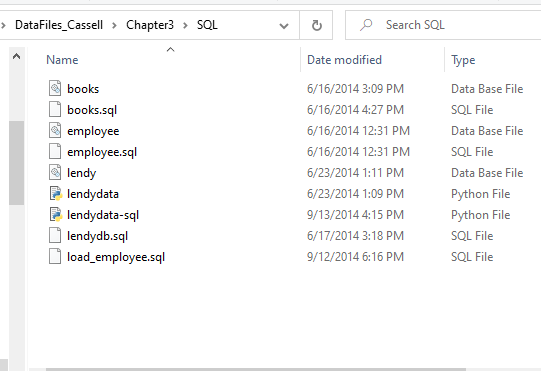
# CPE106L (Software Design Laboratory) Experiment 5 Data Modeling and Database Systems

**Content:**

|  |
| --- |
| **PreLab**   1. **Readings, Insights, and Reflection METIS ebooks**:    * A Guide to SQL. Philip J. Pratt; et al.9780357419830    * **VBID:** 9780357419830    * Chapter 1:      1. What is a Database      2. Database Requirements of TAL Distributors, Colonial Adventure Tours, and Solmaris Condominium Group   Chapter 2:   1. Database Concepts 2. Database Design Fundamentals 3. Normalization  * Core Python Programming. R. Nageswara Rao. (Not available in METIS)   VBID: 9789351198918  Chapter 24:   * 1. Types of Databases Used with Python   2. Using MySQL from Python * Python Projects. Laura Cassell. 9781118908891 Chapter 3 Chapter 3:   1. Relational Database Concepts   2. Structured Query Language. DML and DDL SQL commands   **Website(s)**:   * <https://www.sqlite.org/index.html> * <https://www.sqlitetutorial.net/sqlite-sample-database/>  1. **Answer to Questions**    * Short Answer      1. What are DML and DDL statements in Structured Query Language? Give examples of each.      2. What are the categories of SQLite Functions? Give 3 examples of each category      3. How do you check if you have SQLite installed in system using the Linux terminal? |
| **InLab**  Note: Leaders should assign task to members   1. Objectives (You can state your own objectives based on the readings in Prelab) 2. Steps Performed with screenshots of tools used:    * Visual Studio Code, PyCharm, Atom or Spyder    * Git,    * SQLite running in Linux terminal    * DB Browser |

**C.** Python SQLite Database Connection sample run with **DISCUSSIONS.** Download the chinook sample database from the website mentioned in PreLab section. Using the chinook database, run the SQLite SELECT commands in the Linux terminal and on the DB browser. Do the same using the source codes and SQL files of Cassel (Blackboard Course Materials). See below screengrabs.



**Download Cassell Codes here >>** [**Cassell SQL Codes**](https://mymailmapuaedu-my.sharepoint.com/:f:/g/personal/dapadilla_mapua_edu_ph/Er3-vwTnSTxMgyjXpWVtfzgBu1RPRj7qaqDHU8QYzcFT4Q?e=tvgcHe)

**IMPORTANT**: Include figure numbers and labels. Edit your screengrabs

# PostLab

Note: Leaders should assign the problems to members

Download SQL Scripts here >> [SQL Scripts (Colonial, etc.)](https://mymailmapuaedu-my.sharepoint.com/:f:/g/personal/dapadilla_mapua_edu_ph/EuIFHtlKek9FvFdHvzcS5REBwq0pC10TetMWbuYIm_TOxQ?e=6ThCoW)

# Machine Problems

1. Colonial Adventure Tours is considering offering outdoor adventure classes to prepare people to participate in hiking, biking, and paddling adventures. Only one class is taught on any given day. Participants can enroll in one or more classes. Classes are taught by the guides that Colonial Adventure employs. Participants do not know who the instructor for a particular class will be until the day of the class. Colonial Adventure Tours needs your help with the database design for this new venture. In each step, represent your answer using the shorthand representation and a diagram. Use crow’s foot notation for the diagram. Follow the sample SQLite chinook database ERD (Download it from Blackboard Course Materials)
   1. For each participant, list his or her number, last name, first name, address, city, state, postal code, telephone number, and date of birth.
   2. For each adventure class, list the class number, class description, maximum number of people in the class, and class fee.
   3. For each participant, list his or her number, last name, first name, and the class number, class description, and date of the class for each class in which the participant is enrolled.
   4. For each class, list the class date, class number, and class description; and the number, last name, and first name of each participant in the class.
2. Solmaris Condominium Group has many condos that are available as weekly vacation rentals. Design a database to meet the following requirements:
   1. For each renter, list his or her number, first name, middle initial, last name, address, city, state, postal code, telephone number, and email address.
   2. For each property, list the condo location number, condo location name, address, city, state, postal code, condo unit number, square footage, number of bedrooms, number of bathrooms, maximum number of persons that can sleep in the unit, and the base weekly rate.
   3. For each rental agreement, list the renter number, first name, middle initial, last name, address, city, state, postal code, telephone number, start date of the rental, end date of the rental, and the weekly rental amount. The rental period is one or more weeks.
3. Use SQLite commands to complete the following exercises.
   1. Create a table named ADVENTURE\_TRIP. The table has the same structure as the TRIP table shown in Figure 3-2 below except the TRIP\_NAME column should use the VARCHAR data type and the DISTANCE and MAX\_GRP\_SIZE columns should use the NUMBER data type. Execute the command to describe the layout and characteristics of the ADVENTURE\_TRIP table.

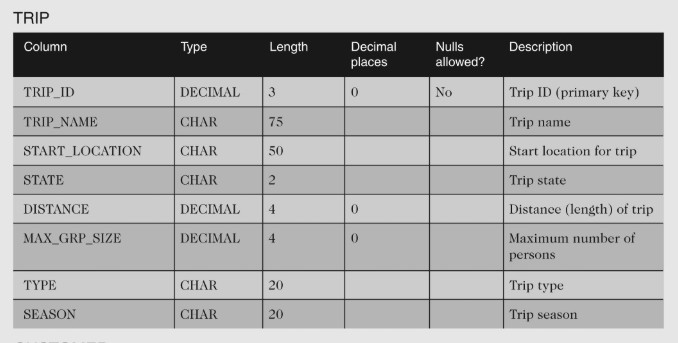


Figure 3-1. Colonial Adventure Tours TRIP Table

* 1. Add the following row to the ADVENTURE\_TRIP table: trip ID: 45; trip name: Jay Peak; start location: Jay; state: VT; distance: 8; maximum group size: 8; type: Hiking and sea- son: Summer. Display the contents of the ADVENTURE\_TRIP table.
  2. Delete the ADVENTURE\_TRIP table.
  3. Open the script file (SQLServerColonial.sql) to create the six tables and add records to the tables. Revise the script file so that it can be run in the DB Browser.
  4. Confirm that you have created the tables correctly by describing each table and comparing the results to the figures shown below. Confirm that you have added all data correctly by viewing the data in each table and comparing the results to Figures 1-4 through 1-8 shown below.



Figure 3-2. Colonial Adventure Tours Database GUIDE Table



Figure 3-2. Colonial Adventure Tours Database TRIP Table (1)

Figure 3-2. Colonial Adventure Tours Database TRIP Table (2)



Figure 3-2. Colonial Adventure Tours Database CUSTOMER Table

Figure 3-2. Colonial Adventure Tours Database RESERVATION Table (1)

TRIPGUIDES

TAIP GUID1ES (CONTINUED:

,

RESERVATION

RESERVATION\_ID TRiP \_ID TRIP\_DATE

1600001

16()()002

16()()00.1

160!)00-I

1600005

[(,()(){)()<:,

160000i

]r,()()j)()S

1600009

16()()()1()

[60()01]

1600012

16(){10D

-10

21

2R

NUM\_PERSONS

J/26/20162

6/1'/2016*2*

9112/2016

26 10/lt,/2016

*.l')*

J2

22

2tl

.18

*2*

,l

1

8

(,/25/2016

(,/18/2016

i/9/2016 W12/201h 9/11/2016

5/H/2016 WJ5/20H, G/12/2016

71'-)/2016

TRIP \_PRICE OTHER\_FEES CUSTOMER\_NUM

$55.00$0.00 llJl

$')5.00$0.IJO 11)1

$]500$000 rn.1

-I$-15.00S15.00 10-1

5$55.00$0.0o 105

]$80.00$20.IJO ] *I)(,*

8$iS.0o$10,00 107

2$.'\5,00$0,111) 11),

2$90.00$40.00 109

3$r.oo$0.011 102

J$25,00$0.00 *W2*

4$15,00$0,00 115

1$20.00$5,011 116

|  |  |
| --- | --- |
| TRIP\_ID GUIDE\_NUM | |
| I **l** | **GZOl** |
| 1 | J-{1101 |
| 2 | A1'.-ro1 |
| 2 | **SLOl** |
| 3 | *LJ* 1 |
| 4 | **BROl** |
| .,i | **UZUl** |
| 5 | **KSOl** |
| 5 | UG01 |
| 6 | **RJ101** |
| 7 | S1J01 I |
| 8 | **BRUl** |
| *L)* | **ROI** |
| 10 | **GZOl** |
| 11 | [)]TO, |
| **11** | **K. 01** |
| 11 | **l GDl** I |
| **12** | **BRUl** |
| I 13 | **RIJO** I |
| 14 | **KS02** |
| 15 | zu1 I |
| **16** | **KSfl.1** |
| 17 | ru1m |
| 18 | **KS02** |

|  |  |
| --- | --- |
| TRIP\_ID GUIDE\_NUM | |
| I 19 | lHIOl I |
|  |
| 20 | sun |
| 21 | ·\M01 |
| 22 | UGUl |
| 23 | DHOl |
| 23 | SLOl |
| .2.,4- | BR.01 |
| ,;.,:l | BROl |
| 26 | JZ01 |
| 27 | (,ZOl |
| 2, | BR l |
| 29 | [)1101 |
| 30 | AMII |
| Jl | l LOl |
| .12 | KS01 |
| .B | ll ,01 |
| 34 | KHOl |
| 35 | UZOl |
| 36 | l'Q,02 I |
|  |
| 37 | IUIOl |
| .11-1 | KSU2 I |
|  |
| .19 | RROl |
| 40 | DJIOl I |
| 4-1 | BROl |

* 1. **Debugging and Sample Run** of Python program connection to your created SQLite database (with edited screengrabs and discussion)

**IMPORTANT**: Include figure numbers and labels. Edit your screengrabs