University of Maryland University College

School of Undergraduate Studies

Faculty Teaching Guide

CMIS 320

Relational Database Concepts and Applications

About CMIS 320

Listed below are important concepts and highlights about the nature of this course and its students.

- Students should have completed a basic programming concepts course prior to this course.
- The focus for this course is on ERD, normalization, and very basic SQL (create tables, select, insert, update, and delete). Joins are not covered.
- This course services CMIS, CMSC, IFSM, CMST, CMIT, and CSIA computing majors, and potentially other non-computing majors. There will be a wide diversity of technical background, and extra assistance and reaching out will be required to help students.

Using this Guide

This guide is designed to help faculty members teach the course effectively as well as to help the department maintain the appropriate alignment with the program curriculum. The guide and the course syllabus should be provided to faculty members teaching the course and should be regularly reviewed and updated. When you have suggestions and recommendations for the guide, see the corresponding 999 conference discussion.

How can you use this guide to help you teach your course? It's a long-term resource to consult as you progress, but here are some immediate ideas to:

- **Plan the flow of your class—**Look at the assignments and activities in the guide—they are extended descriptions of what you will find in the model syllabus, and will help you think about timing.

- **Introduce and explain your course to students—**Deliberately point to the intended course outcomes—you'll find them right after the course description. Also be sure to explain the level and place of the course and the kind of activities the students will experience. Students need to understand the focus and purpose of their learning.

- **Choose topics for class discussions—**Besides the outcomes, look at the Learning Activities section, which may give you ideas for discussions or techniques. The Required Concepts, Skills, and Issues to Be Covered section will also remind you of content to present to achieve the outcomes.

- **Explain class assignments—**Sample language is provided for major assignments, as well as assistance with grading weights and criteria. Feel free to cut and paste this language into your explanations. The assignments are designed to fit coherently with the course design. Make sure you specify the course outcomes for each assignment. This will show students there is purpose in the assignment.

- **Monitor student progress—**The course design and guide are planned for a sequence to achieve the learning outcomes. Look at how assignments are split up or staged in the syllabus and the suggestions in the guide for evaluating student work. You may want to institute subprojects, such as deadlines for drafts or pieces of larger projects, to let you know if students are ready for the big assignments.

- **Make sure you are teaching at the right level—**Besides the explanation of the place and level of the course at the beginning of the guide, review the Bloom's taxonomy chart at the end, which gives a quick picture of the cognitive level for this course. It can help you make sure you aren't teaching an introductory course at a level beyond your students' abilities or, conversely, teaching an advanced course at too low a level.

Course Number and Title

CMIS 320: *Relational Database Concepts and Applications*

Course Description

Prerequisite: CMIS 102 or CMIS 141. A study of the functions, underlying concepts, and applications of enterprise relational database management systems (RDBMS) in a business environment. The aim of the course is to appropriately use databases to meet business requirements. Discussion covers entity/relationship diagrams, relational theory, normalization, integrity constraints, the Structured Query Language (SQL), and physical and logical design. Business case studies and projects include hands-on work using an industry-standard RDBMS. Students may receive credit for only one of the following courses: CMIS 320 or IFSM 410.

Intended Course Outcomes

List the learning outcomes for this course (phrased as "After completing this course, students should be able to ...").

1. evaluate options to make informed decisions that meet data storage, processing, and retrieval needs
2. design and document data structures incorporating integrity constraints to satisfy business rules by applying the relational model
3. build, populate, and document a secure, normalized database that meets business requirements using industry standards and best practices
4. develop structured query language (SQL) queries to create, read, update, and delete relational database data

Position of Course in Curriculum

- **Role in Program and Degree Requirements**

How does this course relate to requirements (e.g., required course, one of several choices for a specific requirement, supplemental course for major, elective)?

This is the introductory relational database course at UMUC. It is a required course for CMIS and IFSM majors. It serves as an elective for other majors.

- **Program Outcomes to Which this Course Maps**

To which program-level outcomes does this course contribute? How does it contribute?

P1. Design, implement, secure, and maintain databases that meet user requirements for both transaction processing and data warehouses.

P5. Identify, learn, and adapt to local and global IT trends, technologies, legalities, and policies, as well as appropriately communicate their impact to key stakeholders.

P6. Work independently or as an effective member of an application development team to determine and implement systems that meet customer requirements.

This course covers the design and maintenance of databases and transactional systems based on user requirements, using best practices and keeping current with IT trends.

- **Relation to Hallmark Competencies, if Appropriate**

Hallmarks are the core competencies that UMUC assures each student will accomplish by the end of their degree. The hallmarks are introduced, reinforced, and emphasized in general education courses as well as the program. Which hallmarks are addressed in this course (effective writing, information literacy, technology fluency, critical thinking, ethics, quantitative literacy, scientific literacy, historical and cultural perspectives)?

H2. Written Communications

H5. Critical Thinking

H6. Technology Fluency

- **Level of Course (100, 200, 300, or 400)**

Why is the stated level appropriate for this course?

This is a 300-level course. Projects and activities include analysis, design, and higher-level critical thinking based on previous foundational courses.

- **Relation to or Sequencing with Other Courses**

How does this course relate to other courses (e.g., to prerequisites; as a prerequisite; where the course will fall in a recommended sequence; role as part of a group of courses)?

Students will take this course to learn relational databases. Students should have had at least one introduction to programming concepts course prior to registering for this course. Students can take CMIS 420, CMIS 430, or CMIS 485 after completing this course.

- **Importance of the Course to Other Majors and/or Disciplines**

How does this course contribute to programs outside this major (e.g., requirement, related requirement or recommended elective for another major, fulfills general education requirement)?

CMIS 320 primarily supports the CMIS and IFSM programs, but students in other disciplines can take this course if they want to learn about relational databases, provided they have a proper prerequisite.

Approach to Course

- **Required Concepts, Skills, and Issues to Be Covered**

List the concepts, skills, and issues that must be included as central to the course (versus optional material that may be included depending on time and instructor/student interest).

As part of the redesign process, the following concepts, skills, and issues were identified as important to student success. Faculty members should design the overall course, reading assignments, weekly discussions, activities, and projects to address these in an appropriate manner to ensure that students are prepared to accomplish the graded activities. These have been arranged in a suggested weekly sequence; however, faculty should align with assigned readings and in a way that best supports the assignments and course activities.

Date	Assignments	Due Date
Week 1	**Concepts** • what a database does • types of databases – various applications • types of data • privacy • user roles • what business rules are • business analysis **Skills** • business/process analysis • communication **Issues** • time management • environment configuration	
Week 2	**Concepts** • relational model • data modeling tools • industry standards and best practices • database life cycle • integration with existing systems • transactions • logical vs. physical modeling • what an ERD consists of • what is an ERD? **Issues** • normalization • modeling—abstract aspects	

Week 3	**Concepts** • normalization • data types information representations • integrity constraints	
Week 4	**Concepts** • spool files • debugging and troubleshooting • create, read, update, delete • using SQL • DML vs. DDL **Skills** • translating an ERD into table specifications • debugging • SQL*plus • iSQL*plus • creating spool files • SQL commands and functions **Issues** • UNIX	
Week 5	**Concepts** • security requirements **Issues** • group dynamics	
Week 6	**Concepts** • data warehousing • data mart • storage considerations • analysis of alternatives • ROI/cost considerations **Skills** • researching technical aspects of systems • writing a memo • presenting information with tradeoffs	
Week 7	**Skills** • weighted analysis • group dynamics • drawing logical conclusions	

Week 8	**Concepts** • what comes next? • emerging and evolving tools and trends • ethical considerations	

Assessment and Learning Activities

- **Assessments, Projects, and Assignments**

List and describe the planned projects/assignments for this course that will fulfill the course outcomes and allow for appropriate assessment. Mark mandatory items with an asterisk (*). If an item is included in the program plan for assessment of a program-level outcome or a hallmark outcome, mark and specify the assessment activity to be required for that purpose as well (e.g., common final exam).

- conferences*
- respond to a business problem with memo paper and presentation (group project): post and discuss one another's projects/presentations*
- final exam*
- project 1: ERD + data dictionary*
- project 2: build, populate, query*

Faculty members are free to slightly modify the assignments' contributions to the final grade by plus or minus 5 percent. For example, weekly conferences are listed as 15 percent in the syllabus; however, faculty members are free to modify this level, provided they stay in the range between 10 percent and 20 percent.

Project Descriptions

Homework

Your instructor will provide homework assignments related to the weekly reading assignments as shown in the schedule. You need to complete each assignment no later than the due date and submit to your assignments folder for grading.

Project 1

This project allows students to demonstrate their skills in the area of designing relational databases to satisfy specific business rules and requirements. The deliverables for this project include an Entity Relationship Diagram and detailed documentation describing the database design and structure.

Project 1 Details:

In this project you will be provided a set of specifications to create an entity-relationship diagram and design accompanying table layout using sound relational modeling concepts and practices.

The relationships between the entities and the attributes for the entities will be identified and described. This database will provide the foundation for the follow-on project. The following paragraphs provide the background and summary of the business requirements.

You are a database consultant with Ace Software, Inc. and have been assigned to develop a database for the Mom and Pop Johnson video store in town. Mom and Pop have been keeping their records of videos and DVDs purchased from distributors and rented to customers in stacks of invoices and piles of rental forms for years. They have finally decided to automate their record keeping with a relational database.

You sit down with Mom and Pop to discuss their business, and watch their operation for about a week. You discover quickly that a video and a DVD are both copies of a movie kept in a separate plastic case that is rented out. They have several copies of each movie they rent; therefore there are several videos and DVDs for each movie title. You learn that in their inventory they have several thousand videos and DVDs, which they get wholesale from about a half dozen distributors. The video and DVD prices to them are based on the quantity of their shipment and the past business they have done with each company.

The price of a DVD for a movie might be different than the price of a video for the same movie, even from the same distributor. Each distributor provides different types of movies (e.g., suspense, horror, mystery, comedy, etc.). A single distributor may provide several different types of movies in both video and DVD format. It is possible to obtain the same movie from multiple distributors, and at different wholesale prices.

Each video and DVD has a unique identification number that Mom and Pop assign in their inventory, in addition to the distributor's serial number for the item. Each movie also has a unique identification number Mom and Pop assign in addition to the title, and any movie IDs the distributors use in their electronic catalogs. Distributors provide electronic catalogs to Mom and Pop and the information from these catalogs must be included in the database.

Mom and Pop need to record when a video or DVD is rented, when a video or DVD is returned, and all customer charges such as late and damaged fees, failure to rewind fees, and taxes. They need a report of which videos are returned late because there are standard and late charges. On occasion there are discount prices for certain movies or types of movies. Customers want to rent movies based on actors or actresses, running length, type of movie, rating, year released, the director, and the Academy awards won (by the movie, the actors, the actresses, and/or the directors). Customers also want to know how many videos they have rented in the last month, year, and so forth. Mom and Pop need to keep only basic information on customers in their database, such as name, address, telephone numbers, etc.

There must be no limit to the number of video and/or DVD copies of a movie that Mom and Pop can have in their inventory. Video/DVD ID numbers, movie ID numbers, and distributor ID numbers for videos, DVDs, and movies are all different. Also, each movie must be able to have an unlimited number of actors, actresses, directors, and Academy awards (i.e., Oscars). Other types of awards (e.g., Golden Globe, People's Choice, etc.) are not of interest for

this application. The rental of equipment, sale of videos, DVDs, popcorn, etc., is not to be kept in the database.

Using this information, you should:

1. Determine and list your **entities**. Then describe fully the **relationships** between entities via pairs of sentences that indicate the two components of the total relationship in both directions between the entities. Relationships may be unary, binary, or ternary with respect to entities. You should not have any many-to-many relationships.

2. Begin the database logical design by identifying the entities, relationships between entities, and entities' attributes as we have done in this course. Use the same **entity/relationship diagram** (ERD) notation as used in class for entities, attributes, and relationships. Sketch your ERD by hand or a drawing program (e.g., Visio, PPT, SQL Modeler …) on **one** single 8-1/2" x 11" page (8-1/2" x 14" maximum), labeled "Mom and Pop Johnson Video Store Database E/R Diagram." Your ERD should not have any many-to-many relationships between entities. Make sure that all relationships between entities are one-to-many to facilitate the construction of relational database tables.

3. Complete the logical database design and start the physical database design by creating metadata (i.e., documentation) that describes the **table(s)** created from each entity and the **column(s)** created from each attribute in the ERD. Attributes should be self-describing. Particular attention will be given to the proper specification of all primary key (via "PK") and foreign key (via "FK") columns in the table layouts. These should match your ERD exactly. Begin these descriptions on a page labeled "Proposed Database Tables and Columns based on E/R Diagram." All tables must be 3rd Normal Form. Indicate any and all assumptions that were made.

4. Make sure your work is neat and legible.

Submissions:

1. You should submit a well-organized Word document that includes your entity list, ERD, and database design description. This document should also include all assumptions and your lesson learned during your project.
2. Submit in your WebTycho portfolio in the Project 1 area before the due date. Please contact the instructor **before** the due date if you have any questions or concerns.

Project 2

In this project you will perform the **physical** design and implementation using SQL Data Definition Language (DDL) and proceed with **populating** the Mom and Pop Johnson Video Store database via Data Manipulation Language (DML) SQL commands. The ERD and detailed documentation from Project 1 will be supplied for your entry point for those who need it.

Project 2 Details:

1. **Create Oracle database tables** using SQL Data Definition Language (DDL) for each table listed in Project 1. Make sure that entity and referential integrity are enforced by declaring a **primary** key for **each** table (these may be composite keys) and declaring **all** appropriate **foreign** keys. Your CREATE TABLE statements must show integrity constraints, as appropriate, for NOT NULL, UNIQUE, PRIMARY KEY, FOREIGN KEY, REFERENCES, and CHECK constraints. Be sure to save your script used to create these tables as yournameproject2step1.sql. You should test your script to make sure it runs without error.
2. Provide two examples of SQL DML (i.e., "INSERT") commands that fail **different** table integrity constraints you set up in one of your tables. Explain why the statements fail. Be sure to save your script used as yournameproject2step2.sql. You can include comments in the SQL script describing why the insert statements failed.
3. Populate each of your tables with at least five **valid** rows of data each and show the SQL you used. Populate other tables in your database, as necessary, to satisfy referential integrity. Be sure to save your script used to create these records as yournameproject2step3.sql. You should test your script to make sure it runs without error.
4. Write SQL to perform the following queries and updates. Be sure to save your script used to create these records as yournameproject2step4.sql. You should test your script to make sure it runs without error:
    a. Retrieve all of your customers' names, account numbers, and addresses (street and zip code only), sorted by account number.
    b. Retrieve all of the videos rented in the last 30 days and sort in chronological rental date order.
    c. Produce a list of your distributors and all their information sorted in order by company name.
    d. Update customer names to change their maiden names to married names. You can choose which records to update.
    e. Delete customers from the database. You can choose which records to delete.

Submissions:

1. You should submit your four SQL scripts satisfying each of the steps above. Feel free to compress the files using WinZip and save as YournameProject2.zip.
2. Submit in your WebTycho portfolio in the Project 2 area before the due date. Please contact the instructor **before** the due date if you have any questions or concerns.

Relational Database Business Problem Short Paper and Discussion: You will write a two-page (single-spaced) executive summary for your boss explaining how a relational data solution can be applied to a current business problem or area for improvement. Assume that your boss knows nothing about relational theory. The goal of this summary is to obtain your boss's approval to proceed with your stated project. Do not focus on technical aspects of a database management system. Focus on how the information will be captured, manipulated, managed, and shared, and the value the database brings to the organization. Include brief examples of how other industries (both domestic and international) have successfully used relational databases to increase efficiency. Be sure to reference all sources properly using APA guidelines.

Submissions:

1. You should submit your well-organized Word document in the WebTycho portfolio in the Short Paper area before the due date.
2. You should name your paper Yournameshortpaper.doc

- **Learning Activities**

List and describe the learning activities for this course that will address learning outcomes and fulfill the principles of the SUS learning model (e.g., working in study groups or learning collaboratively in small groups; investigating and discussing a resource; responding to material or taking a quiz in an online course module; reviewing and commenting on another student's draft; revising a draft based on specific criteria). These may or may not be graded activities. Specify required learning activities as well as suggested default learning activities that can be used at the instructor's option.

Weekly conferences should be related to the concepts and allow students to comment on one another's work and contributions.

- **Cognitive Level and Assessing Course Outcomes**

Use the table on the following page to indicate the level at which each course outcome will be addressed and assessed in the course as per Bloom's taxonomy. (See more on Bloom's taxonomy below the table.) Mark the appropriate cell with an X to indicate the level for the outcome.

Course Outcomes Mapped to Bloom's Taxonomy

Bloom's taxonomy is a tool for classifying the cognitive demand level of instructional activities or questions. As one moves through the hierarchy from knowledge to evaluation, the activities and questions require increasingly higher-level thinking skills. See the list below this table for a summary of Bloom's taxonomy and examples of verbs corresponding to each cognitive level.

	Cognitive Demand Level – Based on Bloom's Taxonomy					
Course Outcome	**Knowledge (Lowest)**	**Comprehension**	**Application**	**Analysis**	**Synthesis**	**Evaluation (Highest)**
1. evaluate options to make informed decisions that meet data storage, processing, and retrieval needs						X
2. design and document data structures incorporating integrity constraints to satisfy business rules by applying the relational model			X			
3. build, populate, and document a secure, normalized database that meets business requirements using industry standards and best practices			X			
4. develop structured query language (SQL) queries to create, read, update, and delete relational database data			X			

Bloom's Taxonomy Terms

Source: Wyatt, A. T. (2001). *Bloom's taxonomy*. Retrieved September 11, 2006, from http://cs1.mcm.edu/~awyatt/csc3315/bloom.htm

Level	Type of Activity or Question	Verbs Used for Outcomes
Lowest level	**Knowledge**	define, memorize, repeat, match, record, list, recall, name, relate, collect, label, specify, cite, enumerate, recite, tell, recount
	Comprehension	restate, summarize, differentiate, discuss, describe, recognize, explain, express, identify, locate, report, retell, review, translate, paraphrase
	Application	exhibit, solve, manipulate, interview, simulate, apply, employ, use, demonstrate, dramatize, practice, illustrate, operate, calculate, show, experiment
Higher levels	**Analysis**	interpret, classify, analyze, arrange, differentiate, group, compare, organize, contrast, examine, scrutinize, survey, categorize, dissect, probe, create an inventory, investigate, question, discover, inquire, distinguish, detect, diagram, chart, inspect
	Synthesis	compose, set up, plan, prepare, propose, imagine, produce, hypothesize, invent, incorporate, develop, generalize, design, originate, formulate, predict, arrange, assemble, construct, create
	Evaluation	judge, assess, decide, measure, appraise, estimate, evaluate, rate, deduce, compare, score, value, predict, revise, choose, conclude, recommend, determine, criticize, test

Appendix A

CMIS 320: *Relational Database Concepts and Applications*

Course Syllabus

Faculty Contact Information

Course Materials

Coronel, Carlos; Morris, Steven; & Rob, Peter. (2009). *Database Systems: Design, Implementation, and Management*, 9th edition. Cengage. ISBN: 978-0-538-46968-5.

Course Description

Prerequisite: CMIS 102 or CMIS 141. A study of the functions, underlying concepts, and applications of enterprise relational database management systems (RDBMS) in a business environment. The aim of the course is to appropriately use databases to meet business requirements. Discussion covers entity/relationship diagrams, relational theory, normalization, integrity constraints, the Structured Query Language (SQL), and physical and logical design. Business case studies and projects include hands-on work using an industry-standard RDBMS. Students may receive credit for only one of the following courses: CMIS 320 or IFSM 410.

Course Outcomes

After completing this course, you should be able to

- evaluate options to make informed decisions that meet data storage, processing, and retrieval needs
- design and document data structures incorporating integrity constraints to satisfy business rules by applying the relational model
- build, populate, and document a secure, normalized database that meets business requirements using industry standards and best practices
- develop structured query language (SQL) queries to create, read, update, and delete relational database data

Course Introduction

This course is a study of the underlying concepts and functions of relational databases. Relational model theory—including relational structures, integrity constraints, data manipulation, and normalization—is emphasized. Data modeling is described, and its application to relational and object relational database systems is discussed. Physical design and implementation issues are addressed. Projects include hands-on work with entity/relationship (E/R) diagramming, SQL (using Oracle), and normalization of data in tables.

Grading Information and Criteria

This course consists of the following graded items:

Participation	15%
Relational DB business problem short paper and discussion	10%
Project 1: Relational Database Design and Modeling	25%
Project 2: Relational Database SQL and implementation	25%
Homework	25%
Total	**100%**

The grading scale, based on 100 points, is as follows:

A =	90-100
B =	80–89
C =	70–79
D =	60–69
F =	0–59

Participation

By registering for a Web-based course, you have made a commitment to participate in course conferences as well as other online activities. Plan to participate regularly. Participation for this course is defined as proactive discussion in weekly conferences and discussion questions. This requires you to actively reflect on weekly readings and to develop original ideas in your responses. You are expected to demonstrate critical thinking and your understanding of the content in the assigned readings as they relate to the issues identified in the conference discussion.

You are expected to respond to a main topic each week and read other student posts. You are encouraged to respond to other students as well as to your instructor. Note that your online conference participation counts significantly toward your final grade.

When communicating with others in this class, always work to be respectful.

Other Information

Project Descriptions

Homework

Your instructor will provide homework assignments related to the weekly reading assignments as shown in the schedule. You need to complete each assignment no later than the due date and submit to your assignments folder for grading.

Project 1

This project allows students to demonstrate their skills in the area of designing relational databases to satisfy specific business rules and requirements. The deliverables for this project include an Entity Relationship Diagram and detailed documentation describing the database design and structure.

Project 1 Details:

In this project you will be provided a set of specifications to create an entity-relationship diagram and design accompanying table layout using sound relational modeling concepts and practices. The relationships between the entities and the attributes for the entities will be identified and described. This database will provide the foundation for the follow-on project. The following paragraphs provide the background and summary of the business requirements.

You are a database consultant with Ace Software, Inc. and have been assigned to develop a database for the Mom and Pop Johnson video store in town. Mom and Pop have been keeping their records of videos and DVDs purchased from distributors and rented to customers in stacks of invoices and piles of rental forms for years. They have finally decided to automate their record keeping with a relational database.

You sit down with Mom and Pop to discuss their business, and watch their operation for about a week. You discover quickly that a video and a DVD are both copies of a movie kept in a separate plastic case that is rented out. They have several copies of each movie they rent; therefore there are several videos and DVDs for each movie title. You learn that in their inventory they have several thousand videos and DVDs, which they get wholesale from about a half dozen distributors. The video and DVD prices to them are based on the quantity of their shipment and the past business they have done with each company.

The price of a DVD for a movie might be different than the price of a video for the same movie, even from the same distributor. Each distributor provides different types of movies (e.g., suspense, horror, mystery, comedy, etc.). A single distributor may provide several different types of movies in both video and DVD format. It is possible to obtain the same movie from multiple distributors, and at different wholesale prices.

Each video and DVD has a unique identification number that Mom and Pop assign in their inventory, in addition to the distributor's serial number for the item. Each movie also has a unique identification number Mom and Pop assign in addition to the title, and any movie IDs the distributors use in their electronic catalogs. Distributors provide electronic catalogs to Mom and Pop and the information from these catalogs must be included in the database.

Mom and Pop need to record when a video or DVD is rented, when a video or DVD is returned, and all customer charges such as late and damaged fees, failure to rewind fees,

and taxes. They need a report of which videos are returned late because there are standard and late charges. On occasion there are discount prices for certain movies or types of movies. Customers want to rent movies based on actors or actresses, running length, type of movie, rating, year released, the director, and the Academy awards won (by the movie, the actors, the actresses and/or the directors). Customers also want to know how many videos they have rented in the last month, year, and so forth. Mom and Pop need to keep only basic information on customers in their database, such as name, address, telephone numbers, etc.

There must be no limit to the number of video and/or DVD copies of a movie that Mom and Pop can have in their inventory. Video/DVD ID numbers, movie ID numbers, and distributor ID numbers for videos, DVDs, and movies are all different. Also, each movie must be able to have an unlimited number of actors, actresses, directors, and Academy awards (i.e., Oscars). Other types of awards (e.g., Golden Globe, People's Choice, etc.) are not of interest for this application. The rental of equipment, sale of videos, DVDs, popcorn, etc., is not to be kept in the database.

Using this information, you should:

1. Determine and list your **entities**. Then describe fully the **relationships** between entities via pairs of sentences that indicate the two components of the total relationship in both directions between the entities. Relationships may be unary, binary, or ternary with respect to entities. You should not have any many-to-many relationships.

2. Begin the database logical design by identifying the entities, relationships between entities, and entities' attributes as we have done in this course. Use the same **entity/relationship diagram** (ERD) notation as used in class for entities, attributes, and relationships. Sketch your ERD by hand or a drawing program (e.g., Visio, PPT, SQL Modeler …) on **one** single 8-1/2" x 11" page (8-1/2" x 14" maximum), labeled "Mom and Pop Johnson Video Store Database E/R Diagram." Your ERD should not have any many-to-many relationships between entities. Make sure that all relationships between entities are one-to-many to facilitate the construction of relational database tables.

3. Complete the logical database design and start the physical database design by creating metadata (i.e. documentation) that describes the **table(s)** created from each entity and the **column(s)** created from each attribute in the ERD. Attributes should be self-describing. Particular attention will be given to the proper specification of all primary key (via "PK") and foreign key (via "FK") columns in the table layouts. These should match your ERD exactly. Begin these descriptions on a page labeled "Proposed Database Tables and Columns based on E/R Diagram." All tables must be 3rd Normal Form. Indicate any and all assumptions that were made.

4. Make sure your work is neat and legible.

Submissions:

1. You should submit a well-organized Word document that includes your entity list, ERD, and database design description. This document should also include all assumptions and your lesson learned during your project.
2. Submit in your WebTycho portfolio in the Project 1 area before the due date. Please contact the instructor **before** the due date if you have any questions or concerns.

Project 2

In this project you will perform the **physical** design and implementation using SQL Data Definition Language (DDL) and proceed with **populating** the Mom and Pop Johnson Video Store database via Data Manipulation Language (DML) SQL commands. The ERD and detailed documentation from Project 1 will be supplied for your entry point for those who need it.

Project 2 Details:

1. **Create Oracle database tables** using SQL Data Definition Language (DDL) for each table listed in Project 1. Make sure that entity and referential integrity are enforced by declaring a **primary** key for **each** table (these may be composite keys) and declaring **all** appropriate **foreign** keys. Your CREATE TABLE statements must show integrity constraints, as appropriate, for NOT NULL, UNIQUE, PRIMARY KEY, FOREIGN KEY, REFERENCES, and CHECK constraints. Be sure to save your script used to create these tables as yournameproject2step1.sql. You should test your script to make sure it runs without error.
2. Provide two examples of SQL DML (i.e., "INSERT") commands that fail **different** table integrity constraints you set up in one of your tables. Explain why the statements fail. Be sure to save your script used as yournameproject2step2.sql. You can include comments in the SQL script describing why the insert statements failed.
3. Populate each of your tables with at least five **valid** rows of data each and show the SQL you used. Populate other tables in your database, as necessary, to satisfy referential integrity. Be sure to save your script used to create these records as yournameproject2step3.sql. You should test your script to make sure it runs without error.
4. Write SQL to perform the following queries and updates. Be sure to save your script used to create these records as yournameproject2step4.sql. You should test your script to make sure it runs without error:
    a. Retrieve all of your customers' names, account numbers, and addresses (street and zip code only), sorted by account number.
    b. Retrieve all of the videos rented in the last 30 days and sort in chronological rental date order.
    c. Produce a list of your distributors and all their information sorted in order by company name.
    d. Update customer names to change their maiden names to married names. You can choose which records to update.
    e. Delete customers from the database. You can choose which records to delete.

Submissions:

1. You should submit your four SQL scripts satisfying each of the steps above. Feel free to compress the files using WinZip and save as YournameProject2.zip.
2. Submit in your WebTycho portfolio in the Project 2 area before the due date. Please contact the instructor **before** the due date if you have any questions or concerns.

Relational Database Business Problem Short Paper and Discussion: You will write a two-page (single-spaced) executive summary for your boss explaining how a relational data solution can be applied to a current business problem or area for improvement. Assume that your boss knows nothing about relational theory. The goal of this summary is to obtain your boss's approval to proceed with your stated project. Do not focus on technical aspects of a

database management system. Focus on how the information will be captured, manipulated, managed, and shared, and the value the database brings to the organization. Include brief examples of how other industries (both domestic and international) have successfully used relational databases to increase efficiency. Be sure to reference all sources properly using APA guidelines.

Submissions:

1. You should submit your well-organized Word document in the WebTycho portfolio in the Short Paper area before the due date.
2. You should name your paper Yournameshortpaper.doc

Academic Policies

Academic Integrity

UMUC is an academic community that honors integrity and respect for others, and it is expected that, as a member of this community, you will maintain a high level of personal integrity in your academic work at all times.

Academic dishonesty is the failure to maintain academic integrity, and includes the intentional or unintentional presentation of another person's ideas or products as your own (plagiarism); the use or the attempt to make use of unauthorized materials, information, or study aids in any academic exercise; and the performance of work for another student (cheating). All academic work you submit during your time at UMUC must be original and must not be reused in other courses.

Turnitin.com

The university has a license agreement with Turnitin, an educational tool that helps identify and prevent plagiarism from Internet resources. I may use the service in class, either by requiring you to submit assignments electronically to Turnitin, by submitting assignments on your behalf, or by providing the option for you to check your own work for originality. The Turnitin Originality Report will indicate the amount of original text in your work and whether all the material that you quoted, paraphrased, summarized, or used from another source is appropriately referenced.

If you or I submit all or part of your assignment to the Turnitin service, Turnitin will by default store that assignment in its database. The assignment will be checked for any matches between your work and other material stored in Turnitin's database. If you object to the long-term storage of your work in the Turnitin database, you must let me know no later than two weeks after the start of this class.

You have three options regarding the storage of your assignment in the Turnitin database: 1) You can do nothing; your assignment will then be stored in the Turnitin database for the duration of UMUC's contract with Turnitin; 2) You can ask me to have Turnitin store your assignment only for the duration of the semester or term, then have your assignment deleted from the Turnitin database once the class is over; or 3) You can ask me to change the Turnitin settings so that your assignment is not stored in the Turnitin database at all.

Please note: I may use other services in addition to or in place of Turnitin to check your work for plagiarism.

Course Expectations

For an eight-week course, you should expect to spend about six hours per week participating in class discussions and activities (online or onsite) and two to three times that number of hours outside class in study, assigned reading, and preparation of assignments. Courses offered in shorter formats will require more time per week. You are expected to meet the same learning outcomes and perform the same amount of work in an online course as in an onsite course. Active participation is required in all online courses, and you should expect to log in to your online course several times a week.

The following links to important academic policies and other information are provided to help you as you complete your coursework at UMUC.

Policies and Procedures

- Policy and Procedures on Affirmative Action, Equal Opportunity, and Sexual Harassment—Nondiscrimination: It is the policy of UMUC that no student or employee of the university or contractor/vendor conducting business with the university may discriminate on the basis of race, religion, color, creed, sex (including sexual harassment), marital status, age, national origin, political affiliation, mental or physical disability, or sexual orientation. Individuals who believe they have been discriminated against because of any factor protected under this policy may file a complaint of discrimination.
- Information on Support for Disabled Students
- University System of Maryland Board of Regents' Policy on Academic Integrity
- UMUC's Policy on Academic Dishonesty and Plagiarism
- UMUC's Policy on the Grade of Incomplete, Grade Pending, and Withdrawal
- UMUC's Policy on the Code of Student Conduct
- UMUC's Policy and Procedures for Review of Alleged Arbitrary and Capricious Grading

For more information about student services and more general information, visit UMUC's website at http://www.umuc.edu.

Faculty Bio

Eight-Week Course Schedule

Date	Assignments	Due Date
Week 1	**Read:** • E&N chapters 1-2 • Module 1 **Do:** • Actively participate in conferences and discussions • Homework 1	
Week 2	**Read:** • E&N chapter 3 • Module 2, sections 1-2 **Do:** • Actively participate in conferences and discussions • Homework 2	
Week 3	**Read:** • E&N chapter 7 • Modules 3, 5 **Do:** • Actively participate in conferences and discussions • Homework 3	
Week 4	**Read:** • E&N chapters 8-10 • Modules 4-5 **Do:** • Actively participate in conferences and discussions • Project 1: ERD + data dictionary	
Week 5	**Read:** • E&N chapters 4 • Module 4 **Do:** • Actively participate in conferences and discussions • Homework 4	
Week 6	**Read:** • E&N chapter 5 **Do:** • Actively participate in conferences and discussions • Project 2: build, populate, query	

Week 7	**Read:** • E&N chapter 11 **Do:** • Actively participate in conferences and discussions • Respond to a business problem memo paper and presentation	
Week 8	**Do:** • Actively participate in conferences and discussions • Homework 5	

Course Outcomes

1.	evaluate options to make informed decisions that meet data storage, processing, and retrieval needs
2.	design and document data structures incorporating integrity constraints to satisfy business rules by applying the relational model
3.	build, populate, and document a secure, normalized database that meets business requirements using industry standards and best practices
4.	develop structured query language (SQL) queries to create, read, update, and delete relational database data