# Sean McLean

# ALY 6030

# Assignment 1

# Tech Crunch

## Part 1: Normalization

1. **What is a good choice for a primary key here? In contrast, give an example of an attribute (or composite) that would *not* be a valid primary key.**

   My selection for the primary key in this dataset would be the 'fund ID' attribute because every row will have a unique identification number that will not be duplicated in another row. Some other qualities that it has are that it does not have any atomic values contained in any of the cells and it will provide stability overall. An attribute that would probably not be a valid primary key for the dataset would be the 'company' column because it could contain duplicate entries with other attributes and therefore would not be unique in its ability to identify the rows. An example of this from the dataset is the company iLike that has two funding dates of 01/01/2006 but have different amounts raised as shown below.

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 89 | 492 | iLike | 28 web | Seattle | WA | 1/1/2006 | 2500000 | USD | a |
| 90 | 493 | iLike | 28 web | Seattle | WA | 1/1/2006 | 13300000 | USD | b |

2. **Does the table satisfy 1NF? Why or why not?**

   After examining the dataset, the table satisfies the first normal form because there are no duplicate rows contained. It also satisfies the 1NF requirements because there are not any atomic values in any of the cells and there are not any repeating groups.

3. **Does the table satisfy 2NF? Why or why not?**

   The table satisfies the second normal form because all of the rows in the dataset are unique and there are no repeating groups. There are also no partial dependencies and composite primary keys with the 'fund id' attribute being the primary key. All the other columns in the dataset rely on the primary key and not just a portion of it.

4. **Does the table satisfy 3NF? Why or why not?**

   The table does not meet the requirements of the third normal form due to evidence that there are some transitive dependencies among the non-key attributes in the dataset. A couple examples of this are with the 'company' attribute which could determine the 'city' and 'state' columns as well as the attribute of the 'company.' Another example is with the attribute 'raised currency' which can depend on the 'raised amount' attribute. These transitive dependencies would need to be removed, and two different tables would need to be created. One table would show company information like 'city,' 'state,' and 'category' and the other would include funding data like 'fund id,' 'company,' 'fund date,' 'raised amount,' etc. This process would lead to the table satisfying the third normal form requirements.

5. **Sketch a proposed Entity-Relationship diagram that would bring this dataset into 3NF. If you answered "yes" to (4), for example, your ERD would just be the raw data table with no changes. If, however your ERD requires multiple tables to be in 3NF, you should draw all relationships between them and indicate their type (one-to-one, one-to-many, etc.)**

| Company | | Fund |
|---|---|---|
| company_id (PK) | 1 <------M | company_id (FK) |
| company_name | | fund_id (PK) |
| category | | numEmps |
| city | | fundedDate |
| state | | raisedAmt |
| | | raisedCurrency |
| | | round |

## Part 2: Case Study

1. Once you run the sql code you'll notice that each table has data populated for two recipes, Chicken Marsala, and Absolute Brownies.
   Use the INSERT INTO statement to insert new information about **two (2) completely new recipes of your choosing into the database.**
   You can make up your own recipes or copy them from a website. **Be sure to insert values into each of the tables generated by the recipe.sql file: recipe_main, rec_ingredients, ingredients, and categories.**
   Use Chicken Marsala and Absolute Brownies as your guides for the problem as the instructions follow similar steps as what was already done in the code, but remember you need to do the same thing for two new recipes.

2. **Write only one SQL query that returns all information on only the two new recipes you inserted from all the tables you created in step 1 above.**
   **Y**our query should show all relevant information from all four of the tables from step 1. Don't worry that the output table may duplicate rows, for example the recipe_main table will duplicate rows for each of the ingredients you enter which is ok. The idea is to get you comfortable with joins. Show the results of your query as screenshots pasted in below:

```
143         #Question 2
144 •       SELECT
145         recipe_main.recipe_id,
146         recipe_main.rec_title,
147         recipe_main.recipe_desc,
148         recipe_main.prep_time,
149         recipe_main.cook_time,
150         recipe_main.servings,
151         recipe_main.difficulty,
152         recipe_main.directions,
153         categories.category_name,
154         ingredients.ingred_name,
155         rec_ingredients.amount
156     FROM
157         recipe_main
158     JOIN
159         categories ON recipe_main.category_id = categories.category_id
160     JOIN
161         rec_ingredients ON recipe_main.recipe_id = rec_ingredients.recipe_id
162     JOIN
163         ingredients ON rec_ingredients.ingredient_id = ingredients.ingredient_id
164     WHERE
165         recipe_main.rec_title IN ('Huevos Rancheros', 'Gnocchi with Brussels Sprouts');
166
```

**Output:**

| recipe_id | rec_title | recipe_desc |
|---|---|---|
| 3 | Huevos Rancheros | Traditional Mexican breakfast with eggs and salsa. |
| 3 | Huevos Rancheros | Traditional Mexican breakfast with eggs and salsa. |
| 3 | Huevos Rancheros | Traditional Mexican breakfast with eggs and salsa. |
| 3 | Huevos Rancheros | Traditional Mexican breakfast with eggs and salsa. |
| 3 | Huevos Rancheros | Traditional Mexican breakfast with eggs and salsa. |
| 4 | Gnocchi with Brussels Sprouts | Gnocchi with roasted Brussels sprouts and garlic. |
| 4 | Gnocchi with Brussels Sprouts | Gnocchi with roasted Brussels sprouts and garlic. |
| 4 | Gnocchi with Brussels Sprouts | Gnocchi with roasted Brussels sprouts and garlic. |
| 4 | Gnocchi with Brussels Sprouts | Gnocchi with roasted Brussels sprouts and garlic. |
| 4 | Gnocchi with Brussels Sprouts | Gnocchi with roasted Brussels sprouts and garlic. |

| prep_time | cook_time | servings | difficulty | directions | category_name | ingred_name | amount |
|---|---|---|---|---|---|---|---|
| 10 | 10 | 2 | 1 | Heat tortillas, cook eggs, layer with salsa and t... | Breakfast | Egg | 2.00 |
| 10 | 10 | 2 | 1 | Heat tortillas, cook eggs, layer with salsa and t... | Breakfast | Tortilla | 2.00 |
| 10 | 10 | 2 | 1 | Heat tortillas, cook eggs, layer with salsa and t... | Breakfast | Salsa | 0.50 |
| 10 | 10 | 2 | 1 | Heat tortillas, cook eggs, layer with salsa and t... | Breakfast | Refried Beans | 0.50 |
| 10 | 10 | 2 | 1 | Heat tortillas, cook eggs, layer with salsa and t... | Breakfast | Avocado | 0.25 |
| 15 | 20 | 4 | 3 | Cook gnocchi, roast Brussels sprouts with garlic... | Italian | Gnocchi | 1.00 |
| 15 | 20 | 4 | 3 | Cook gnocchi, roast Brussels sprouts with garlic... | Italian | Brussels Sprouts | 1.00 |
| 15 | 20 | 4 | 3 | Cook gnocchi, roast Brussels sprouts with garlic... | Italian | Garlic | 1.00 |
| 15 | 20 | 4 | 3 | Cook gnocchi, roast Brussels sprouts with garlic... | Italian | Parmesan Cheese | 0.25 |
| 15 | 20 | 4 | 3 | Cook gnocchi, roast Brussels sprouts with garlic... | Italian | Olive Oil | 2.00 |

3. Write a SELECT query that identifies the recipe name, category name, ingredient name, and ingredient amount. No other variables should be included.
   Your output should be sorted first by descending category name, then by ascending recipe name, followed by descending ingredient name.