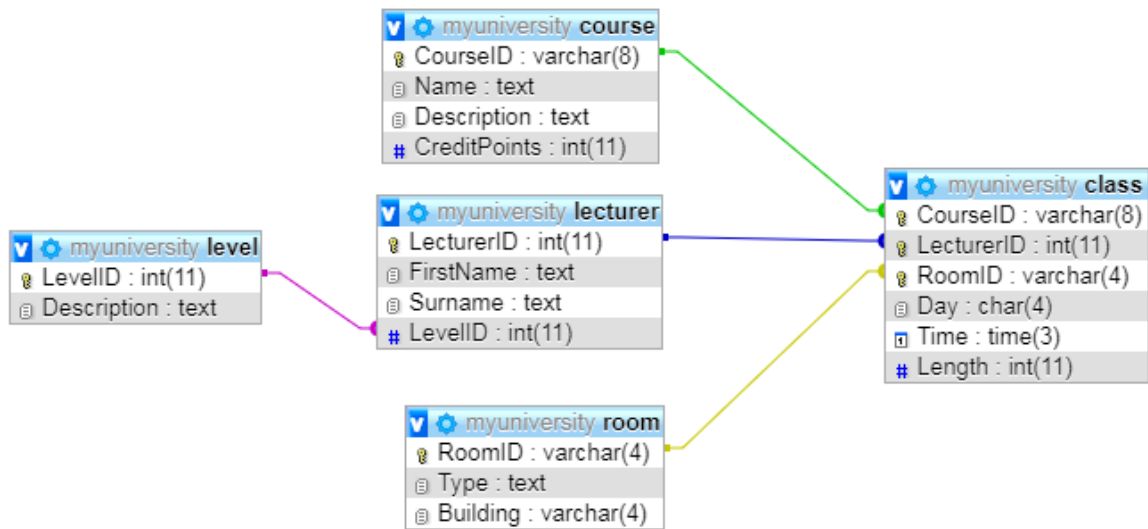
+ Add constraint

- Click on **Save**.

#### g) Viewing the relationships between tables:

- click on **MyUniversity** in the list of database names
- Click on the **More** tab (this step might not be necessary if the **Designer** tab is already displayed) and select **Designer**– this provides a visual representation of the database, its entities, the relationships between them and the attributes.

You can see the **three entities** and **FOUR 1:M** relationships as:



*Note: The notations above are a bit different from the Crow's Foot model.*

#### Notes:

- If you cannot finish the activities in the Workshop/Lab, complete them on your own time.
- Make sure to export your database. Follow the instruction from Week 3 Workshop/Lab.