



Credit Hours: 3

Contact Hours: This is a 3-credit course, offered in accelerated format. This means that 16 weeks of material is covered in 8 weeks. The exact number of hours per week that you can expect to spend on each course will vary based upon the weekly coursework, as well as your study style and preferences. You should plan to spend 14-20 hours per week in each course reading material, interacting on the discussion boards, writing papers, completing projects, and doing research.

Faculty Information: Faculty contact information and office hours can be found on the faculty profile page.

COURSE DESCRIPTION AND OUTCOMES

COURSE DESCRIPTION:

This course teaches students to design, implement, and use database management systems. Students gain a working knowledge of available software packages, concepts of query languages, software integration services, and security considerations. Students will also learn fundamentals of structured query language (SQL) in developing common queries and reports.

Note: Access to a Windows-based operating system is required for this class. Recommended Prior Course: ITS310 or ITS315; ITS320.

COURSE OVERVIEW:

Relational database management allows organizations to manipulate large amounts of data in order to make strategic decisions. This course provides database administrators, programmers, and other newcomers to MySQL with the knowledge and skills needed to install, configure, and program a MySQL database. This course also provides a basic understanding of Structured Query Language, or SQL, programming

COURSE LEARNING OUTCOMES:

1. Install and configure MySQL.
2. Store data in MySQL.
3. Select and modify data stored in MySQL tables.
4. Connect to MySQL from a Python script.
5. Improve MySQL database query performance.

PARTICIPATION & ATTENDANCE

Prompt and consistent attendance in your online courses is essential for your success at CSU-Global Campus. Failure to verify your attendance within the first 7 days of this course may result in your withdrawal. If for some reason you would like to drop a course, please contact your advisor.

Online classes have deadlines, assignments, and participation requirements just like on-campus classes. Budget your time carefully and keep an open line of communication with your instructor. If you are having technical problems, problems with your assignments, or other problems that are impeding your progress, let your instructor know as soon as possible.

COURSE MATERIALS

Required:

Boronczyk, T. (2015). *Jump start MySQL*. VIC, Australia: SitePoint. ISBN 978-0992461287

Suggested:

Shelton, R. and Moes, G. (2005). *Beginning MySQL*. Wrox Press. ISBN 9780764579509

Learning MySQL Development in [Lynda.com](http://www.lynda.com)

MySQL Tutorial. Retrieved from <http://www.mysqltutorial.org>

NOTE: All non-textbook required readings and materials necessary to complete assignments, discussions, and/or supplemental or required exercises are provided within the course itself. Please read through each course module carefully.

COURSE SCHEDULE

Due Dates

The Academic Week at CSU-Global begins on Monday and ends the following Sunday.

- **Discussion Boards:** The original post must be completed by Thursday at 11:59 p.m. MT and peer responses posted by Sunday 11:59 p.m. MT. Late posts may not be awarded points.
- **Opening Exercises:** Take the Opening Exercise before reading each week's content to see which areas you will need to focus on. You may take these exercises as many times as you need. The Opening Exercises will not affect your final grade.
- **Mastery Exercises:** Students may access and retake Mastery Exercises through the last day of class until they achieve the scores they desire.
- **Critical Thinking:** Assignments are due Sunday at 11:59 p.m. MT.

WEEKLY READING AND ASSIGNMENT DETAILS

MODULE 1

Readings

- Read the following sections in *Jump Start MySQL*:
 - Getting Started with MySQL
 - Storing Data

Opening Exercise (0 points)

Discussion (25 points)

Mastery Exercise (10 points)

Critical Thinking (60 points)

Choose one of the following two assignments to complete this week. Do not do both assignments. Identify your assignment choice in the title of your submission.

Option #1: Configure a MySQL Database

Download the *world-x* database from the following MySQL website: <http://dev.mysql.com/doc/index-other.html>

1. Create a database user named *jump* with password *start*.
2. Load the *world-x* database into your MySQL database instance.

3. Grant *create, drop, alter, insert, update, select, delete, and index* privileges for user *jump* on the *world-x* database.
4. Submit a 2- to 3-page Microsoft Word document containing a screen shot of the tables in the *world-x* database.
5. Provide brief explanation of the MySQL grant command. How is this command used to implement database security?

Your paper should be 2-3 pages in length and conform to *CSU-Global Guide to Writing and APA*. Include at least two scholarly references in addition to the course textbook. The CSU-Global Library is a good place to find these references.

Option #2: Configure a MySQL Database

Download the *classicmodels* database from the following MySQL website: <http://www.mysqltutorial.org/mysql-sample-database.aspx>

1. Create a database user named *jump* with password *start*.
2. Load the *classicmodels* database into your MySQL database instance.
3. Grant *create, drop, alter, insert, update, select, delete, and index* privileges for user *jump* on the *classicmodels* database.
4. Submit a 2- to 3-page Microsoft Word document containing a screen shot of the tables in the *classicmodels* database.
5. Provide brief explanation of the MySQL grant command. How is this command used to implement database security?

Your paper should be 2-3 pages in length and conform to *CSU-Global Guide to Writing and APA*. Include at least two scholarly references in addition to the course textbook. The CSU-Global Library is a good place to find these references.

MODULE 2

Readings

- Read the following sections in *Jump Start MySQL*:
 - Storing Data
 - Creating Tables
 - Data Types and Storage Requirements
- *MySQL/Language/Data Types*. Retrieved from https://en.wikibooks.org/wiki/MySQL/Language/Data_Types.

Opening Exercise (0 points)

Discussion (25 points)

Mastery Exercise (10 points)

Critical Thinking (60 points)

Choose one of the following two assignments to complete this week. Do not do both assignments. Identify your assignment choice in the title of your submission.

Option #1: Changing a MySQL Table Structure

Using the *world_x* database you installed in Module 1 and your MySQL Workbench, create a new table named "independence" with the following attributes (columns):

- A field named "id" which has data type `auto_increment`,
- A field named "country_name" which has data type `varchar(50)`, and
- A field named "independence_date" which has type "date."

After you create the table, run the following SQL command:

```
INSERT INTO independence(contry_name, independence_date)
VALUE ('United States','1776-07-04')
```

Submit a 1-page Microsoft Word document that shows the following:

- The SQL command you used to create the “independence” table and a list of the rows in the “independence” table.
- Explain the purpose of the field named “id.” Why do you not have to include it in an insert statements? How would change the incrementing value to a number greater than one?

Option #2: Changing a MySQL Table Structure

Using the *classicmodels* database you installed in Module 1 and your MySQL Workbench, add a JSON field named “customers_contacts” to the *customers* table with the following keys:

- customerNumber,
- contactLastName, and
- contactFirstName.

Write a SQL statement to insert at least three values for the “customers_contacts” field. Use data from the *customers* table. If you need a hint on how to write the SQL insert statement for a JSON data type, refer to the examples in <http://www.mysqltutorial.org/mysql-json/>.

Submit a 1-page Microsoft Word document that shows the following:

- The SQL command you used to insert data into the “customers_contacts” field and a listing of rows from the *customer* table where the “customers_contacts” column has data.
- What is JSON? What are the advantages of using JSON as a field type? What are the disadvantages of using JSON as a field type?

MODULE 3

Readings

- Read the following section in *Jump Start MySQL*:
 - Retrieving and Updating Data
- *MySQL/Language/Queries/Select*. Retrieved from <https://en.wikibooks.org/wiki/MySQL/Language/Queries#SELECT>

Opening Exercise (0 points)

Discussion (25 points)

Mastery Exercise (10 points)

Critical Thinking (60 points)

Choose one of the following two assignments to complete this week. Do not do both assignments. Identify your assignment choice in the title of your submission.

Option #1: Performing an Advanced MySQL Query

Using the *world_x* database you installed in Module1, list the codes for countries where English is spoken by more than 30% of the population. Submit your query and query results in a Word file.

In your document, suggest another query where you list the country codes for countries where English is spoken by less than 30% of the population. Be sure to identify the literary sources you used to look up any SQL syntax you used to formulate your query.

Option #2: Performing an Advanced MySQL Query

Using the *classicmodels* database you installed in Module 1, list the office codes, phone numbers, and addresses located in San Francisco, Paris, Sydney, and London. Sort your results by office code. Submit your query and query results in a text file.

In your document, suggest another query where you list the same information cities other than San Francisco, Paris, Sydney, and London. Be sure to identify the literary sources you used to look up any SQL syntax you used to formulate your query.

MODULE 4

Readings

- Read the following section in *Jump Start MySQL*:
 - Working with Multiple Tables
- *MySQL/Language/Queries/Joins*. Retrieved from <https://en.wikibooks.org/wiki/MySQL/Language/Queries#Joins>

Opening Exercise (0 points)

Discussion (25 points)

Mastery Exercise (10 points)

Critical Thinking (60 points)

Choose one of the following two assignments to complete this week. Do not do both assignments. Identify your assignment choice in the title of your submission.

Option #1: Working with MySQL Table Joins

Using the *world_x* database you installed in Module 1, list the countries and the capitals of each country. Modify your query to limit the list to those countries where less than 30% of the population speaks English. Be sure to identify the literary sources you used to look up any SQL syntax you used to formulate your query.

Submit your query and query results Word file.

Option #2: Working with MySQL Table Joins

Using the *classicmodels* database you installed in Module 1, list the customers who have not placed any orders. Sort your results by customer number.

- Modify your query to list customers who have placed fewer than five orders.
- Submit your query and query results in a Word file.

MODULE 5

Readings

- Read the following section in *Jump Start MySQL*:
 - Retrieving and Updating Data
 - Aggregate Functions and Grouping
- *MySQL/Language/Queries/Subqueries*. Retrieved from <https://en.wikibooks.org/wiki/MySQL/Language/Queries#Subqueries>

Opening Exercise (0 points)

Discussion (25 points)

Mastery Exercise (10 points)

Critical Thinking (65 points)

Choose one of the following two assignments to complete this week. Do not do both assignments. Identify your assignment choice in the title of your submission.

Option #1: Aggregating MySQL Table Data

Using the *world_x* database you installed in Module 1, list the languages spoken in each country. Order your results by the most popular language. Submit your query and query results in a Word file. Cite any sources you used to research the syntax you needed to formulate your query.

Option #2: Aggregating MySQL Table Data

Using the *classicmodels* database you installed in Module 1, list the total sales for years after 2003. Submit your query and query results in a Word file. Cite any sources you used to research the syntax you needed to formulate your query.

Portfolio Milestone (25 points)

Choose one of the following two assignments to complete this week. Do not do both assignments. Identify your assignment choice in the title of your submission.

Portfolio Milestone Option #1

Make appropriate corrections to the assignments you submitted in Modules 3 and 4 (Critical Thinking Assignments). Resubmit your rework from Modules 3 and 4 with all outlined corrections.

Portfolio Milestone Option #2

Make appropriate corrections to the assignments you submitted in Modules 3 and 4 (Critical Thinking Assignments). Resubmit your rework from Modules 3 and 4 with all outlined corrections.

MODULE 6

Readings

- Read the following section in *Jump Start MySQL*:
 - Connecting from Code
 - Connecting from Python with Connector/Python
- *MySQL. 5.1 Connecting to MySQL using Connector/Python*. Retrieved from <https://dev.mysql.com/doc/connector-python/en/connector-python-example-connecting.html>

Opening Exercise (0 points)

Discussion (25 points)

Mastery Exercise (10 points)

Critical Thinking (65 points)

Choose one of the following two assignments to complete this week. Do not do both assignments. Identify your assignment choice in the title of your submission.

Option #1: Integrating MySQL Table Data with a Python Script

In this assignment, you will write a Python script to increase the population of each city in the *world_x* database by 10% (rounded).

First, add a new column to the *world_x* city table using the following MySQL command:

- `ALTER TABLE `world_x`.`city` ADD COLUMN `Population` DOUBLE NULL DEFAULT NULL AFTER `Info`;`

The existing population data are stored as *JSON* datatype in the city table in a field named *Info*. You learned about JSON data types in Module 2. To obtain the population data from the *Info* column in the city table, run the following MySQL command:

- select info->'\$.Population' from city

Your Python script should do the following:

- Copy the population data from the *Info* column into the *Population* column,
- Increase the population by 10% in the *Population* column, and
- List the population before the increase and the population after it was increased.

Submit your Python script and Python script output as two separate text files included in a zip file. Submit your zip file to the Module 6 folder. Provide a detailed technical report on the steps needed to integrate Python and MySQL. In addition, provide details on how you iterate through a JSON field in a Python script.

Option #2: Integrating MySQL Table Data with a Python Script (65 points)

Using the *classicmodels* database you installed in Module 1, write a Python script that deducts one percent from the amounts paid for purchases made in the month of December. The amounts customers paid are listed in the payments table. The output of your script should show the payments before the rebate and the payments after the rebate.

Submit your Python script and Python script output as two separate text files included in a zip file. Submit your zip file to the Module 6 folder.

Provide a detailed technical report on the steps needed to integrate Python and MySQL. In addition, provide details on how you implement a table join in a Python script. Show an example of such a join.

MODULE 7

Readings

- *MySQLTutorial*. Managing database index. Retrieved from <http://www.mysqltutorial.org/mysql-create-drop-index.aspx>
- *MySQL/Optimization/Optimising the MySQL Server*. Retrieved from https://en.wikibooks.org/wiki/MySQL/Optimization#Optimising_The_MySQL_Server

Opening Exercise (0 points)

Discussion (25 points)

Mastery Exercise (10 points)

Portfolio Milestone (25 points)

Make appropriate corrections to the assignments you submitted in Modules 5 and 6 (Critical Thinking Assignments). Resubmit your rework from Modules 5 and 6 with all outlined corrections.

MODULE 8

Readings

- Read the following section in *Jump Start MySQL*:
 - Backups and Replication
- *MySQL/Replication*. Retrieved from <https://en.wikibooks.org/wiki/MySQL/Replication>

Opening Exercise (0 points)

Discussion (25 points)

Mastery Exercise (10 points)

Portfolio Project (300 points)

Choose one of the following two assignments to complete this week. Do not do both assignments. Identify your assignment choice in the title of your submission.

Option #1:

Your Portfolio Project consists of two items, a lessons learned reflection report and a database design and analysis project.

Lessons Learned and Reflection

Write a 2- to 3-page summary that outlines the lessons you learned in this database class. Reflect on how these lessons can be applied toward more effective database management and reporting.

Database Design and Analysis

Use the Accidents_2016.csv file below (see assignment page) to implement the following steps for your database project.

1. Create a MySQL schema named "accidents."
2. Within the accidents schema, create a table named *accidents_2016* with the following columns:
 - o accident_index as a varchar(13),
 - o accident_severity as an int
3. Within the accidents schema, create a table named *vehicles_2016* with the following columns:
 - o accident_index as a varchar(13),
 - o vehicle_type as a varchar(10)
4. Within the accidents schema, create a table named *vehicle_type* with the following columns:
 - o vcode int,
 - o vtype as a varchar(100)
5. Next, you will load the data for the three tables.
 - o Load the accidents data. Note that *@dummy* is a placeholder for a column in the .csv file that you want to ignore during the load.

```
load data local infile '...\\data\\Accidents_2016.csv'
into table accidents_2016
fields terminated by ','
enclosed by '"'
lines terminated by '\n'
ignore 1 lines
(@col1, @dummy, @dummy, @dummy, @dummy, @dummy, @col2
,@dummy, @dummy, @dummy, @dummy, @dummy
,@dummy, @dummy, @dummy, @dummy, @dummy
,@dummy, @dummy, @dummy, @dummy, @dummy
,@dummy, @dummy, @dummy, @dummy, @dummy
)
set accident_index=@col1,accident_severity=@col2;
o Load the vehicle data.
```

```
load data local infile '...\\data\\Vehicles_2016.csv'
into table vehicles_2016
fields terminated by ','
enclosed by '"'
lines terminated by '\n'
ignore 1 lines
(@col1, @dummy, @dummy, @col2
```



```
,@dummy, @dummy, @dummy, @dummy, @dummy
,@dummy, @dummy, @dummy, @dummy, @dummy
,@dummy, @dummy, @dummy, @dummy, @dummy
,@dummy, @dummy, @dummy, @dummy, @dummy
)
```

```
set accident_index=@col1,vehicle_type=@col2;
```

- Load the vehicle type data.

```
load data local infile '...\data\vehicle_type.csv'
into table vehicle_type
fields terminated by ','
enclosed by '"'
lines terminated by '\n'
ignore 1 lines
```

6. After the data are loaded, you will perform the analysis. First, find the average accident severity and the number of accidents for vehicles of type *motorcycle*. Note the performance of your query. Your query may run so slowly that MySQL aborts running completing.
7. Improve Query Performance
 - Look at the explain tool output and save the results to a graphic file.
 - From the explain results, how many rows have to be read per join?
 - Add an index named “accident_index” of type “index” on the accident_index column in the accidents_2016 table and another index named “accident_index” of type “index” on the vehicles_2106 table.

```
alter table accidents_2016
add index accident_index (accident_index asc);
```

```
alter table vehicles_2016
add index accident_index (accident_index asc);
```

After adding the indices, rerun the query explanation tool and determine the number of rows to be read per join.

8. Find the median accident severity.

MySQL does not have a median function so to find the median accident severity, you will have to write a Python script.

- You'll need to install Python and the PyMySQL module.
- Install Python version 2.7 or 3.4 from www.python.org.

To install the PyMySQL module, run the following command in a Windows command prompt after Python has been installed:

```
python -m pip install --index-url=https://pypi.python.org/simple/ --trusted-host pypi.python.org PyMySQL
```

b) Create an accident median table

```
create table accident_medians
(
    vtype varchar(100),
    severity int
);
```

- Run the following Python script:

```
import pymysql
myConnection = pymysql.connect(host='localhost', user='****', passwd='****', db='accidents')
```

```

cur = myConnection.cursor()
cur.execute('SELECT vtype FROM vehicle_type WHERE vtype LIKE "%otorcycle%";')
cycleList = cur.fetchall()
selectSQL = (''
SELECT t.vtype, a.accident_severity
FROM accidents_2016 AS a
JOIN vehicles_2016 AS v ON a.accident_index = v.Accident_Index
JOIN vehicle_type AS t ON v.Vehicle_Type = t.vcode
WHERE t.vtype LIKE %s
ORDER BY a.accident_severity;''')
insertSQL = (''INSERT INTO accident_medians VALUES (%s, %s);''')

for cycle in cycleList:
    cur.execute(selectSQL,cycle[0])
    accidents = cur.fetchall()
    quotient, remainder = divmod(len(accidents),2)
    if remainder:
        med_sev = accidents[quotient][1]
    else:
        med_sev = (accidents[quotient][1] + accidents[quotient+2][1])/2
    print('Finding median for',cycle[0])
    cur.execute(insertSQL,(cycle[0],med_sev))
myConnection.commit()
myConnection.close()

```

Write each query you used in Steps 1 – 8 in a text file. If a query produced a result set, the list then first ten rows of each row set after the query. Bundle your lessons learned report, your queries and your query results text file, and your MySQL query explanations from both before and after adding table indexes into a single zip file and submit that zip file as your final portfolio. Please be sure to utilize, include and reference in your project at least three sources in APA format.

Option #2

Your Portfolio Project consists of two items, a lessons learned reflection report and a database design and analysis project.

Lessons Learned and Reflection

Write a 2- to 3-page summary that outlines the lessons you learned in this database class. Reflect on how these lessons can be applied toward more effective database management and reporting.

Database Design and Analysis

The following database project will create an educational attainment “demand” forecast for the state of California for years greater than 2010. The demand forecast is the expected number of population who have obtained a certain level of education. The population is divided into age groups and education attainment is divided into different levels. The population of each group is estimated for each year up to year 2050. Implement the following steps to obtain and educational demand forecast for the state of California. The files can be downloaded below (see assignment page).

1. Create a ca_pop schema in your MySQL database.
2. Using your ca_pop schema, create an *educational_attainment* table which columns match the columns in the Excel spreadsheet *ca_pop_educational_attainment.csv*.
3. Using your ca_pop schema, create a *pop_proj* table which columns match the columns in the Excel spreadsheet *pop_proj_1970_2050.csv*.

4. Using the data loading technique for a csv file you learned in Module 1, load the data in *ca_pop_educational_attainment.csv* into the table *educational_attainment*.
5. Using the data loading technique for a csv file you learned in Module 1, load the data in *pop_proj_1970_2050.csv* into the table *pop_proj*.
6. Write a query to select the total population in each age group.
7. Use the query from Step 6 as a subquery to find each type of education attained by the population in that age group and the fraction of the population of that age group that has that educational attainment. Label the fraction column output as *coefficient*. For instance, the fraction of the population in age group 00 - 17 who has an education attainment of Bachelor's degree or higher is 0.0015, which is the *coefficient*.
8. Create a demographics table from the SQL query from Step 7.
9. Create a query on the *pop_proj* table which shows the population count by date_year and age.
10. Use that query from Step 9 as a subquery and join it to the demographics table using the following case statement:

```

demographics.age =
    case
        when temp_pop.age < 18 then '00 to 17'
        when temp_pop.age > 64 then '65 to 80+'
        else '18 to 64'
    end

```

“temp_pop” is an alias for the subquery. Use the following calculation for the demand output:

```
round(sum(temp_pop.total_pop * demographics.coefficient)) as demand
```

Output the demand grouped by year and education level.

Write each query you used in Steps 1 – 10 in a text file. If a query produced a result set, the list then first ten rows of each row set after the query. Bundle your lessons learned report and your query results text file into a single zip file and submit that zip file as your final portfolio. Please be sure to utilize, include and reference in your project at least three sources in APA format.

COURSE POLICIES

Course Grading

20% Discussion Participation
0% Opening Exercises
8% Mastery Exercises
37% Critical Thinking Assignments
5% Portfolio Milestones
30% Final Portfolio Project

Grading Scale	
A	95.0 – 100
A-	90.0 – 94.9
B+	86.7 – 89.9
B	83.3 – 86.6
B-	80.0 – 83.2
C+	75.0 – 79.9
C	70.0 – 74.9
D	60.0 – 69.9
F	59.9 or below

IN-CLASSROOM POLICIES

For information on late work and incomplete grade policies, please refer to our [In-Classroom Student Policies and Guidelines](#) or the Academic Catalog for comprehensive documentation of CSU-Global institutional policies.

Academic Integrity

Students must assume responsibility for maintaining honesty in all work submitted for credit and in any other work designated by the instructor of the course. Academic dishonesty includes cheating, fabrication, facilitating academic dishonesty, plagiarism, reusing /repurposing your own work (see *CSU-Global Guide to Writing and APA Requirements* for percentage of repurposed work that can be used in an assignment), unauthorized possession of academic materials, and unauthorized collaboration. The CSU-Global Library provides information on how students can avoid plagiarism by understanding what it is and how to use the Library and Internet resources.

Citing Sources with APA Style

All students are expected to follow the *CSU-Global Guide to Writing and APA Requirements* when citing in APA (based on the APA Style Manual, 6th edition) for all assignments. For details on CSU-Global APA style, please review the APA resources within the CSU-Global Library under the “APA Guide & Resources” link. A link to this document should also be provided within most assignment descriptions in your course.

Disability Services Statement

CSU-Global is committed to providing reasonable accommodations for all persons with disabilities. Any student with a documented disability requesting academic accommodations should contact the Disability Resource Coordinator at 720-279-0650 and/or email ada@CSUGlobal.edu for additional information to coordinate reasonable accommodations for students with documented disabilities.

Netiquette

Respect the diversity of opinions among the instructor and classmates and engage with them in a courteous, respectful, and professional manner. All posts and classroom communication must be conducted in accordance with the student code of conduct. Think before you push the Send button. Did you say just what you meant? How will the person on the other end read the words?

Maintain an environment free of harassment, stalking, threats, abuse, insults, or humiliation toward the instructor and classmates. This includes, but is not limited to, demeaning written or oral comments of an ethnic, religious, age, disability, sexist (or sexual orientation), or racist nature; and the unwanted sexual advances or intimidations by email, or on discussion boards and other postings within or connected to the online classroom. If you have concerns about something that has been said, please let your instructor know.