URL
http://people.oregonstate.edu/~dunnbrit/CS340/index.html

Project Step 6
Changes during Draft process:

      I updated the data definitions and project outline to include what happens with foreign keys upon delete and update for the restaurant table, customers table, and reviews_restaurants_customers table. I decided to remove the delete account option because it was not correctly satisfying the requirement of setting a many-to-one relationship to null. Instead I added a remove option to the cuisine table which upon delete would set a customer's cuisine preference to null however it will not allow a cuisine to be deleted which is connected to a restaurant.

Feedback by the reviewers:

Actions based on the feedback:
Upgrades to the Draft version:

Project Step 5
Feedback by the reviewers:

Eddie Kovsky

READ functionalities

    1.   Are rows being listed for all entities? Yes

    2.   Are rows being listed for all relationships, as described in the Specs? Yes

    3.   Is the Search OR filter functionality present and working? Yes

DELETE functionalities

Deleting works. The app even reports if an entity has been deleted.

Andrew Huson

    1.   Are rows being listed for all entities? Yes

2. Are rows being listed for all relationships, as described in the Specs? Yes

3. Is the Search OR filter functionality present and working? Yes, you might consider adding wildcards (%) around the users input and using LIKE instead of searching for exact matches for the restaurant name.

4. Is there a better way that data could be displayed on these pages? OR Could the style of the webpage be improved? Maybe consider adding a navigation panel to each page to link between pages or at least a link back to the main page. When displaying the restaurant information along with the rating possibly a count of reviews (to show how much weight the rating should be given). It might also be nice to list all the reviews for that restaurant instead of just the summary data.

5. Anything else that you think is important for READ functionalities? Everything is working well. When displaying the users reviews it might be helpful to display the rating (like/dislike) they gave along with the restaurant name. Since the same user can review a restaurant more than once maybe a date column for the review could also be added.

DELETE functionalities: Delete functions (user/review) both work. When deleting a review for a restaurant it deletes all reviews by that user for that restaurant. It might be better to give the list of reviews the person has done with the option to delete any individual review. The remove review page displays the 'Review deleted' message even if the form is given bad data (restaurant with user id who has not reviewed that restaurant, or an invalid user id).

Russell Moon

READ functionalities

1. Are rows being listed for all entities?

All rows are listed for Cuisines and Restaurants. Locations appear along with their related

restaurants in the restaurant list, but there's not a separate table listing just locations

(not sure if this is required). Likewise, there's not a table which lists all customers.

2. Are rows being listed for all relationships, as described in the Specs?

I can infer from the Add Reviews page that there's a many-to-many relationship between

Customers and Restaurants. Again, there doesn't appear to be a table listing the

Customers-Restaurants relation.

3. Is the Search OR filter functionality present and working?

Yes. I was able to search for my User ID and I was also able to search for a Restaurant.

4. Is there a better way that data could be displayed on these pages? OR Could the style of

the webpage be improved?

Include a horizontal menu on all pages for easier navigation.

5. Anything else that you think is important for READ functionalities?

Maybe add a comprehensive list of Customers? Again, I'm not sure if this is a

requirement. Guidelines are a little vague.

DELETE functionalities

I was able to delete reviews.

Hillary Dreikorn

READ functionalities

1. Are rows being listed for all entities?
2. Are rows being listed for all relationships, as described in the Specs?
3. Is the Search OR filter functionality present and working?
4. Is there a better way that data could be displayed on these pages? OR Could the style of the webpage be improved? Could definitely benefit from a back or home button. The Add restaurant form was a bit confusing.
5. Anything else that you think is important for READ functionalities? Nope, seems like everything is displayed. Good job!

DELETE functionalities

1. Delete functionalities seemed to work, but not getting validation by seeing a table was a bit off.

Christopher Brown

READ functionalities:
1. Are rows being listed for all entities?
It looks like the drop down for setting cuisine preferences is not working correctly. It just says $cuisine_type.

2. Are rows being listed for all relationships, as described in the Specs?
Yes

3. Is the Search OR filter functionality present and working?
Search works wonderfully.

4. Is there a better way that data could be displayed on these pages? OR Could the style of the webpage be improved?
As mentioned by others, having links to the home page would be helpful so the user doesn't have to hit back many times.

5. Anything else that you think is important for READ functionalities?
Having a text field for entering the user id seems prone to errors. It might be better to have a drop down of all currently available user ids.

DELETE functionalities:
I don't think name/email are necessary for deleting a user account. What happens if the fields are not entered exactly as they are in the table?

Lee Rice

READ functionalities

Are rows being listed for all entities?
Yes, no problems here.

Are rows being listed for all relationships, as described in the Specs?
Yes, works as expected.

Is the Search OR filter functionality present and working?
Search by cuisine on the find restaurant page works fine and fulfils the requirements.

Is there a better way that data could be displayed on these pages?
OR Could the style of the webpage be improved?
Consider adding navigation links to each page so the user doesn't have to go back to home page each time.

Anything else that you think is important for READ functionalities?
I believe the search on the find restaurant page requires exact information to locate a restaurant. It might be better to allow the user to search for partial queries.

DELETE functionalities

Is DELETE functionality present and working?
Yes, I was able to delete a review.

Anything else that you think is important for DELETE functionalities?
Nope, delete is working as it should. No problems here.

William Darnell

READ functionalities
Are rows being listed for all entities? Yes
Are rows being listed for all relationships, as described in the Specs? Yes
Is the Search OR filter functionality present and working? Yes
Is there a better way that data could be displayed on these pages? OR Could the style of the webpage be

improved? A back or home button as mentioned above would make navigating less cumbersome. Anything else that you think is important for READ functionalities? nah, it all works well!

DELETE functionalities

Delete seems to work correctly!

Actions based on the feedback:

As Andrew suggested, I updated the query for finding a restaurant by name by using LIKE instead of =. Also, I added the rating to be displayed for the reviewed restaurants. I added the option to delete an individual review which resolved the issue of bad data still giving the review deleted message.

Russell mentioned that all my tables are not being displayed but he was unsure of the requirements. The project specs say "It should be possible to add entries to every table individually and every table should be used in at least one select query." so I do not believe I need to display all tables and as I noted previously about how I would like the customer and reviews to work to keep some privacy and restaurants from deleting reviews I do not want to display these two tables. I have used every table in a select query.

I will add a navigation bar before the next step's draft as suggested. The cuisine preference page does not currently work because that is my update query. I am not adding a drop down of userIDs as that is something I want the user to look up so that restaurants cannot try to use userIDs to erase negative reviews.

Upgrades to the Draft version:

I changed how the remove review worked. Instead of selecting a restaurant and entering the userID, the user now enters their userID and a table is displayed with all of their reviews with a delete button for each row.

Project Step 4

Feedback by the reviewers:

Christopher Brown

CREATE functionalities

1.) Does the INSERT form actually work for entities and relationships, as required in the Specs? When trying to add a review I received this message:

Error:Cannot add or update a child row: a foreign key constraint fails (`cs340_dunnbrit`.`reviews_restaurants_customers`, CONSTRAINT `reviews_restaurants_customers_ibfk_1` FOREIGN KEY (`customerID`) REFERENCES `customers` (`id`) ON DELETE SET NULL)

2.) Does INSERTing rows in the "M entity" of the 1-to-M relationship rows affect the INSERTing of rows in the "1 entity"? No, I did not have any issue with the cuisine table after inserting a new restaurant. I would say that it is working appropriately.

3.) Does INSERTing rows in the "M entity" of the M-to-M relationship rows affect the INSERTing of rows in the other "M entity"? I think there may be an issue with this, as noted above.

4.) Can data be inserted for all entities? See error above.

5.) Anything else that you think is important for the CREATE functionalities? It might be helpful to have a dropdown for restaurant location id to prevent users from entering an id that doesn't exist.

Eddie Kovsky

1. Does the INSERT form actually work for entities and relationships, as required in the Specs? Most forms seem to work correctly.
2. Does INSERTing rows in the "M entity" of the 1-to-M relationship rows affect the INSERTing of rows in the "1 entity"?
3. Does INSERTing rows in the "M entity" of the M-to-M relationship rows affect the INSERTing of rows in the other "M entity"? I'm seeing a similar error on the add review page:

Error:Cannot add or update a child row: a foreign key constraint fails (`cs340_dunnbrit`.`reviews_restaurants_customers`, CONSTRAINT `reviews_restaurants_customers_ibfk_1` FOREIGN KEY (`customerID`) REFERENCES `customers` (`id`) ON DELETE SET NULL)
4. Can data be inserted for all entities?

5. I wasn't sure what locationID means. Should that be an autoincrementing number hidden from the interface?

Suite should probably be allowed to be NULL; otherwise it populates a 0 when blank.

For next week's assignment, will it be possible to look up a user ID with only one piece of information?

Hillary Dreikorn

CREATE functionalities

1. Does the INSERT form actually work for entities and relationships, as required in the Specs? All, but 1.

2. Does INSERTing rows in the "M entity" of the 1-to-M relationship rows affect the INSERTing of rows in the "1 entity"? Not that I could tell.

3. Does INSERTing rows in the "M entity" of the M-to-M relationship rows affect the INSERTing of rows in the other "M entity"? Was not able to make a review for the reasons mentioned above.

4. Can data be inserted for all entities? All, but review.

5. Anything else that you think is important for the CREATE functionalities? maybe not putting the id number as something the user can put in, seems like most users would remember/or want to remember an id that identifies them

Good work! :)

Russell Moon

1. Does the INSERT form actually work for entities and relationships, as required in the Specs? I was able to submit to the customer, but I couldn't tell if anything had been added in the database because there's no comprehensive list for these tables (and I was unable to look up my customer ID). I was unable to submit to the review form (because I could not determine a User ID). I was able to add to restaurant, location and cuisine and can confirm that they're added to a persitent record.

2. Does INSERTing rows in the "M entity" of the 1-to-M relationship rows affect the INSERTing of rows in the "1 entity"? When I add a new cuisine, I am able to select it in the creation of a new restaurant. This is likewise true form the cuisine and customers relationship.

3. Does INSERTing rows in the "M entity" of the M-to-M relationship rows affect the INSERTing of rows in the other "M entity"? Adding the the restaurant table allowed me to select that new restaurant in the Add Review form. Because I'm unable to add a review, I cannot tell whether or not this updates the restaurants rating score

4. Can data be inserted for all entities? I can confirm inserts for restaurants, location and cusine, but cannot confirm for customers

5. Anything else that you think is important for the CREATE functionalities? I think it would be useful to list customers along with IDs on the Create Customer Profile page.

Andrew Huson

CREATE functionalities

1. Does the INSERT form actually work for entities and relationships, as required in the Specs? It appears creating a user works, finding the userID is not yet implemented but attempting to create a second user with duplicate information returns an error. Was able to add addresses, restaurants, and cuisines. Was unable to add a review.

2. Does INSERTing rows in the "M entity" of the 1-to-M relationship rows affect the INSERTing of rows in the "1 entity"? Works correctly

3. Does INSERTing rows in the "M entity" of the M-to-M relationship rows affect the INSERTing of rows in the other "M entity"? Unable to create or see reviews at this time.

4. Can data be inserted for all entities? No, unable to create review.

5. Anything else that you think is important for the CREATE functionalities? It appears there may be missing constraints on some of your attributes, these are also missing restrictions on fields. I was able to add an address with all attributes blank, zip code defaulted to 0. The zip code should also probably be a string, I attempted to enter an address with a zip code starting with 0 and the leading 0's were dropped.I was also able to create a user with a future birthdate.

Lee Rice

CREATE functionalities
Does the INSERT form actually work for entities and relationships, as required in the Specs?

I wasn't able to add a review. Entering an id and pressing submit gave an error. Couldn't verify if adding a customer worked or not as there doesn't seem to a table displayed. Adding restaurant, location, and cuisine type worked fine.

Does INSERTing rows in the "1 entity" of the 1-to-M relationship rows affect the INSERTing of rows in the "M entity"?
I was able to choose the same cuisine type for multiple restaurants. There doesn't seem to be a table showing cuisines, so not sure if it worked. The query in the data manipulation file looks correct though

Does INSERTing rows in the "M entity" of the M-to-M relationship rows affect the INSERTing of rows in the other "M entity"?
I don't think the M-to-M restaurant to customer relationship is working currently.

Anything else that you think is important for the CREATE functionalities?
Minor thing, but on the create account screen, Cuisine Preference says optional however there doesn't appear to be a null entry. From a usability standpoint, it might be better to populate the User ID box on the add review page with usernames rather having the user enter their own id.

William Darnell

Does the INSERT form actually work for entities and relationships, as required in the Specs? Received an error message about child rows.
Does INSERTing rows in the "M entity" of the 1-to-M relationship rows affect the INSERTing of rows in the "1 entity"? Unable to tell.
Does INSERTing rows in the "M entity" of the M-to-M relationship rows affect the INSERTing of rows in the other "M entity"? Unable to tell.
Can data be inserted for all entities? Unable to tell.
Anything else that you think is important for the CREATE functionalities? I agree with above points about a dropdown menu for location id, as well as I think it would be helpful for some sort of indication to happen when creating a profile.

Actions based on the feedback:

As suggested I added a dropdown box for the user to choose a location for a restaurant when adding a restaurant to the database. I fixed the location table by defaulting suite to NULL, so it would stop defaulting the suite number to 0. I added a on screen message that tells the customer their profile has been created. I had some suggestions that I should show a list of all the customers or not use the user IDs. I do not want to show a list of customer information because if this was a real-life application that would be a privacy issue and if customer information was available a restaurant would be able to use that to delete negative reviews of their restaurant. I also had feedback about getting rid of the user IDs. I think a lot of that feedback is because my look up user ID page did not work at the time of the draft, so it made many parts of my website difficult to use. I have made the look up user ID page functioning now. I am not getting rid of the user IDs because it is being used as a user identification process. In order for a user to retrieve their id they must know their name, birthdate, and email. Any user would typically know this information easily, but anyone trying to impersonate a user would have more difficulty retrieving the user ID. This serves two purposes. One, restaurants would not be able to

delete negative reviews without knowing the user ID associated with the review. The total dislikes and likes of a restaurant will be displayed on the website but never the reviewers' user IDs. Two, only users who have create profiles can leave reviews. Anyone can add a location, cuisine, or restaurant but a user must create a profile to add a review. This prevents users from reviewing the same restaurant multiple times. Users can remove their reviews and add a new review, but they cannot review a restaurant twice. I removed the optional option from cuisine preference when creating an account. I fixed the errors with the `reviews_restaurants_customers` table. My query had the values in the wrong order, so restaurant ID was being inserted into the customer ID and the other way around which was causing an error since the IDs being used did not exist in the opposite tables.

Upgrades to the Draft version:

I updated some of the display tables, so it could be verified that the different INSERT queries were working. I also made the look up user ID page functioning.

Project Step 3

Feedback by the reviewers:

Eddie Kovsky

Data Manipulation Queries

1. Queries are syntactically correct.

2. All the required functionality is covered.

HTML Pages

I don't see a url to your website in the PDF?

Andrew Huson

Are the queries syntactically correct? Yes - there are some that I don't think are necessary or might not work how you want them to, for example summing reviews where review = 0 will always equal 0, you might want to count(review) where review = 0 if you are looking for the number of dislikes? There is only one comment for both queries so I'm not sure whether this was intended to be two separate queries or not, if you want both on one table you could do something like SELECT SUM(review) AS 'number of likes', (COUNT(review)-SUM(review)) AS 'number of dislikes' FROM reviews_restaurant_customer;

Are there queries providing all functionalities as required by the CS340 Project Specs? Yes
Database has at least 4 entities and at least 4 relationships, one of which must be a many to many relationship? Yes
Possible to add entries to every table? Yes
Every table used in at least one select query? Yes
Website has the ability to search using text or filter using a dynamically populated list of properties to filter on? Yes
At least one delete function? Yes

At least one update function? Yes
Possible to add and remove things from at least one many-to-many relationship? Yes
Possible to add things to all relationships? Yes
Each functionality listed above have a corresponding HTML page? Yes
Is there a better way that data could be displayed on SHOW functionality pages? For removing
the review, you might consider using just one form where they input their user ID and select a
restaurant and then submit.
Is there a better way that the forms for UPDATE and ADD functionalities could be implemented?
See notes below, I think if you split the cuisine attributes from the restaurants and users into
separate tables with many to many relationships it would also require additional
pages/functions to modify those tables, add or remove preferences for users and cuisines for
restaurants.

Notes: Maybe this would make things too complicated for the project but you might consider
making a couple changes: Instead of having a rating that is a static attribute of an restaurant the
rating should be dynamically calculated with a query that divides the total number of likes by
the number of reviews (or returns 0 or NULL if there have been no reviews). You might also
want to create separate tables for cuisine to restaurant in case a restaurant serves more than
one cuisine, or customers to preferred cuisines as a customer can have more than one preferred
cuisine. Overall I think you have a really good idea and design.

Lee Rice

Data Manipulation Queries
Are the queries syntactically correct?
  I noticed in the queries, the 'restaurants' table is named 'restaurant', so those queries won't
work. Otherwise, looks good.

Are there queries providing all functionalities as required by the CS340 Project Specs?
  I like how you listed the page the queries belong too. Good organization. All queries
mentioned on the project spec appear to be present.

HTML Pages
Does each functionality listed in the CS340 Project Specs have a corresponding HTML page?
   Yes. All functionality appears to be present.

Is there a better way that data could be displayed on SHOW functionality pages?
   Looks great so far. No suggestions come to mind.

**Is there a better way that the forms for UPDATE and ADD functionalities could be
implemented?**
   My only question is what happens when the Add it button is pressed on the Add restaurant
page. It doesn't feel intuitive, but that's is likely just because the back end isn't hooked up
yet. Looks good otherwise.

Christopher Brown

Data Manipulation Queries

1.) Syntactically correct?
Table created was 'restaurants' but the DMQ reads 'restaurant'. I believe you want count(review) for the number of likes/dislikes

2.) All required queries?
Appears so. I see multiple SELECT queries (with at least one from each table), INSERT, UPDATE, and DELETE. Might be helpful to add a DELETE for cuisine.

HTML Pages

1.) Contains all required functionality?
Yup, looks good!

2.) Is there a better way the data can be displayed?
Links back to the home page would be appreciated. It may be helpful to add tables to some of the pages such as 'Add Cuisine Type' to show all of the currently added cuisine types.

3.) Is there a better way to UPDATE/ADD?
Not that I can think of.

Russell Moon

Hi Brittany, It seems as though most issues have been covered in the other peer reviews. For brevity's sake, I'll try to cover some new ground (or embellish what's already been discussed): 1) Repeating what one reviewer mentioned, maybe include a table list of current entries on each "Add" page, with a "Delete" button and an "Update" button appearing in each row, where appropriate. 2) Since I'm also doing a restaurant review db and have given this issue far too much thought, I'll second what another reviewer mentioned and suggest adding a cuisine table, making the relationship between restaurant and cuisine many-to-many. In addition to items that are typically understood to be cuisines (e.g., French, Mexican, Italian, etc.), I'm also including general restaurant descriptors in my cuisine table (e.g., casual, finedining, gastro-pub). "Cuisine" really doesn't capture what I'm going for, but I'd like to be able to basically tag a restaurant with a bunch of keywords that will show up in the restaurant description.

Hillary Dreikorn

I like how there's lots of food/restaurants based projects here. Food is great, haha!

DMQ:

In addition to those points mentioned above, I feel like the commenting could be a bit better. Other than that looks like it will work well for your site/

HTML:

Meets the specs, so there's not much to add there. I'm sure once you get to styling you'll had the feature of a back button and all that extra stuff. I like it it, looks good so far though.

Actions based on the feedback:

Eddie's feedback was not useful because he barely looked at my draft to find the URL and never updated his review when I told him where in the document the URL was. As Andrew suggested, I changed the query in the find_restaurant page to use COUNT and SUM to display the number of likes/dislikes in a table. Also, I fixed the display restaurant query to dynamically display the rating. Andrew suggested for removing a review I have user only enter their user ID and not all their information. I am not changing this because requiring a user to enter all their information can help to eliminate people who are not the user deleting their reviews. Andrew suggested splitting cuisines from restaurants and making the relationship many-to-many instead of many to one. I decided not to do this because I want the restaurants to be labeled only with one cuisine, so they can be identified more specially than broadly with multiple types of cuisines. I fixed the table name for queries using `restaurants` instead of `restaurant` as Lee mentioned. I did not do anything with the add button Lee mentioned because I think when the page is styled, it will make sense, but I am keeping it in mind for once we get to that step. Christopher suggested adding a home button to each page. I will do this in a later step when I can style it and add a navigation bar on each, but currently without styling it would just look more confusing to the user. As Russell suggested I added tables for all the current cuisine types and all current restaurants. I added the needed queries to display these as well.

Upgrades to the Draft version:

While working on this draft, I updated the foreign key on the reviews_restaurants_customers table. When customers are deleted from the database I did not want their reviews to also be deleted. So, on delete customerID is set to null and I updated customerID to allow null.

Project Step 2

Feedback by the peer reviewer:

The best peer review for ERD would answer all of the following questions.
Are the attributes for each entity in the ERD same as that described in the database outline?
    Yes, the attributes in the ERD match those in the outline.
Is the participation of entities in the relationships same as that described in the outline?
    Yes, the participation of restaurants to customers, restaurants to locations, and restaurants to cuisine all have correct participation. I am confused about the customers to cuisine relationship. If a customer will have only one preference, then must they have a preference? If a customer doesn't have to have a preference, then the ERD is correct.
Is the cardinality of entities in the relationships same as that described in the outline?
    Yes, the cardinality of the entities all match.
Is there something that could be changed/improved in the E R Diagram and/or the overall database design?
    No, I think it looks really well thought-out. I particularly like how the customers and cuisine entities are linked logically.

The best peer review for a Schema would answer all of the following questions:
Are the relationship tables present where required and correctly defined, when compared with the database outline?
    Yes, all the relationship tables are present and correctly defined.
Are foreign keys present where required and correctly defined, when compared with the database outline?
    Is the location attribute in the Restaurants entity a foreign key? Even though it is not explicitly defined as one in the outline, it seems like it would be.
Do the entity attributes match those described in the outline?
    Yes, all the entity attributes match those in the outline.
Is there something that could be changed/improved in the Schema and/or the overall database design?
    No, I think it is a very clean and efficient design that is well integrated.

The ideal peer review for a DDQ file would answer all of the following questions:
Is the SQL file syntactically correct? This can be easily verified by importing/copy-pasting it in phpmyadmin. (Do not forget to take backup of your own database before you do this!)
    Yes.
Are the data types appropriate considering the description of the attribute in the database outline?

Yes.
Are the foreign keys correctly defined when compared to the Schema?
Yes.
Are relationship tables present when compared to the ERD/Schema?
Yes, reviews_restaurants_customers is present and populated.

David Rider, Oct 10 at 7:23am

Hi Brittany, the changes you made based on previous reviewers were good. I like your ERD and Schema diagrams, thanks for using blue arrows (made it easier to see and distinguish from table borders). My only confusion is with the reviews. Once a restaurant entry is created, you say the review value defaults to 0. However, it's also stated that the review is based off 1 (like) vs 0 (dislike), so wouldn't it be unfair to new restaurants that haven't had any reviews yet? Unless you somehow "subtract" that initial 0 value from the average rating. Your SQL queries look nice, thanks for spacing them out and using several samples/examples. I like your use of the AVG function. Only thing I noticed was you don't drop any tables if it exists, so if you were to import this twice, some tables may be overwritten while others aren't. I was thinking perhaps you could put in DROP TABLE IF EXISTS so that you have a fresh table every time you import this file in.

Hyung Jun Kim, Oct 10 at 10:48am

Actions based on the feedback:

I went back in the outline and clarified that it is not required for a customer to have a cuisine preference. I made no changes to the ERD since it is already correct. I went back in the outline and labeled all the foreign keys, so it would be clearer.

I did not change the default rating of 0 even though it may seem unfair. There are a few reasons I did not change this. One I want it to be more obvious to the customer that a restaurant has no reviews yet. When this database is more fully implemented, I hope to set it up so if a customer is viewing the rating they can easily access the number of reviews and how they are divided among likes and dislikes. A customer is more likely to question why a restaurant has a 0 than a 100 which makes them more likely to look at the number of reviews. The second reason is I do not expect the default rating to last very long. Customers are the ones who will be adding the restaurants to the database, so I would expect that if a customer is going through the process of adding the restaurant it is because they are wanting to leave their review. The idea is the customer would add the restaurant and then review it shortly after. Finally, the default value will be deleted by the average function as soon as the first review is added so that the initial 0 value is not skewing the results. To clarify the default value of 0 in the rating attribute when the table is created is just a placeholder until the first review is added and then rating becomes a calculated value based on the reviews relationship table.

I did add drop table statements as suggested. I originally did not because I was worried about a peer reviewer accidently deleting their tables when importing mine in case they were named the same. I did see the instructions for peer reviewing do warn us about backing up our own databases, so drop table statements should be okay.

Upgrades to the Draft version:

When checking the foreign keys in the schema after peer comments, I noticed I forgot to underline CusinieID under the customer entity to indicate it is a key. I fixed this mistake.

Project Step 1

Feedback From Peer Reviewers:

Comments:
Outline makes sense and is understandable why you would create a database. There are enough entities but I don't see a reason why addresses need to be their own entity. It would probably be fine to just attach it to the restaurant entity. Review entity is somewhat confusing. If a customer enters a restaurant, will they automatically be set to like it? A record could be created if a user clicks a like, with a person id and a restaurant id in the record. If recording both likes and dislikes, the boolean attribute can stay in the record as well.

Jonathan Abantao, Oct 1 at 11:18pm

Hi Brittany, It is strange my database is based on menus for restaurants and the other student I was assigned is also doing a food related database. I am not sure of your intentions for this database but for the sake of the project you could probably simplify this a bit. You could have the restaurants location reduced to latitude and longitude or just use a generic "location ID". If this is a database to track restaurants that the user likes perhaps a personal note or favorite meal would be helpful to remember. I know I rarely know what I ordered but just that I had a good meal.

Jacob Powers, Oct 2 at 4:36pm

Actions Based on Feedback:

I am leaving location as an entity because location is a thing which is made up of attributes and will be stored together in a table. Another reason I am leaving location as an entity is location should be unique and restaurants cannot be at the same location. The restaurant entity table will show the one-to-one relationship between the location entity and restaurant entity. In this table I would use a location id (as mentioned by the reviewer) and restaurant id. I do not want to change location to latitude and longitude as I want location to be something that can easily be looked up and used by a customer for locating a restaurant.

I tried to clarify how the review process would work in the project outline and I changed the review to not have a default choice. If a customer selects a restaurant to review, they will then choose between a like (1) and dislike (0), but their entry cannot be null. This database is less to track a customer's restaurants they like and more to help customers choose a restaurant based on their cuisine preference, location, and other customer reviews.

Upgrades to Draft Version:

I added more detail to the project outline to better explain how I would like this database to work as there seemed to be a bit of confusion from the reviewers. I also changed reviews to an attribute instead of an entity because I realized it was an attribute of the relationship between restaurants and customers. While working on project step 2 I realized there was not a Boolean data type, so I changed this to a bit which cannot be null.

Project Outline

This database will represent a review and information system for restaurants and their customers. Customers can review restaurants which will generate a rating for the restaurant. Customers must add their information in order to add a review. Restaurants will have information about their name, location, cuisine, and rating.

This database would work as follows:

- customers would create an account with their information (name, email, birthdate, and preferences)
- restaurants would be added to the database by customers and would include the restaurant information (name, location, rating, and cuisine)
    - name of restaurant would be entered by customer
    - location would be added when a restaurant is added, but it cannot be the same as a location already in the database
    - rating would default to 0 at creation because no reviews exist
    - cuisine can be chosen from the already existing cuisines or a new cuisine type can be created
- Customers can select an existing restaurant and choose either like or dislike. Restaurants are reviewed after creation and never during. During creation the rating (which is made up of reviews) is defaulted to 0.
- Customers would be able to find all restaurants matching their cuisine preference or a certain rating
- When a customer looks up a restaurant they would receive its information including location, rating (which is the average like rate from customers who have left reviews), and cuisine type

The focus of this database is to allow customers to give their opinion of a restaurant in the form of a like or dislike which in turn will help other customers choose a restaurant. Customers searching for a restaurant narrow down their options by location, cuisine preferences, and a restaurant's rating.

Database Outline

Entities and Attributes:

- **Restaurants**
    - **Id:** This is an auto-incremented, not null, and unique number used to identify a restaurant. This is the primary key.
    - **Name:** This is the restaurant's name. This is a string with a max of 200 characters. It cannot be blank or null.
    - **Rating:** This is an average rating based on customer reviews. This is a decimal number to two decimal places to represent a percentage. It cannot be blank or null. The numbers will be calculated as an average of likes (value of 1) and dislikes (value of 0) based on the number of customers who left reviews. When a restaurant is first added to the database the default is 0.
    - **Location:** This is the physical location of the restaurant. This will be an id of the location entity. Only one restaurant can be at any given location, so the id must be unique. Location cannot be null or blank. This is a foreign key. A location cannot be deleted if it is linked to a restaurant. If a location id is updated in the location table then it will be updated here too.
    - **Cuisine:** This is the cuisine which the restaurant serves. This will be an id of the cuisine entity. The cuisine does not have to be unique. This is a foreign key. A location cannot be deleted if it is linked to a restaurant. If a location id is updated in the location table then it will be updated here too

- **Location**
  - o **Id:** This is an auto-incremented, not null, and unique number used to identify a location. This is the primary key.
  - o **Address**

    This is the physical address of a restaurant.
    - ▪ **Street:** This is the street name of the restaurant. This is a string with a max of 255 characters. This cannot be null or blank.
    - ▪ **Suite number:** This is the suite number of the restaurant. This can be blank. The default is null.
    - ▪ **City:** This is the city where the restaurant is located. This cannot be null or blank. This is a string with a max of 100 characters.
    - ▪ **State:** This is the state abbreviation where the restaurant is located. This cannot be null or blank. This is a string of 2 characters.
    - ▪ **Zip code:** This is the zip code of where the restaurant is located. This is an integer with 5 numbers. This cannot be blank or null.
- **Customers**
  - o **Id:** This is an auto-incremented, not null, and unique number used to identify a customer. This is the primary key.
  - o **First name:** This is the customer's first name. This is a string with the max of 100 characters. This cannot be blank or null.
  - o **Last name:** This is the customer's last name. This is a string with the max of 100 characters. This cannot be blank or null.
  - o **Email:** This is the customer's email. This is a string with the max of 100 characters. This cannot be blank or null. This must be unique.
  - o **Birthdate:** This is the customer's birthdate. This is a date in the format YYYY-MM-DD. It will default to 0000-00-00.
  - o **Preferences:** This is the customer's cuisine preference. This will be an id of the cuisine entity. Customers are not required to have a cuisine preference. The default is null. This is a foreign key. If a cuisine is updated within the cuisine table then it will be updated here as well. If a cuisine from the cuisine table is deleted, then it will be set to null here.
- **Cuisine**
  - o **Id:** This is an auto-incremented, not null, and unique number used to identify a cuisine. This is the primary key.
  - o **Type:** This is a description of the type of food a restaurant serves. This is a string with a max of 255 characters.
- **Reviews (This is an attribute of the relationship between restaurants and customers. It will be used to derive the rating attribute of restaurants)**
  - o **Like/Dislike:** This represents the customers review of a restaurant. This is an int. 1 for like and 0 for dislike. This cannot be null. To clarify reviews are only likes or dislikes they will not include comments.

Relationships:

- **Restaurants serve customers:** Many restaurants have many customers. This is a many-to-many relationship
    - **And these customers generate reviews:** This is an attribute of the restaurant-customer relationship
    - **And these reviews create a rating:** A restaurant's rating is calculated from reviews left by its customer
    - **Example of this relationship:**

Restaurant ID	Customer ID	Review
ABC	123	1
DEF	123	0
ABC	456	0

      Restaurant- ID: ABC **Rating:** 50%

- **Restaurants are at locations:** Every restaurant has a single location which belongs to it. This is a one-to-one relationship.
- **Restaurants specialize cuisine:** A restaurant serves one cuisine, but a cuisine can be served at many restaurants. This is a one-to-many relationship.
- **Customers have cuisine preferences:** Every customer has a single cuisine preference, but a cuisine can be a preference for many customers. This is a one-to-many relationship.