

GiftHorse Project Instructions

Objective:

Test the database design for GiftHorse, LLC. To do that you will create and populate a MySQL database according to the design in the Relational Model provided for you. Then you'll be writing queries to prove that the database design will meet the user requirements.

A Test Plan has been provided to outline the user requirements you'll be testing and help to describe the test data that will be needed for testing the queries you will write.

Be sure to use the supporting documents in the Database Project Description item in E360 for this project – and read all instructions carefully!

Instructions:

- 1) **Review the Test Plan**, and review the **Supporting Documents** in the Database Project Description item in E360.
- 2) **Create the database called GiftHorse**, where the xxxxx is your username. Feel free to use either MySQL Workbench or SQL scripts to create the database and tables. (Note: using a script to create the database and tables is a more secure method. If your database eventually becomes lost or corrupted, you can just rerun the script!)

Tables: Use the relational data model provided in E360 to determine which tables should be in the database.

Attributes: Use the relational model to find the primary key for each table. *Just remember, the intersection tables on the relational model will include more than one primary key.*

Use the ERD to determine most of the non-key columns that should be included in each table. But - be sure to check the relational data model to include any *additional non-key columns that were added as foreign keys* in the modeling process.

As for data types and sizes, use your best judgment! Looking at

sample data will help you. And, if you find you've chosen the wrong data type or size – you can always change it when you start entering data. Hint: Dates, regardless of format/style written in our data should be stored as a DATE or DATETIME file. Currency fields should allow 2 decimal places even if your data shows integer numbers.

Intersection Tables: In addition, you will need to add the following non-key attributes to the intersection tables that were added during the data modeling process:

<u>OrderLine table</u>	<u>BasketContents table</u>
ProdQty (a numeric field)	ItemQty (a numeric field)
UnitPrice (a decimal field)	

- 3) **Add test data to the tables.** *Before you begin, refer to the "Test Data" section of the Test Plan.* You may have to make up some test data yourself, but you'll find a lot of the test data values provided for you in the supporting documentation in E360.

Use Sample Data for test data when you can. For example, you'll find:

- test data for the Orders and OrderLine and Customer tables in the sample invoices
- a sample of the Product Catalog that will provide data for the Product table
- a sample list of vendors that will help with the Vendor table
- a sample list of items to help with the item table
- samples of the intersection tables below that you should find very helpful!

Data in Intersection Tables: Typically in intersection tables we find data that describes "the relationship" between two tables. OrderLine is a table that tracks the "line items" on each order.

For example, ProdQty is the number of a given product that was ordered on a given order. If you look at sample Order/Invoice number 100426, you can see that Alex Gee ordered 3 of the Good Four You baskets. The number "3" would go in the ProdQty attribute in the **OrderLine** table. Some of the rows in the table will look like

this:

OrdID	ProdID	ProdQty	UnitPrice
100426	s04	3	40
100422	g01	1	75.25
100422	w02	1	45.99
100400	s04	1	40
100400	w01	1	45.99

You'll have to add rows to OrderLine to track the items on the rest of the orders.

In the BasketContents table, the ItemQty attribute tells us how many of each item would be found in a specific product.

For example, in the description for product g03 it specifically says that there are:

- 2 card decks
- 2 poker chip sets
- 2 cribbage boards
- one medium wicker basket
- a fabric lining

So, the rows in the **BasketContents** table for that product will look like this:

ProdID	ItemID	ItemQty
g03	2300	2
g03	2400	2
g03	2500	2
g03	0002	1
g03	1000	1

You'll have to fill in the rest of the items and quantities for the other products used in the orders in your test database.

- 4) **Test your database relationships.** Before you begin on the queries from the test plan, do a "pre-test" and try some queries to make sure that your tables are joining properly. For example, you could try to do quick queries to join the following tables:

- Customer and Orders
- Orders, OrderItem, and Product
- Product, BasketContents, and Item
- Item Vendor

If your queries return the correct answers, then you can successfully join your tables – you’re ready to work on the queries from the test plan!

If your queries didn’t work, you need to check:

- Do your foreign keys have the same data types and sizes as the primary keys they’re supposed to match?
- Do you have a data value in the primary key in one table that matches the data value in the foreign key in the matching table?
 - Watch for spelling errors in attribute names AND in data values
 - Watch for upper-case versus lower-case differences in data values

- 5) **Develop and run your test queries.** Write ***and save*** your four test queries in a script. The specifications for the queries are found in the Test Plan. Test your queries until you believe the queries are working.

Keep in mind that you are TESTING the database design. Sometimes the test proves that the design is *correct*. Sometimes, the query is written correctly, but you can’t get the correct answer. In that case, the test proves that something is *incorrect or missing* from the database design. Only by writing and running queries can you find database design errors!

- 6) **Develop and post your conclusion.** Write just one or two sentences in the discussion post for your project to note ONE the following choices:

- Did you find the database design to be adequate? If so, all of the queries must have worked and returned correct answers.
- Did you find any errors in the design? If so, what was the error, and which query or part of a query proved the error?

Note that you will not be able to view comments posted by other students.

7) **Submit evidence of your work via E360.** Submit the following items:

- Your database file exported from MySQL. Be sure to export ALL of your tables.
- Your script containing the four test queries. Be sure that you've run and tested your script! I'll run it myself to see your query results! **If a query doesn't run for me due to errors from incorrect/missing SQL statements the query will receive 0 points. Budget time to develop/test/debug your queries.**

Save the script and the exported database file together in a folder – and zip the ENTIRE folder. That's the zip file you'll submit via E360.