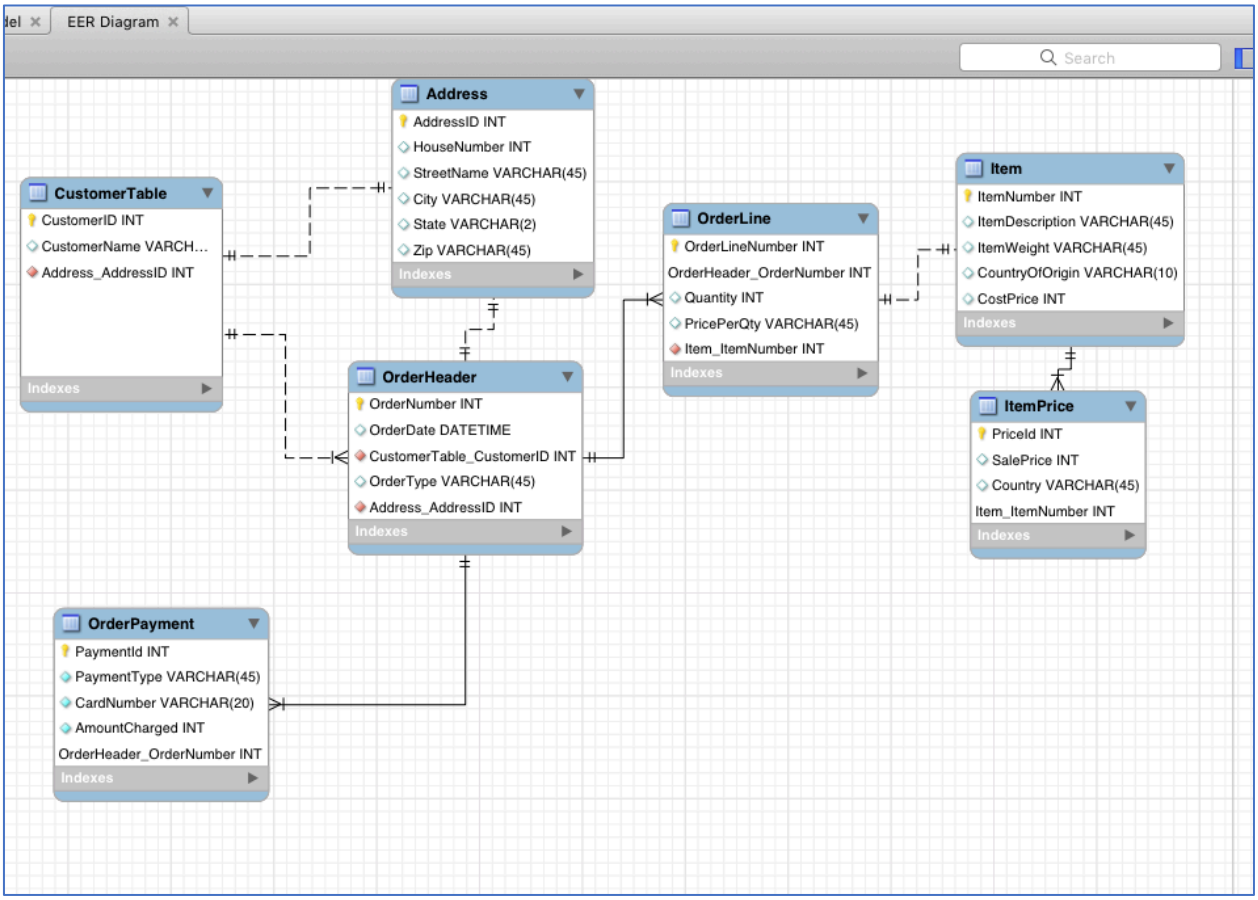
Unit 4: Homework

Setup - Create Ecommerce Order Database from ER Model

1. Use MySqlWorkbench to load ER model from “ERModel_EcommerceOrderDB.mwb”



2. Explore the tables modeled in the entity and show how you explored the tables

3. Explore the referential integrity constraints modelled in the ER model and show how you explored the constraints

4. Generate DDL from ER Model from File – Export – Forward Engineer
    a. Ensure “Omit Schema Qualifier in Object Name” is selected

Turn in the DDL generated from the ER model and screen shots and explanations showing your exploration in steps 1, 2, and 3.

Create Schema in MySQL

1. Log into mysql as admin and create a database schema for the Ecommerce database.

   create schema EcommerceDB;

2. Run the DDL script generated from the ER model from

   File – Run SQL Script
   (Ensure the EcommerceDB schema is selected , char set is UTF8)

   Alternately , run the script from mysql command line using

   mysql> use EcommerceDB;
   mysql> source <<DDL Script>> ;

   Verify the tables created using
   mysql > show tables;

   Turn in a screenshot showing the loaded tables.

Load Data

1. Load sample data in the database by running the script:

   mysql> use EcommerceDB ;
   mysql> source LoadEcommerceDBData.sql ;

Turn in a screenshot showing the loaded data for at least three tables including the commands used to show the data (if any).

Exercises

Create SQL queries to address the questions below:

1. Write a SQL query to list the total number of orders in the database
2. Write a SQL query to list the total number of customers in the database
3. What is the total number of customers that have placed an order in the system?
4. Write a SQL query to list all the unique Zipcodes across all addresses
5. Write a SQL query to list the 10 most frequently ordered items

Design Question

Building on the ecommerce database design we are working on, the retailer wants to add support for “Gift orders”. The customer will now be able to order gifts for family and friends on a single order and have them shipped to different addresses.

- How would you change your database design to accommodate this requirement?
- Would you create new tables or change existing tables?
- Your ecommercedb now has close to 10k orders already. How will the requirement

  impact the existing orders in the tables? Are there special design considerations for handling the existing orders?

  Turn in:

1. an updated ER model for the Ecommerce DB showing the design changes
2. a Word document addressing the questions above.

Feel free to collaborate with your friends. Mention the names of the people you collaborated with and ensure each of you turn in your submissions individually.