

## AM05 Data Mgmt – Lab 2

---

This lab has two parts. The first part asks you to complete some SQL queries while the second asks you to complete some logical database design tasks.

### Part 1. SQL Queries

---

These queries are based on the ap database used in the hands on exercises for AM05 Session 1. The questions are the same as on p164 of Murach's MySQL (3<sup>rd</sup> Edition). I have kept the original question numbering so that you can easily find the solutions in the zip file you should have downloaded from the murach.com website if you already completed the hands on exercises. These solutions can be found in the mysql>ex\_solutions>ch05 directory.

As you test whether the actions have been taken you can write appropriate SQL queries to examine the target table before and after you UPDATE, INSERT, and DELETE queries.

1. Write an INSERT statement that adds the row to the Terms table:

terms_id:	6
terms_description:	Net due 120 days
terms_due_days:	120

Use MySQL Workbench to review the columns definitions for the terms table, and include a column list with the required columns in the INSERT statement.

2. Write an UPDATE statement that modifies the row you just added to the terms table. This statement should change the terms\_description column to "Net due 125 days" and it should change the terms\_due\_days to 125.
3. Write a DELETE statement that deletes the row you added to the terms table in exercise 1.
4. Write an INSERT statement that adds this row to the Invoices table:

invoice_id	The next automatically generated ID
vendor_id:	32
invoice_number	AX-014-027
invoice_total	\$434.58
payment_total:	\$0.00
credit_total:	\$0.00
terms_id:	2
invoice_due_date:	8/31/2018
payment_date:	NULL

Write this statement without using a column list

5. Write an INSERT query statement that adds these rows to the Invoice\_Line\_Items table:

invoice sequence	1	2
account_number:	160	527
line_item_amount	\$180.23	\$254.35

line\_item\_description: Hard drive

Exchange Server Update

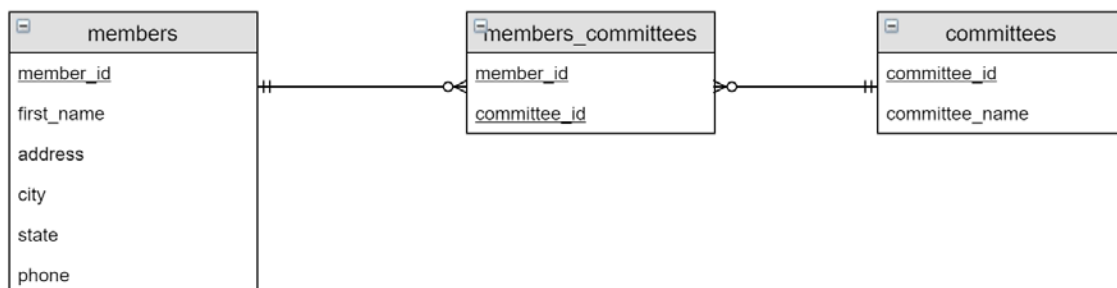
Set the invoice\_id column of these two rows to the invoice ID that was generated by MySQL for the invoice you added in exercise 4.

6. Write an UPDATE statement that modifies the invoice you added in exercise 4. This statement should change the credit\_total column so it is 10% of the invoice\_total column, and it should change the payment\_total column so the sum of the payment\_total and credit\_total columns are equal to the invoice\_total column.
7. Write an UPDATE statement that modifies the vendors table. Change the default\_account\_number to 403 for the vendor with an ID of 44.
8. Write an UPDATE statement that modifies the Invoices table. Change the terms\_id column to 2 for each invoice that's for a vendor with a default\_terms\_id of 2.
9. Write a DELETE statement that deletes the row that you added to the invoices table in exercise 4. When you execute this statement, it will produce an error since the invoice has related rows in the Invoice\_Line\_Items table. To fix that, precede the DELETE statement with another DELETE statement that deletes the line items for this invoice.

**IMPORTANT** The database needs to be restored to its initial state before you attempt your homework exercises. Rerun the create\_my\_guitar\_shop.sql script again. That should restore the data that's in the database.

This question is also on p379 of Murach's MySQL (3<sup>rd</sup> Edition). I have kept the original question numbering so that you can easily find the solutions in the zip file you should have downloaded from the murach.com website if you already completed the hands on exercises. These solutions can be found in the mysql>ex\_solutions>ch11 directory.

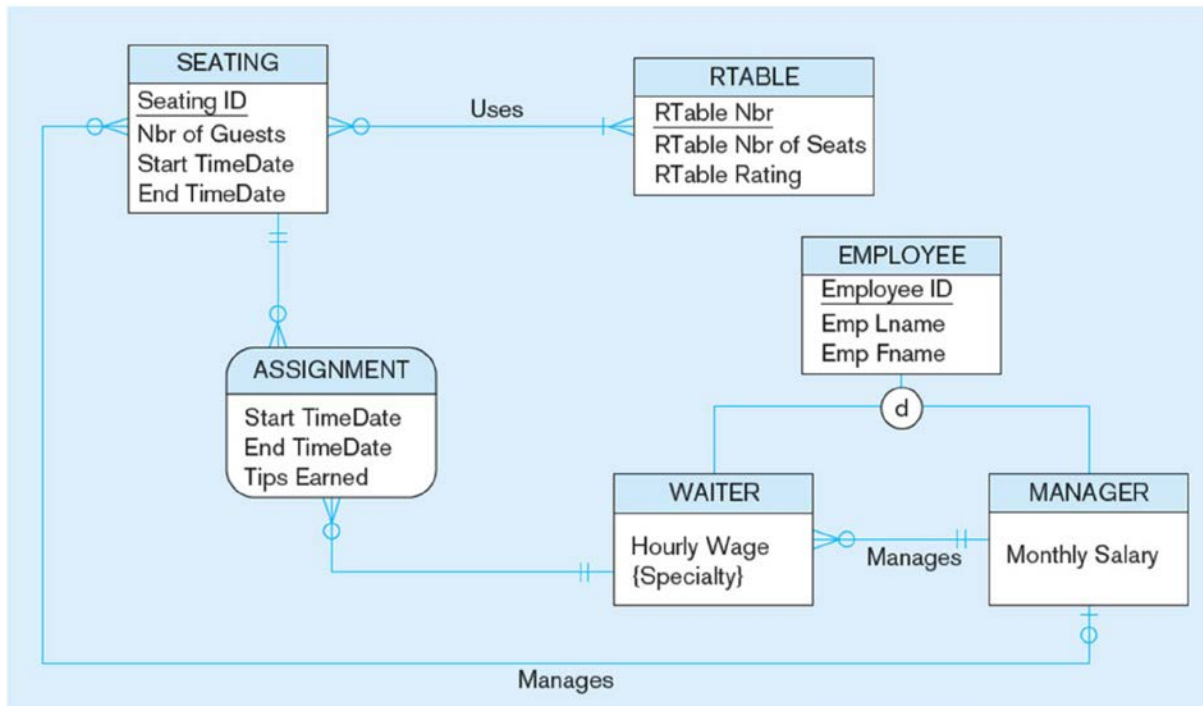
2. Write a script that contains the CREATE TABLE statements needed to implement the following design in the EX database. Include any constraints or defaults that you think are necessary. Include statements that drop the tables if they already exist.



## Part 2. Logical Database Design

In this part you can demonstrate your understanding of logical database design.

1. This is an ERD for a restaurant, its tables, and the waiters and waiting staff managers who work at the restaurant.



- a. Develop a relational schema
  - b. Identify the foreign keys that are needed to ensure referential integrity.
2. This is a relation called GRADE REPORT used at a university.

Grade Report								
StudentID	StudentName	CampusAddress	Major	CourseID	CourseTitle	Instructor Name	Instructor Location	Grade
168300458	Williams	208 Brooks	IS	IS 350	Database Mgt	Codd	B 104	A
168300458	Williams	208 Brooks	IS	IS 465	Systems Analysis	Parsons	B 317	B
543291073	Baker	104 Phillips	Acctg	IS 350	Database Mgt	Codd	B 104	C
543291073	Baker	104 Phillips	Acctg	Acct 201	Fund Acctg	Miller	H 310	B
543291073	Baker	104 Phillips	Acctg	Mkgt 300	Intro Mktg	Bennett	B 212	A

- a. Draw a relational schema and diagram the function dependencies in the relation.
- b. In what normal form is the relation? Explain.
- c. Draw a relational schema for your 3NF relations and show the referential integrity constraints.