Project Step 3 Draft - JL Liquor GROUP 47

URL: https://web.engr.oregonstate.edu/~malonlia/index.html

Step 3 Reviews

Jiehui Huang

Does the UI utilize a SELECT for every table in the schema? In other words, data from each table in the schema should be displayed on the UI. Note: it is generally not acceptable for just a single query to join all tables and displays them.

Yes, the UI utilize a SELECT for every table in the schema.

Does at least one SELECT utilize a search/filter with a dynamically populated list of properties?

It shows the tables, and it didn't utilize a search/filter with a dynamically populated list of properties.

Does the UI implement an INSERT for every table in the schema? In other words, there should be UI input fields that correspond to each table and attribute in that table.

Yes, the UI implement an INSERT for every table in the schema.

Does each INSERT also add the corresponding FK attributes, including at least one M:M relationship? In other words if there is a M:M relationship between Orders and Products, INSERTing a new Order (e.g. orderID, customerID, date, total), should also INSERT row(s) in the intersection table, e.g. OrderDetails (orderID, productID, qty, price and line_total).

Yes, there is at least one M:M relationship.

Is there at least one DELETE and does at least one DELETE remove things from a M:M relationship? In other words, if an order is deleted from the Orders table, it should also delete the corresponding rows from the OrderDetails table, BUT it should not delete any Products or Customers.

Yes, there is at least one DELETE, and it remove the things from a M:M relationship.

Is there at least one UPDATE for any one entity? In other words, in the case of Products, can productName, listPrice, qtyOnHand, e.g. be updated for a single ProductID record?

Yes, there is at least one UPDATE on the LiquorsOrders.

Is at least one relationship NULLable? In other words, there should be at least one optional relationship, e.g. having an Employee might be optional for any Order. Thus it should be feasible to edit an Order and change the value of Employee to be empty.

Yes, there is at least one NULLABLE such as Customers.

Do you have any other suggestions for the team to help with their HTML UI? For example using AS aliases to replace obscure column names such as fname with First Name.

It seems there are CRUD on every tables, and we need one update and delete should be good.

Mikaila Martin

Does the UI utilize a SELECT for every table in the schema? In other words, data from each table in the schema should be displayed on the UI. Note: it is generally not acceptable for just a single query to join all tables and displays them.

Yes, each page displays a table from the schema, and all seven entities are there.

- *Does at least one SELECT utilize a search/filter with a dynamically populated list of properties?*

It looks like all of the selects are just to show the tables, but the CRUD forms do prepopulate the corresponding data.

- *Does the UI implement an INSERT for every table in the schema?* In other words, there should be UI input fields that correspond to each table and attribute in that table.

Yes, every page provides a form to add a row to the table.

- *Does each INSERT also add the corresponding FK attributes, including at least one M:M relationship?* In other words if there is a M:M relationship between Orders and Products, INSERTing a new Order (e.g. orderID, customerID, date, total), should also INSERT row(s) in the intersection table, e.g. OrderDetails (orderID, productID, qty, price and line_total).

Yes, the DDL includes cascades that should update the intersection table.

- *Is there at least one DELETE and does at least one DELETE remove things from a M:M relationship?* In other words, if an order is deleted from the Orders table, it should also delete the corresponding rows from the OrderDetails table, BUT it should not delete any Products or Customers.

There is a delete on the orders page that removes things from a M:M relationship, and it appears to be correctly implemented in the queries.

- *Is there at least one UPDATE for any one entity?* In other words, in the case of Products, can productName, listPrice, qtyOnHand, e.g. be updated for a single ProductID record?

Yes, there are update forms for every entity!

- *Is at least one relationship NULLable?* In other words, there should be at least one optional relationship, e.g. having an Employee might be optional for any Order. Thus it should be feasible to edit an Order and change the value of Employee to be empty.

Yes, the relationship between customers and orders is optional and nullable.

- *Do you have any other suggestions for the team to help with their HTML UI? For example using AS aliases to replace obscure column names such as fname with First Name.*

Just a reminder that you don't have to do all CRUD operations for every table! You only have to implement one update and one delete.

Virgilio Viernes

Does the UI utilize a SELECT for every table in the schema? **In other words, data from each table in the schema should be displayed on the UI. Note: it is generally not acceptable for just a single query to join all tables and displays them.**

Yes, all tables from the database are properly displayed on the UI.

Does at least one SELECT utilize a search/filter with a dynamically populated list of properties?

It does not appear to be the case that the project has a SELECT statement that utilizes a search or filter feature. As this is a draft, I imagine that the group has specific SELECT queries that they have in mind, but here are suggestions for searching/filter a particular table: All liquors of a specific product size, All customers of a specific rewards tier, All liquors at a specific store location.

Does the UI implement an INSERT for every table in the schema? **In other words, there should be UI input fields that correspond to each table and attribute in that table.**

Yes, the UI implements an INSERT for each table.

Does each INSERT also add the corresponding FK attributes, including at least one M:M relationship? **In other words, if there is an M:M relationship between Orders and Products, INSERTing a new Order (e.g. orderID, customerID, date, total), should also INSERT row(s) in the intersection table, e.g. OrderDetails (orderID, productID, qty, price, and line_total).**

Yes, each INSERT adds the appropriate FK attributes. In addition, the M:M relationships present in the database containing appropriate CASCADE statements so that changes in individual tables occur it their corresponding intersection tables.

Is there at least one DELETE and does at least one DELETE remove things from an M:M relationship? **In other words, if an order is deleted from the Orders table, it should also delete the corresponding rows from the OrderDetails table, BUT it should not delete any Products or Customers.**

Yes there are at least one DELETE. Additionally, the addition of ON DELETE CASCADE statements are appropriately implemented.

Is there at least one UPDATE for any one entity? **In other words, in the case of Products, can productName, listPrice, qtyOnHand, e.g. be updated for a single ProductID record?**

Yes, there are UPDATE queries for each entity (Orders, Stores, Customers, Employees, Liquors, RewardTiers, LiquorOrders).

Is at least one relationship NULLable? **In other words, there should be at least one optional relationship, e.g. having an Employee might be optional for any Order. Thus it should be feasible to edit an Order and change the value of Employee to be empty.**

Yes it appears that there is a NULLable relationship between Customer and Order. The customerID attribute of orders can be null.

Do you have any other suggestions for the team to help with their HTML UI?

Not at this time. As this point, the draft of the project looks to be a solid foundation to work from. Great job, Group 47!

Edgar Palaquibay

Does the UI utilize a SELECT for every table in the schema? **In other words, data from each table in the schema should be displayed on the UI. Note: it is generally not acceptable for just a single query to join all tables and displays them.**

Yes, all tables are displayed on the UI with data in each of them.

Does at least one SELECT utilize a search/filter with a dynamically populated list of properties?

No dropdown or dynamic search/filter. It listed all it's data with tables.

Does the UI implement an INSERT for every table in the schema? **In other words, there should be UI input fields that correspond to each table and attribute in that table.**

Yes.

Does each INSERT also add the corresponding FK attributes, including at least one M:M relationship? **In other words, if there is an M:M relationship between Orders and Products, INSERTing a new Order (e.g. orderID, customerID, date, total), should also INSERT row(s) in the intersection table, e.g. OrderDetails (orderID, productID, qty, price, and line_total).**

Yes! Cascade specifications are set correctly.

Is there at least one DELETE and does at least one DELETE remove things from an M:M relationship? **In other words, if an order is deleted from the Orders table, it should also delete the corresponding rows from the OrderDetails table, BUT it should not delete any Products or Customers.**

Yes there is at least one delete and is cascaded properly.

Is there at least one UPDATE for any one entity? **In other words, in the case of Products, can productName, listPrice, qtyOnHand, e.g. be updated for a single ProductID record?**

Yes! I saw at least one update query for Movies, Genres, Actors, and Directors.

Is at least one relationship NULLable? **In other words, there should be at least one optional relationship, e.g. having an Employee might be optional for any Order. Thus it should be feasible to edit an Order and change the value of Employee to be empty.**

productPrice attribute for Liquors is NULLAble, but it's not part of a relationship so no.

Do you have any other suggestions for the team to help with their HTML UI?

Nothing else I haven't mentioned.

Actions based on the feedback

- Made a NULLable relationship between Customers and RewardTiers. So a customer does not have to have a rewards tier ID tied to their entry.
- Removed extra UPDATE and DELETE statements to simplify down to what the actual Project Requires

Upgrades to the Draft version

- Removed composite key in document version of liquorsorders sample data
- Changed the DML for the Orders and Customers tables to use JOINs to make human readable columns instead of just Foreign Keys, to comply with the rubric

ALL THE ITEMS FROM STEP 1-2

Step 2 Reviews

Katie Schaumleffle

- **Does the schema present a physical model that follows the database outline and the ER logical diagram exactly?**
    - **The schema is missing the attribute for “AddressCity” in the “Stores” entity. It’s shown in the ERD diagram but is just missing in the schema. Otherwise, great job!**

- Is there consistency in a) naming between overview, outline, ER and schema entity/attributes b) entities plural, attributes singular c) use of capitalization for naming?
    - For the most part, naming looks consistent. The only thing I notice is that when you created the Schema, the FK automatically put the entity name and an underscore in from of the attribute name (ie “Customers_customerID”). It did this to our group as well and we had to go in and change them. I don’t think we’re using this convention in this class, but I could be wrong (and our project might be wrong here too, so I’m not quite sure what the “proper” method is.) However it’s not consistent with your ERD diagram so I would just go back and check to make them match.
- Is the schema easy to read (e.g. diagram is clear and readable with relationship lines not crossed)?
    - Yes, the schema is very easy to read, however I think it looks more like the ERD diagram than the Schema represented in the modules. My understanding is that there should be lines showing how attributes are represented between the entities (where the FK are coming from).
- Are intersection tables properly formed (e.g. two FKs and facilitate a M:N relationship)?
    - Yes, the intersection tables look like they are properly formed here.
- Does the sample data suggest any non-normalized issues, e.g. partial dependencies or transitive dependencies?
    - I don’t see any partial dependencies outside of what’s already been mentioned (and already given the clear from the information we’ve gotten from the module). The sample data and design look great!
- Is the SQL file syntactically correct? This can be easily verified by using PhPMyAdmin and your CS 340 database (do not forget to take backup of your own database before you do this!)
    - Yes, the SQL file looks good and appears to be syntactically correct. I was able to download and run it without any issues.
- In the SQL, are the data types appropriate considering the description of the attribute in the database outline?
    - Good job! The data types look great. The main discrepancy I see is that your outline shows more attributes for the Customers and Orders tables allowing to be NULL, but they are listed as NOT NULL in your SQL code. Otherwise I think it looks good!
- In the SQL, are the primary and foreign keys correctly defined when compared to the Schema? Are appropriate CASCADE operations declared?
    - Yes, I see both PK and FK accurately defined. It does look like the appropriate CASCADE operation are declared.
- In the SQL, are relationship tables present when compared to the ERD/Schema?
    - Yes, the tables look great. I would just suggest making you’re your FK’s are consistent with the outline and ERD.
- In the SQL, is all example data shown in the PDF INSERTED?
    - They are mostly the same, but there is some discrepancies in the Employees table phone numbers and in the Customers table TotalSales amount. Otherwise, it looks good!

Reply

Joseph Doiron 3d

- Does the schema present a physical model that follows the database outline and the ER logical diagram exactly?
    - Not quite, the stores entity is missing the addressCity attribute in the schema.
- Is there consistency in a) naming between overview, outline, ER and schema entity/attributes b) entities plural, attributes singular c) use of capitalization for naming?
    - It looks consistent to me.I suggest getting rid of the Entity_EntityName formatting for foreign keys. it is redundant and makes the column names longer than they need to be. it'll end up making query writers write more. Less typing is preferred.
- Is the schema easy to read (e.g. diagram is clear and readable with relationship lines not crossed)?
    - Yes - I see nothing wrong with it.
- Are intersection tables properly formed (e.g. two FKs and facilitate a M:N relationship)?
    - Yes - I honestly can't think of anything to add here. It's all correct.
- Does the sample data suggest any non-normalized issues, e.g. partial dependencies or transitive dependencies?
    - custTotalSales is an aggregate column that can be derived through aggregate sql queries. Databases are write heavy (more data is written then read) and should be optimized as such. Having this column means that you'll need to update the Customer table every time a sale happens. It also increases the chances data becomes out of sync.
- Is the SQL file syntactically correct? This can be easily verified by using PhPMyAdmin and your CS 340 database (do not forget to take backup of your own database before you do this!)
    - It works. No comment.
- In the SQL, are the data types appropriate considering the description of the attribute in the database outline?
    - yes.
- In the SQL, are the primary and foreign keys correctly defined when compared to the Schema? Are appropriate CASCADE operations declared?
    - Yes, I see proper use of cascading in the sql.
- In the SQL, are relationship tables present when compared to the ERD/Schema?
    - Yes, nothing to note here.
- In the SQL, is all example data shown in the PDF INSERTED?
    - Yes.

Reply

Joseph Heinz 3d

- Does the schema present a physical model that follows the database outline and the ER logical diagram exactly?

Yes, the schema image in the report is consistent with the ERD. Furthermore, importing your .SQL file and rearranging the schema diagram generated there (I am looking at phpMyAdmin, but I'm sure mySQL would be the same) yields the same result, so your tables and relationships modeling is looking great to me.

- **Is there consistency in a) naming between overview, outline, ER and schema entity/attributes b) entities plural, attributes singular c) use of capitalization for naming?**

Naming looks consistent to me, outside of what you are doing with prepending the Table name to the front of the field name for Foreign Key fields in the schema diagram and in your actual .SQL file. I have not seen this convention you use for foreign key field names (ReferencedTableName_fieldName), e.g. "RewardsTiers_rewardsTierId" in the "Customers" table, used or recommended in this class before. Don't get me wrong, I totally understand this convention and it's clear to me and I see the utility in doing so, just calling this out. It might be wise to check with a Professor or ULA on this to ensure it's not recommended against, and then update your ERD or .SQL file + schema diagram to all be consistent here (again, what you did is clear to me, just don't want you to get dinged later if someone is being knit-picky on consistency there).

- **Is the schema easy to read (e.g. diagram is clear and readable with relationship lines not crossed)?**

Schema looks super clear and readable to me, and the orientation is consistent with the ERD layout which makes it quite easy to follow.

- **Are intersection tables properly formed (e.g. two FKs and facilitate a M:N relationship)?**

Yep! "LiquorsOrders" table looks great here, and sample data illustrates this very logical setup.

Maybe unnecessary for the scope of this project, but is there any relationship between "Employees" and "Stores" that could also be represented as M:M via another intersection table? A given Store could have many Employees, and a given Employee might work at a few different Stores. Probably not something you want to mess with this far in, just food for thought!

- **Does the sample data suggest any non-normalized issues, e.g. partial dependencies or transitive dependencies?**

The sample data generally looks great in terms normalization. The one small dependency, like the example in the Module notes, is City + State are theoretically dependent on Zip in the "Customers" table. However, most DB designers choose to live with this partial dependency, to avoid the complexity introduced by adding a separate table for "Zips" with City and State listed as attributes there.

So given practically, I think this design is great, and I see no partial dependencies anywhere else.

- **Is the SQL file syntactically correct? This can be easily verified by using PhPMyAdmin and your CS 340 database (do not forget to take backup of your own database before you do this!)**

SQL file worked great upon importing, and created sample data just as expected and matching the sample data included in the report. Great work!

- **In the SQL, are the data types appropriate considering the description of the attribute in the database outline?**

Some of the currency-related fields, like "custTotalSales" on Customers table, might want to be reflected in a way that allows 2-digits of decimal values. Other than that, the data types look great to me!

- **In the SQL, are the primary and foreign keys correctly defined when compared to the Schema? Are appropriate CASCADE operations declared?**

The primary and foreign keys look great, and I do see "ON DELETE" and "ON UPDATE" cascade operations declared in the relevant spots, so looking good there!

- **In the SQL, are relationship tables present when compared to the ERD/Schema?**

Yes, all tables (including intersection) look great in SQL after importing. Great job!

- **In the SQL, is all example data shown in the PDF INSERTED?**

Yes, as explained above. Nailed it!

Reply

Jonathan Ross 3d

Hi Jenna and Liam! Please see my review below. I hope you find my feedback helpful!

- **Does the schema present a physical model that follows the database outline and the ER logical diagram exactly?**

Yes. The same entities are present in the outline, schema, and ER diagram. The only minor discrepancy appears to be a typo in the table 'RewardsTiers' which is documented as 'RewardTier' in the ER diagram.

- **Is there consistency in a) naming between overview, outline, ER and schema entity/attributes b) entities plural, attributes singular c) use of capitalization for naming?**

I noticed a few differences between the schema and the ER diagram in your orders table. Your ER diagram has three foreign keys (customerID, employeeID, storeID) which appear to have different names in the schema (Employees_employeeED, Stores_storeID, Customers_customerID).

I also noticed a difference between the schema and the ER diagram for the Customers table. The foreign key rewardTierID in the Customers table of the ER diagram is show as RewardsTiers_rewardsTierId in the schema.

On further review in looks like there are differences between the schema and ER diagram for all of your foreign keys. I suspect this is because of the way MySQLWorkbench auto-populates a foreign key table when the tables are linked. You can fix this by double-clicking on the attribute in the table and editing as needed.

Your outline appears consistent with your schema but has the same discrepancies as noted above when compared to the ER diagram.

- **Is the schema easy to read (e.g. diagram is clear and readable with relationship lines not crossed)?**

Your schema is easy to ready. My only suggestion would be to straighten up the lines - I suspect you have put bends in the lines to make it more compact, but I think it will be eaier to read if they are straight.

My other two suggestions here are 1) make the positioning of entities between your ER diagram and schema the same so they are easier to compare and 2) remove crossed lines from your ER diagram (which would be solved by implementing 1).

- **Are intersection tables properly formed (e.g. two FKs and facilitate a M:N relationship)?**

It looks like you have one M:N relationships, one between Liquors and Orders and it appears properly formed. I have one general suggestion here which is to make a dedicated primary key for the LiquorsOrders table which will make your querying easier down the road.

- **Does the sample data suggest any non-normalized issues, e.g. partial dependencies or transitive dependencies?**

There does not appear to be any non-normalized issues here. The only transitive dependency I see (which results in duplicate data) is in your address data (city, state, zip) but this is an issue that you have complete discretion on whether or not to address.

Just as a general comment I noticed that your Employees table has an SSN attribute which is unique but that you also have a separate primary key for this table. I suspect you could have used SSN as a primary key here but maybe that makes for some confidentiality issues when it is used as a foreign key in other tables.

- **Is the SQL file syntactically correct? This can be easily verified by using PhPMyAdmin and your CS 340 database (do not forget to take backup of your own database before you do this!)**

Yes! Database uploaded without any errors and all entities and transactions were recorded without issue.

- **In the SQL, are the data types appropriate considering the description of the attribute in the database outline?**

The data type for orderTotal in your Orders table is documented as int in your outline but is a decimal(10,2) per your sql. I think decimal(10,2) makes more sense here so I would recomend modifying your outline accordingly.

The data type for productPrice in your Liquors table is documented as int in your outline but is a decimal(10,2) per your sql. I think decimal(10,2) makes more sense here so I would recomend modifying your outline accordingly. You've also noted this as NOT NULL in your outline but in your sql this is allowed to be null.

Your outline lists several attributes from the Customers table that are allowed to be null (eg addressStreet, addressCity) but these are NOT NULL in your sql data. I noticed the same with the customerID attribute in your Orders table.

- **In the SQL, are the primary and foreign keys correctly defined when compared to the Schema? Are appropriate CASCADE operations declared?**

Primary and foreign keys look great and match the schema perfectly. However, I did not see any CASCADE operations declared in the SQL so this probably needs to be addressed.

- In the SQL, are relationship tables present when compared to the ERD/Schema?

Yes. The same relationship tables appear to be present in the SQL. My only suggestion here is to make the naming of foreign keys consistent with your outline and ER diagram. As it stands you have adopted the same naming convention as your schema which is a discrepancy I noted above.

- In the SQL, is all example data shown in the PDF INSERTED?

In the Employees table the phone numbers from your pdf are different that the phone numbers in your sql file.

In your RewardsTiers table, the rewardsTierDiscount for ID 1 and ID 2 in your pdf are different that the values in your sql file.

In your Customers table, the TotalSales amount for ID 2 and ID 3 in your pdf are different than the values in your sql file.

I would also suggest making the table attribute names in your sample data match exactly with those of your schema / outline / ER diagram / sql so there is no possible confusion by the grader. For example, the Customers table has an attribute cusTotalSales that is documented as Total Sales in your sample data table (pdf).

Great job Liam and Jenna! Good luck with your final project.

Cheers,

Jonathan

Reply

Shawn Kim 5d

- Does the schema present a physical model that follows the database outline and the ER logical diagram exactly?
    - Almost! Stores entity is missing "addressCity" attribute in the schema.
- Is there consistency in a) naming between overview, outline, ER and schema entity/attributes b) entities plural, attributes singular c) use of capitalization for naming?

    - General suggestion: in the schema, all of the foreign keys in each entity state the entity in the beginning but the outline doesn't show that. I suggest keeping it consistent (preferably the way the outline states them). As mentioned before, Stores is missing "addressCity".
- Is the schema easy to read (e.g. diagram is clear and readable with relationship lines not crossed)?
    - Yes! Lines go to the appropriate entities.
- Are intersection tables properly formed (e.g. two FKs and facilitate a M:N relationship)?
    - Yes! LiquidOrders has 2 FKs and facilitates a M:M relationship between Orders and Liquors.
- Does the sample data suggest any non-normalized issues, e.g. partial dependencies or transitive dependencies?
    - Employees, Customers, and Stores all have the same transitive dependency (addressZip relies on addressCity and addressState) but as the lecture mentioned, there's exceptions to 3NF and this warranted it.
- Is the SQL file syntactically correct? This can be easily verified by using PhPMyAdmin and your CS 340 database (do not forget to take backup of your own database before you do this!)
    - When I put the file as is (without comments) into PhPMyAdmin, it worked fine the first time. But when I tried to put in the same query again, I get an error saying "#1451 - Cannot delete or update a parent row: a foreign key constraint fails" for the line "DROP TABLE IF EXISTS Customers;". I'm assuming you need to drop the RewardsTiers table before you drop Customers due to the FK if you were to use this in the future.
- In the SQL, are the data types appropriate considering the description of the attribute in the database outline?
    - Yes
- In the SQL, are the primary and foreign keys correctly defined when compared to the Schema? Are appropriate CASCADE operations declared?
    - Yes
- In the SQL, are relationship tables present when compared to the ERD/Schema?
    - Yes
- In the SQL, is all example data shown in the PDF INSERTED?
    - Yes, although in Rewards Tiers, the Discount rounds up in PhPMyAdmin (ex: .995 is 1.0, .9875 is .99, etc.). Not sure if this will be a problem later.

Step 1 Reviews

Caleb Verstraete

Does the overview describe what problem is to be solved by a website with DB back end?

The overview is a little brief and I don't really see a problem just that the liquor store is looking to fix other than just creating a database to track customers, employees, inventory, and sales. Is the store trying to track which customers spend the most, better categorize their inventory to put their best selling products up front. The outline also includes a rewards tier program but there is no mention in the overview. The outline includes an entity for multiple stores but the overview only seems to be about one store.

Does the overview list specific facts?

The overview does list specific numbers on employees, annual income and total inventory which is great, but you could add something about the approximate number of customers you see in a year.

Are at least four entities described and does each one represent a single idea to be stored as a list?

Looks like you guys have four entities with the customers, employees, inventory, and sales. The outline looks to have a lot more detail than the overview about them though. My suggestion would be to use what is in the outline to add more info about the entities in the overview.

Does the outline of entity details describe the purpose of each, list attribute datatypes and constraints and describe relationships between entities? Does the outline clearly indicate which entities (tables) will be implemented and which team member is primarily assigned to the associated page(s)?

The outline attributes for the entities listed look great. Names are descriptive and easy to tell what is going on, data types and info appear appropriate, and entities include descriptions to know what it is. Relationships are also easy to understand and know how

they will connect to the other entities and appear correct. Not sure but I don't think the LiquorOrders connecting table is needed at this point so you'll be ahead of the game for your schema. I also don't see mention of which team member will be implementing what.

Are 1:M relationships correctly formulated? Is there at least one M:M relationship? Does the ERD present a logical view of the database?

The relationships look correctly formulated for what is presented with 3 1:M and one M:M with Orders and Liquors. The ERD also looks good with the relationships mapped out properly. My only note would be I'm not sure the rewards tier is needed as a separate entity. Looks like this could just be included under the customer table as a tiered discount code. The ERD also looks more like a full schema to me with the connecting table between Orders and Liquors. This just took me an extra second to understand that is where your M:M relationship is located and as an ERD diagram I would just remove it for now.

Is there consistency in a) naming between overview and entity/attributes b) entities plural, attributes singular c) use of capitalization for naming?

Your naming consistency looks great between the outline and the ERD. Entities are capital and plural, attributes are singular and also use camel case and also make clear what it is going to be used for. Between the outline and the overview is where I see an issue. Sales information turns into Orders in the outline and inventory turns into Liquors. For ease of reading I would change these in the overview to match the outline.

Derrick Priebe

Does the overview describe what problem is to be solved by a website with DB back end?

Great job on this database design by the way. The design appears well thought through and well-executed overall. To get to the question answer - Yes. The team notes that their database will help track notable aspects of a liquor store including customer information, employee information, sales information, and inventory information. I think the description of the problem could potentially be elaborated. For example, one could say the stores need a way to manage orders and the associated monetary value to facilitate sales statistics and returns, manage liquor inventory across stores to enable optimal product stock across stores, manage specific customer information, customer sales and loyalty status for analytics and marketing purposes, as well as manage the employees that operate the stores to ensure proper staffing levels.

Does the overview list specific facts?

Specific facts listed in the descriptive overview include 12 employees, 10,000 liquor bottle inventory, and annual sales of $4M.

Are at least four entities described and does each one represent a single idea to be stored as a list?

There are at least 4 entities which include: Employees, Customers, Stores, RewardsTiers and Liquors.

Does the outline of entity details describe the purpose of each, list attribute datatypes and constraints, and describe relationships between entities? Does the outline clearly indicate which entities (tables) will be implemented and which team member is primarily assigned to the associated page(s)?

The outline of the entity details does describe the purpose of each. I would add that it may help to clarify “liquor products” rather than or along with “variety of liquors” and potentially use an example like Bulleit Bourbon instead of categories which may help with clarity.

It is not specifically stated which entities will be implemented but it appears that the expectation is that all will be implemented. I do not see team member page assignments.

Are 1:M relationships correctly formulated? Is there at least one M:M relationship?

There are 3 1:M relationships that include Employees/Orders, Customers/Orders, and Stores/Orders. There is one M:M relationship consisting of Liquors/Orders.

Is there consistency in a) naming between overview and entity/attributes b) entities plural, attributes singular c) use of capitalization for naming?

There is consistency between naming of overview and entity/attributes. All entities are plural and attributes (ex. Customers) are singular (ex. firstName). The use of capitalization for naming was consistent with standard first letter capitalization on tables and camelcase for attributes.

Zhiwei Ma
6 days ago

Does the overview describe what problem is to be solved by a website with DB back end?

Yes, though there isn't a problem that is explicitly mentioned, for a liquor store or any other store a DB backend would be extremely helpful to keep track of things like inventory, sales, customer information, and general employee information. Being able to keep track of sales can potentially help with increased profits by analyzing things like which customers are frequent, and which items are best sellers and adjusting business practices accordingly.

Does the overview list specific facts?

Yes. There is good background knowledge about the company in terms of employees, inventory, and overall sales. More information could be given though like how a backend database could help with operations.

Are at least four entities described and does each one represent a single idea to be stored as a list?

Yes, at least four entities are shown such as employees, customers, liquors, and orders. Each can be represented as a single idea and stored as a list.

Does the outline of entity details describe the purpose of each, list attribute datatypes and constraints and describe relationships between entities?

The entities do have details that describe the purpose of each in its attributes. Data types are also listed in each attribute as well as the relationships between entities are properly listed.

Does the outline clearly indicate which entities (tables) will be implemented and which team member is primarily assigned to the associated page(s)?

The outline does indicate what tables will be implemented but doesn’t state which team member contributed to which pages, though if each member contributed equally it shouldn’t matter too much.

Are 1:M relationships correctly formulated? Is there at least one M:M relationship? Does the ERD present a logical view of the database?

Yes, the 1:M relationships are correctly formulated and there is at least one M:M relationship - between liquors and orders.

Is there consistency in a) naming between overview and entity/attributes b) entities plural, attributes singular c) use of capitalization for naming?

Yes, every attribute is in camelCase and is consistent with each entity. Entities are also in plural and capitalized.

Katie S

4 days ago

·Does the overview describe what problem is to be solved by a website with DB back end?

Yes, the overview does a good job at explaining the purpose of their database. In this case, they are creating a database to manage specific data about a liquor store. They are planning to track the customer information, sales information, employees, and inventory. By being able to track this information, the store should be able to increase profits by analyzing the data they collect.

· Does the overview list specific facts?

The overview does list specific facts, however I feel like they could benefit with more details and specific information. For example, they could include information on how the database will help with the overall business operations.

·Are at least four entities described and does each one represent a single idea to be stored a s a list?

Yes, there were seven entities described each representing a different idea to be stored. The entities they have described are Employees, Customers, Orders, RewardsTiers, Stores, LiquorsOrders, and Liquors.

·Does the outline of entity details describe the purpose of each, list attribute data types and constraints and describe relationships between entities? Does the outline clearly indicate which entities (tables) will be implemented and which team member is primarily assigned to the associated page(s)?

The outline does include a description of the purpose of each entity with a sentence describing the purpose of each entity. It is assumed that each of the entities presented will be implemented. Datatypes, constraints and relationships are also all included. The outline does not indicate which team member is primarily responsible for each page.

·Are 1:M relationships correctly formulated? Is there at least one M:M relationship? Does the ERD present a logical view of the database?

Yes, there is one m:m relationship identified between liquors and orders. It also appears that the 1:m relationships between employees to orders, customers to orders, and stores to orders are correctly identified. The ERD does present a logical view and overall, looks like this group did an excellent job!

·Is there consistency in a) naming between overview and entity/attributes b) entities plural, attributes singular c) use of capitalization for naming?

Yes, there is consistency with the naming between the overview and ERD diagram. The entities are all plural and the attributes are all singular. Also, the use of capitalization for naming with consistent throughout. The naming on the outline matched the ERD diagram.

Actions based on the feedback

- Added the reward tier description into the outline
- Added addressCity to the stores table in the ER diagram
- Fixed the overview to specify that the company owns multiple retail locations
- Fixed the overview to better elaborate the problem being solved with a database

Upgrades to the Draft version

- **None**

Project Title: J.L. Liquors

Team Name: JL47

Project Group #: 47

Team Members: Liam Maloney, Jenna Rowan

Overview

We will be designing and implementing a website and database backend that allows a small chain of liquor stores to better understand their business by tracking everything a liquor store needs to know: customer information, employee information, sales information, and inventory information. The stores will be able to better track which liquors sell the best by location, time, etc., as well as which stores and/or employees are the most productive. Information can also be used for data analytics and marketing. The owners hope to increase sales with all of this information.
We envision these stores to have 31 employees, on-hand inventory of roughly 40,000 bottles of liquor at any given time, and annual sales of 8 million dollars. Customers will be eligible for our rewards tier program that will offer discounts based on their spend, tiers will be either bronze, silver, or gold.

Database Outline

Customers - customers of our stores

- customerID: int, auto_increment, unique, not NULL, PK
- email: varchar, not NULL
- firstName: varchar, not NULL
- lastName: varchar, not NULL
- addressStreet: varchar
- addressCity: varchar
- addressState: varchar
- addressZip: int
- custTotalSales: int, not NULL, 0 default value
- rewardTierID: int, FK
- Relationships:
    - 1:M between customers and orders is implemented with customerID FK inside of orders
    - 1:M between customers and rewardsTier with rewardsTierID FK in Customers

Employees - hired workers that operate our stores

- employeeID: int, auto_increment, unique, not NULL, PK
- socialSecurityNum: int, not NULL
- phoneNumber: int, not NULL
- firstName: varchar, not NULL
- lastName: varchar, not NULL
- addressStreet: varchar, not NULL
- addressCity: varchar, not NULL
- addressState: varchar, not NULL
- addressZip: int, not NULL
- Relationship: 1:M between employees and orders they ring up is implemented with employeeID as a FK inside of orders

Stores - physical location of the stores

- storeID: int, auto_increment, unique, not NULL, PK
- addressStreet: varchar, not NULL
- addressCity: varchar, not NULL
- addressState: varchar, not NULL
- addressZip: int, not NULL
- Relationship: 1:M between stores and orders is implemented with storeID as a FK inside of orders

Orders - what a customer purchases at store checkout.

- orderID: int, auto_increment, unique, not NULL, PK
- employeeID: int, not NULL, FK

- storeID: int, not NULL, FK
- customerID: int, FK # Optional since not everyone will be in the system
- orderTotal: int, not NULL, 0 default value
- Relationships:
    - 1:M between employees and orders with employeeID implemented as a FK here
    - 1:M between stores and orders with storeID implemented as a FK here
    - M:M between liquors and orders with orderID implemented as a FK inside of LiquorsOrders
    - 1:M between customers and orders with customerID and rewardTierID implemented as FKs here

LiquorsOrders - Intermediary table to facilitate the M:M relationship between orders and liquors.

- orderID: int, auto_increment, unique, not NULL, FK
- productID: int, auto_increment, unique, not NULL, FK
- productQuantity: int, not Null
- Relationship:
    - 1:M between orderId and LiquorsOrders with orderId FK in LiquorsOrders
    - M:1 between LiquorsOrders and productId with productId FK in LiquorsOrders

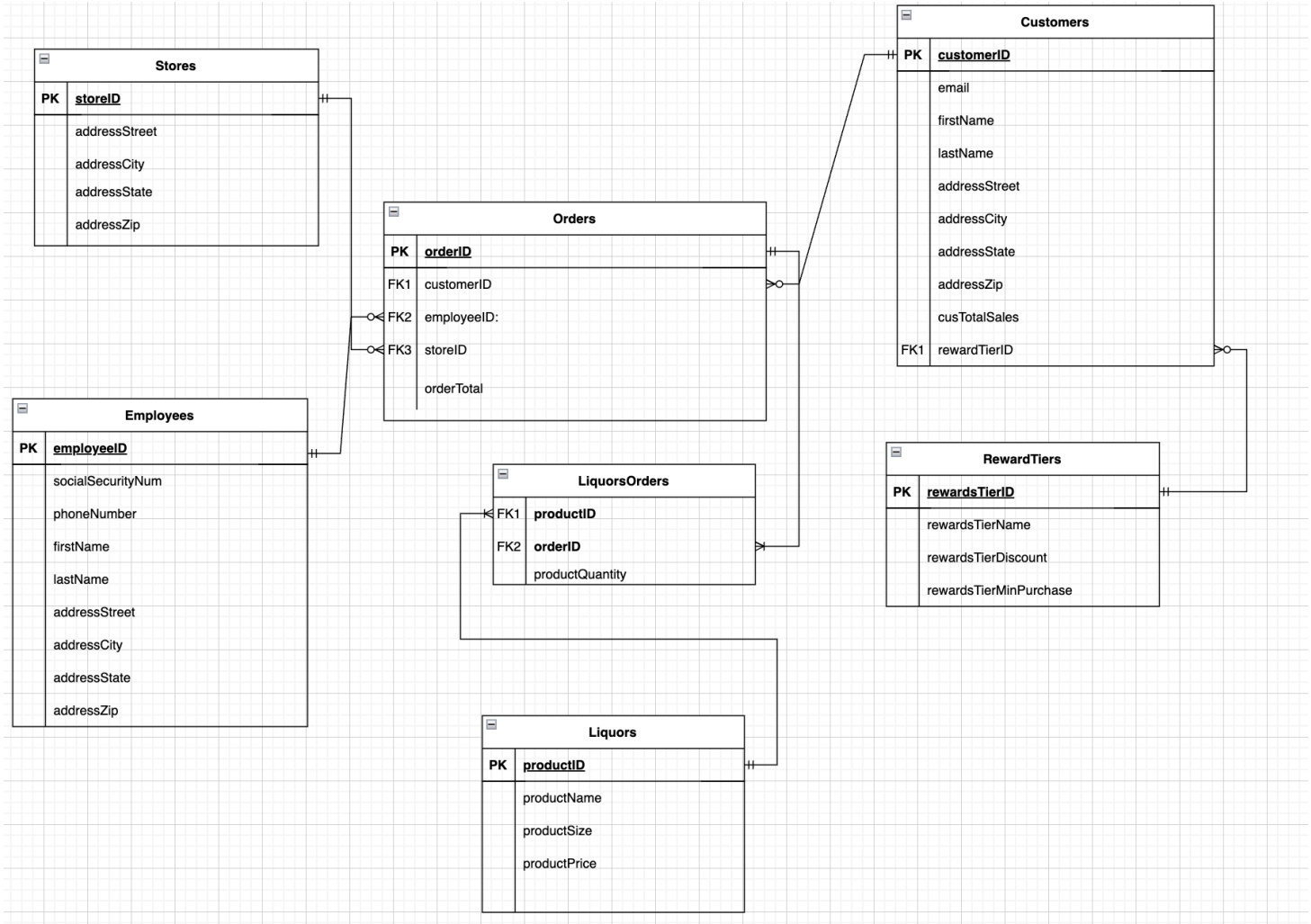
Liquors - inventory of all the varieties of liquor we carry (whiskey, vodka, etc..)

- productID: int, auto_increment, unique, not NULL, PK
- productName: varchar, not NULL
- productSize: int, not NULL
- productPrice: int, not NULL
- Relationship: M:M between liquors and orders is implemented with productID as a FK inside of LiquorsOrders

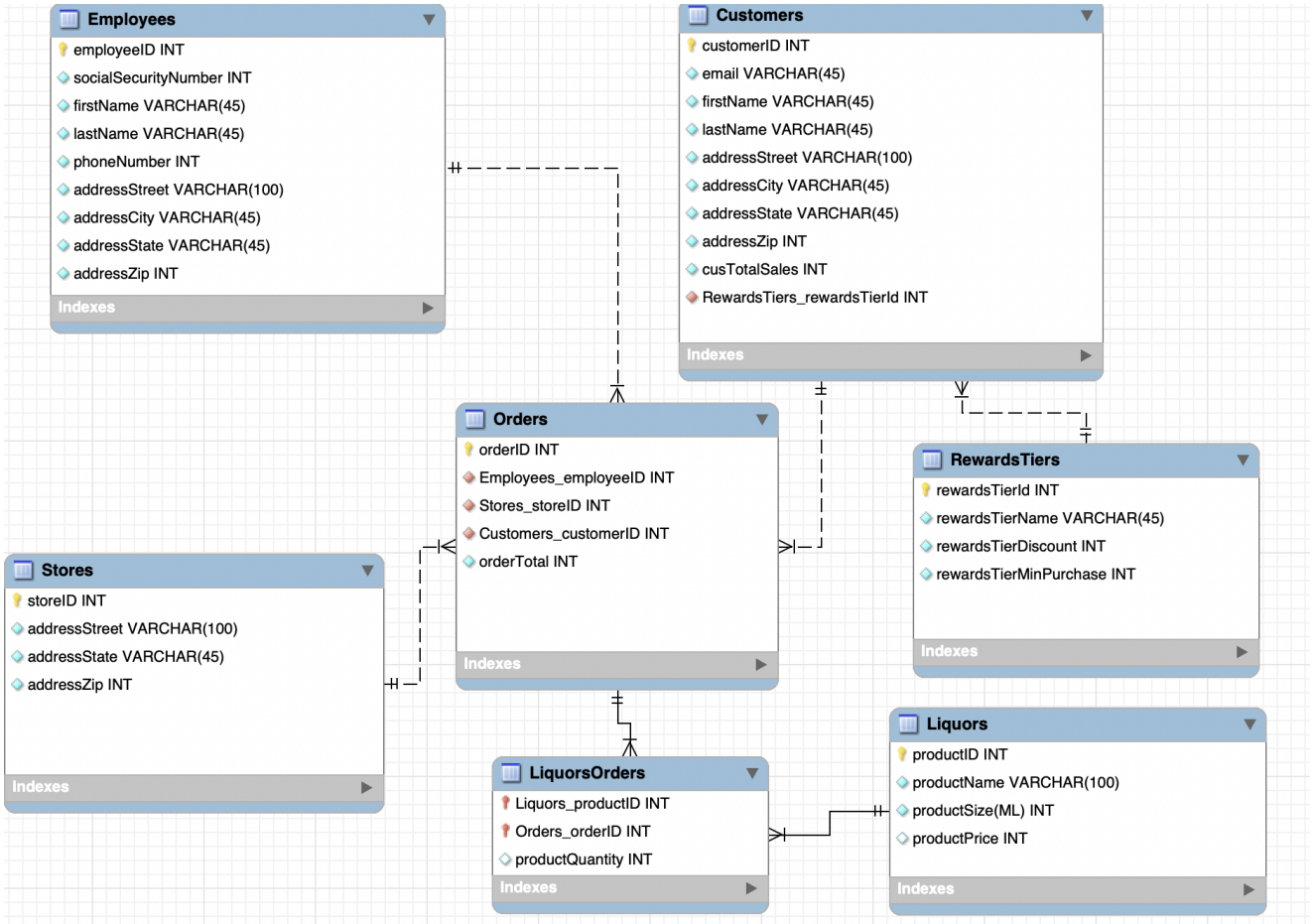
RewardsTiers - rewards program status of the customer, updated based on them passing a spend threshold (This is summary data) Silver, Gold, Platinum, Diamond

- rewardTierID: int, unique, not NULL, PK
- rewardTierName: varchar, not NULL # Bronze, Silver, Gold
- rewardTierDiscount: int, not NULL # 0.005, 0.0125, 0.02
- rewardTierMinPurchase: int, not NULL
- Relationship: 1:M between rewards tier and customers is implemented with rewardTierID as a FK inside of customers

Entity-Relationship Diagram



Schema


Employees
employeeID INT
socialSecurityNumber INT
firstName VARCHAR(45)
lastName VARCHAR(45)
phoneNumber INT
addressStreet VARCHAR(100)
addressCity VARCHAR(45)
addressState VARCHAR(45)
addressZip INT
Indexes
Customers
customerID INT
email VARCHAR(45)
firstName VARCHAR(45)
lastName VARCHAR(45)
addressStreet VARCHAR(100)
addressCity VARCHAR(45)
addressState VARCHAR(45)
addressZip INT
cusTotalSales INT
RewardsTiers_rewardsTierId INT
Indexes
Orders
orderID INT
Employees_employeeID INT
Stores_storeID INT
Customers_customerID INT
orderTotal INT
Indexes
RewardsTiers
rewardsTierId INT
rewardsTierName VARCHAR(45)
rewardsTierDiscount INT
rewardsTierMinPurchase INT
Indexes
Stores
storeID INT
addressStreet VARCHAR(100)
addressState VARCHAR(45)
addressZip INT
Indexes
LiquorsOrders
Liquors_productID INT
Orders_orderID INT
productQuantity INT
Indexes
Liquors
productID INT
productName VARCHAR(100)
productSize(ML) INT
productPrice INT
Indexes

Sample Data

Employees

ID (PK)	SSN	First Name	Last Name	Phone Number	Street	City	State	Zip
1	111223333	Hank	Schrader	5016204338	4901 Cumbre Del Sur Court	Albuquerque	NM	87106
2	222334444	Walter	White	5051930809	308 Negra Arroyo Lane	Albuquerque	NM	87111
3	333445555	Jesse	Pinkman	5051483369	9809 Margo St	Albuquerque	NM	87104

Stores

ID (PK)	Street	City	State	Zip
1	9516 Snow Heights Cir NE	Albuquerque	NM	87112
2	322 16th St SW	Albuquerque	NM	87104
3	2660 Fritts Crossing SE	Albuquerque	NM	87105

Rewards Tiers

ID (PK)	Name	Discount*	Minimum Purchase
1	Bronze	.995	100
2	Silver	.9875	500
3	Gold	.98	1000

**Discount is calculated by multiplying purchase price by total percent due (for example our Silver Tier gives a 1.25% discount, so we multiply purchase price by 98.75% to get Total Due)*

Customers

ID (PK)	Email	First Name	Last Name	Street	City	State	Zip	Total Sales	Rewards Tier
1	gaslover@aol.com	Hank	Hill	72 Desert Rose Lane	Arlen	TX	73104	3450	Gold
2	watchoutfornsa@protonmail.com	Dale	Gribble	63 Desert Rose Lane	Arlen	TX	73104	7288.37	Gold
3	lonelybill@gmail.com	Bill	Dauterive	68 Desert Rose Lane	Arlen	TX	73104	867.21	Silver

Orders

Order #	Employee	Store Location	Customer	Order Total
1	Hank Schrader	9516 Snow Heights Cir NE	Hank Hill	39.18
2	Walter White	322 16th St SW	Dale Gribble	109.72
3	Jesse Pinkman	2660 Fritts Crossing SE	Bill Dauterive	12.93

Liquors

ID (PK)	Name	Size (mL)	Price
1	100 Stories Gold Rush	750	19.99
2	1792 Small Batch	750	27.99
3	360 Vodka	1000	12.99

Liquors Orders

Liquor ID (FK)	Order ID (FK)	Quantity
1	1	2
2	2	4
3	3	1