
Comprehensive Database Projects

Introduction

- There are three projects:
 - Project 1: Database Design
 - Project 2: Database Build
 - Project 3: MVC Applications
- Projects are completed on a weekly basis based on the sequence of chapter content. The schedule provided is based on a 15-week term with a study week in week 8. The instructor may adjust the weekly schedule based on the number of weeks in the term
- The requirements outlined in the projects below are the minimum requirements. Your instructor may require additional requirements.
- Weekly submissions are due Friday 8:00 p.m. on the week specified
- Weekly submissions are meant to keep the projects on schedule
- Grading:
 - Weekly submissions are graded on a submitted/not submitted basis
 - Final grades for the projects are based on a marking scheme. A suggested marking scheme is provided at the end of this appendix and the instructor may use this marking scheme or a modified version.
 - Final project marks are evaluated independently from the weekly submissions and getting 100% on the weekly submissions does not impact the final project marks

Project 1: Database Design Requirements

- Each student is assigned a product by the instructor
- A database is to be designed and implemented based on the Customer Sales Invoice and other information provided
- Final database design project includes:
 - Database design technical word document
 - Converted PDF file

Week 3: Company Overview and Product

1. Create a Word document (.docx) called DB1_99999, where 99999 is your student number. This document is a database proposal developed for the IT Director and contains the components for the database design.
2. Format the document as follows:
 - a. Front cover page including your name, student number, project name, term, and professor name
 - b. Styles for titles, headings, and so on
 - c. A header that contains the project title
 - d. A footer that contains your name, student number, and page number
 - e. Table of contents generated from the heading styles
3. Include headings and content identified in these instructions for each milestone. For example, the following is included for week 3:
 - Company Overview and Product (heading 1)
 - Company Name (heading 2)
 - Company Overview/Description (heading 2)
 - Product (Heading 2)
 - Product Attributes (Heading 2)And so on for each week

Company Name

- Must be a unique name that is not an existing company name

Company Overview/Description

- Include an overview/description of the company that is unique and not from an existing website

Product

- Identity your product

Product Attributes

- List at least 8-10 attributes (characteristics), such as model, brand, type, serial number, description, size, color, height, width, price, and so on. Example attributes can be found on many websites.
- Identify the primary unique identifier (UID). This UID can be called product id, product code, or something similar. You decide.

Week 4: Customer Sales Invoice

Customer Sales Invoice

- Modify the sample Customer Sales Invoice to reflect your company and the company's product. The invoice must be based on the assigned product only.
- In addition to product id, quantity, unit price, and extended price, include three attributes that identify the product purchased. Use the three attributes that best describes the product purchased. It is not necessary to include all product attributes on the invoice.
- Include at least five products purchased on the invoice.

Week 5: Entity Relationship Diagrams

ER Diagram including M:M Relationships

- Create an ER diagram, including many-to-many relationships, based on the customer sales invoice and the product attribute list. In addition, include the following requirements:
 - The company operates warehouses in different regions of the country
 - Each warehouse stores products and each product may be stored at each warehouse. Not all products are stored at each warehouse. In addition, a warehouse could store a product, but be out of stock of that product. The company wants to maintain the inventory of each product at each warehouse.
- The entity-relationship diagram includes:
 - Entities
 - Attributes (mandatory or optional)
 - Unique Identifiers (primary key)
 - Unique keys
 - Relationship cardinality (and optionality)
 - Identifying and non-identifying relationships
 - Foreign Keys
 - Named Relationships
 - ERD language

ER Diagram including M:M Relationships Resolved

- Create a second ER diagram with many-to-many relationships resolved

Relational Model

- Create a relational model in table format of the proposed database:
 - Tables
 - Key Types (PK, FK, UK)
 - Column Optionality (mandatory or optional)

- Column Names
- Data Types
- Length/Decimal Positions

Additional Business Rules

- List additional business rules including:
 - Table columns that are auto-generated (CUSTOMERS and ORDERS tables)
 - Columns that have a default value
 - Specific restrictions to column values. Include at least one each of the following:
 - Upper or lower limit (e.g. salary not greater than 175,000)
 - Range of values (e.g. salary between 60,000 and 150,000)
 - A list of valid values (e.g. job code must be 'T,' 'J,' 'M')
 - Boolean value (e.g. active status is 'T' or 'F', 1 or 0, 'Y' or 'N')

Week 7: Project 1 Database Design Due

- Database design project due Friday week 7 at 8:00 p.m.:
 - Database design technical word document (.docx file)
 - Converted PDF file (.pdf file)

Project 2: Database Build Requirements

In this project, you create the SQL database script based on the database design from project 1.

Week 6: Database Build – SQL File

- Create a file called `SQL1_99999.SQL`, where 99999 is your student number. This file will contain the SQL script that builds the database
- All non-code must be commented so the entire code will run as a script
- At the top of the file, include your name, student number, project name, term, and professor name
- Include headings and comments throughout the document that clearly identify the SQL code and each step in the process
- Upon completion, the SQL script file will contain (1) `CREATE TABLE` statements; (2), `INSERT` statements, database constraint statements; and (3) database constraint testing

Database Structure

1. Construct `CREATE` statements for the database developed in Project 1
2. Auto-generated keys
 - `CUSTOMERS` table – auto generated `IDENTITY` primary key
 - `ORDERS` table – auto-generated `SEQUENCE` primary key

Populate Database

3. Create **INSERT** statements that populate each table:
 - Minimum of 15 customers
 - Minimum of 10 associates
 - Minimum of 30 products/parts
 - Minimum of 30 orders
 - Minimum of 10 orders with one product on each order
 - Minimum of 10 orders with two products on each order
 - Minimum of 10 orders with three or more products on each order
 - Minimum of 5 Warehouses
 - Minimum of 60 rows for product inventory
 - Minimum of 20 rows in the other tables
 - Minimum of 10 rows in the intersection tables

Week 7: Database Constraint and Testing

DDL Database Constraint Statements

- Use **ALTER TABLE** statements to create the constraints identified in the relational model and additional business rules
- The **NOT NULL** constraints may be defined at the column level
- Include the following constraint testing:
 - a. Data types (one test for each different data type)
 - b. **NOT NULL** (1)
 - c. **DEFAULT** (1)
 - d. Primary keys (2)
 - e. Unique keys (1)
 - f. Foreign keys (4)
 - i. Foreign key **ON UPDATE** constraint (1)
 - ii. Foreign key **ON DELETE** constraint (1)
 - g. **CHECK** constraints (3 required)
 - h. **CHECK** constraint that simulates a **BOOLEAN** data type (1)
- Test each constraint and provide unit-testing documentation:
 - i. Provide a test that validates valid data is accepted
 - j. Provide a test that validates invalid data is refused. For constraints such as ranges, test the upper and lower ranges. In other words, verify that all conditions work correctly.

Week 8: Study Week

These projects are based on a 15-week term with a study week in week 8. You instructor may adjust these weekly submissions based on the number of weeks in your term.

Week 9: Database Build Project Due

- Submit final database build project SQL script file (.sql file)

Project 3: MVC Applications

- Create four MVC applications using the template provided
- There are no submissions for these applications. Instead, the applications run directly on the website based on the instructions provided by the instructor
- Use the genGrid.html and grid.css files to create these applications
- Put a link under the Reports drop-down menu for each application

Week 15: MVC Applications Due

The four applications are due Monday of week 15 at 9:00 a.m. Complete the four web applications below.

WEB1 Application

- WEB1 application is a SQL Query on the Products table
- Create a view called WEB1V over the PRODUCTS table that include 4 – 8 columns
- Create the WEB1.HTML and WEB1.CSS files using the genGrid.html and gridStyles.css templates
- Add a link to the REPORTS drop-down menu for the WEB1 application

WEB2 Application

- WEB2 application is a query on customer orders
- Create a view called WEB2V
- Include customer name, order date, order number, product id, description, quantity ordered, price, extended price (quantity ordered * price). Your column names may vary
- Create the WEB2.HTML and WEB2.CSS files
- Sort product id within order number within order date within customer name

WEB3 Application

- WEB3 application is a query on customer order totals
- Create a view called WEB3V
- Include customer name, order id, and order total
- Sort order total DESC within customer name
- Create the WEB3.HTML and WEB3.CSS files

WEB4 Application

- WEB4 application is a query on total inventory by product
- Create a view called WEB4V

- Include product id, product name, product inventory value (quantity on hand * cost)
- Sort product inventory value DESC within product name
- Create the WEB4.HTML and WEB4.CSS files

Marking Scheme

Figure B-1 provides a sample marking scheme for these projects. The instructor may provide a different marking scheme.

	Project 1												Project 2					Project 3							
	Tech Doc						Database Design						Database Build					MVC Apps							
Final Marks	Front Cover	Styles	Header & Footer	Generated Table of Contents	Company Overview	Company Product/Attributes	Customer Sales Invoice	ERD (M:M Relationships)	ERD (M:M Relationships Resolved)	ER Language	Relational Model	FK Relationship Mapping	Additional Business Rules	Final Mark	Build - CREATE Statements	Build - Constraints - ALT	Build - INSERT Statements	Sequence	Identity	Constraint Testing	Final Mark	Web 1	Web 2	Web 3	Web 4
20	1	1	1	1	1	1	2	2	2	2	2	2	2	12	2	2	2	2	2	2	8	2	2	2	2

Project 1

- Technical Document: One mark for each task. Either one (100%) or zero
- Database Design components are marked out of two as follows:
 - Two marks if component completed 100% according to the specifications
 - One mark if component is included but one item is incomplete or incorrect
 - Zero if component not included or more than one item is incomplete or incorrect

Project 2

- Database build components are marked out of two as follows:
 - Two marks if component completed 100% according to the specifications
 - One mark if component is included but one item is incomplete or incorrect
 - Zero if component not included or more than one item is incomplete or incorrect