Project Phase-I Report

Table of Contents

Vikas Seth (Team 140)

vseth3@illinois.edu

Dataset

I will be using NYPL Menu data set for the final project which is available at http://menus.nypl.org/. This is dataset of Restaurant Menu collection dating from 1840s collected by New York Public Liberary. This collection is one of the largest in the world, used by historians, chefs, noveists and every food enthusiasts. This collection mainly contains details about different Menus from the past along with details of dishes, price range, menu size, location, sponsor details of the Menus.

Target Use Cases

U_1 - Main Use Case

One of the interesting use case with this data-set could be about dishes. Like when a dish appeared for the first time and on which Menu, who was the sponsor and which year it appeared. For this we will need to clean this dataset to make sure that we have cleaned information about name of the dishes, Restaurants, sponsor and year information. Based on that we should be able to search this database to filter out this details for any dish.

U_0 - No Data Cleaning Required

To find out average number of dishes in Menu for each by each year or decade.

U_2 - Data Clearning not Sufficient

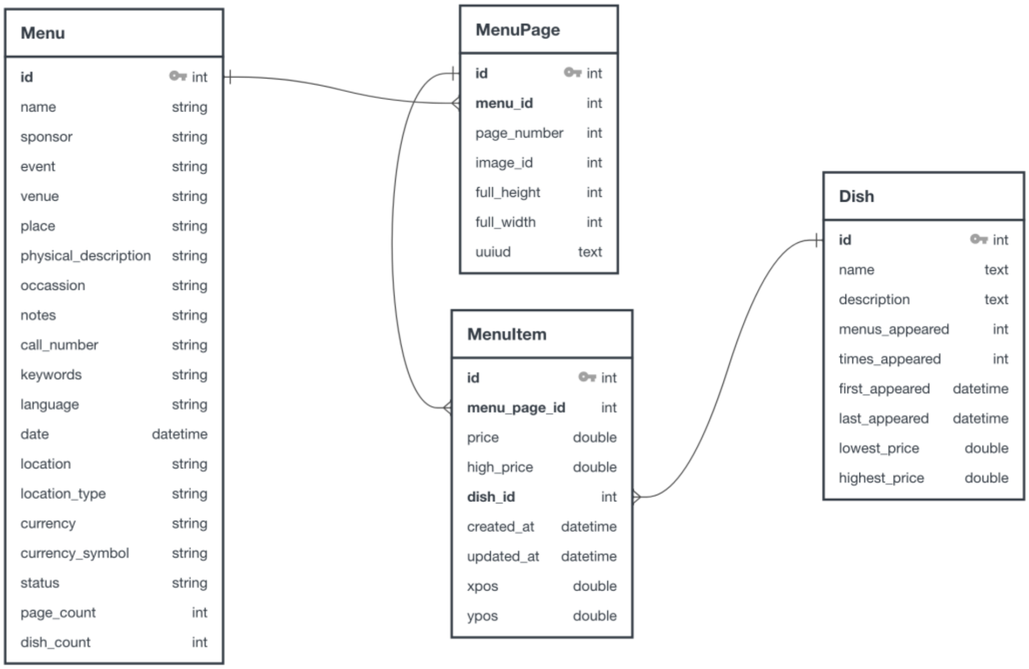
Origin Language of the Menu. This information is available for some of the menus in Notes field, but it is not available for all of them. Also, there is a language column. But this information is not available there as well. So, even if we try to clean the data under notes column we will not be able to find this information for all the Menus.

Dataset Details

This is data set for Menu details collected by NYPL with the help of Crowdsourcing. This menu collection go back to the year 1840 and contains details of Menus like dishes, occassion, sponsor, venue type, place, location, number of dishes along with physical properties of Menu like number of pages, bind type etc. This dataset mainly contains data in following 4 files

1. **Menu** - It contains overall details of each menu like Date, Sponsor, Event, Venue, Restaurant, Number of pages, Number of dishes in the Menu
2. **Menu Page** - Unique key in this set is Page id and for each Menu along with Page Number it contains Image details and dimensions of the Menu
3. **Menu Item** - Contains details of items/dishes, price, and dish id in a Menu
4. **Dish** - It contains details of the dish including name, description, number of occurrence in Menus, Occurrence timeline and price details.

Basic database schema of this dataset is as follows



Menu	
id	int
name	string
sponsor	string
event	string
venue	string
place	string
physical_description	string
occassion	string
notes	string
call_number	string
keywords	string
language	string
date	datetime
location	string
location_type	string
currency	string
currency_symbol	string
status	string
page_count	int
dish_count	int

MenuPage	
id	int
menu_id	int
page_number	int
image_id	int
full_height	int
full_width	int
uuiud	text

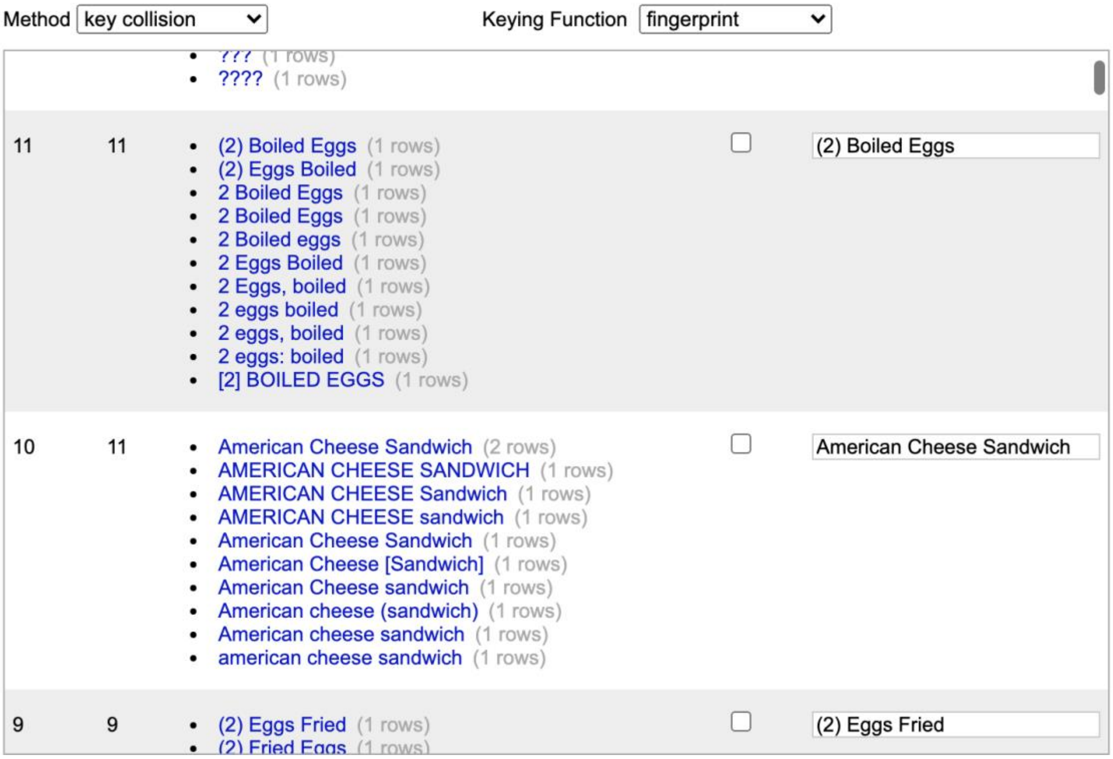
MenuItem	
id	int
menu_page_id	int
price	double
high_price	double
dish_id	int
created_at	datetime
updated_at	datetime
xpos	double
ypos	double

Dish	
id	int
name	text
description	text
menus_appeared	int
times_appeared	int
first_appeared	datetime
last_appeared	datetime
lowest_price	double
highest_price	double

Data Quality Problems

At minimum we will need to address following data quality problems with this dataset to make it usable for use case U_1

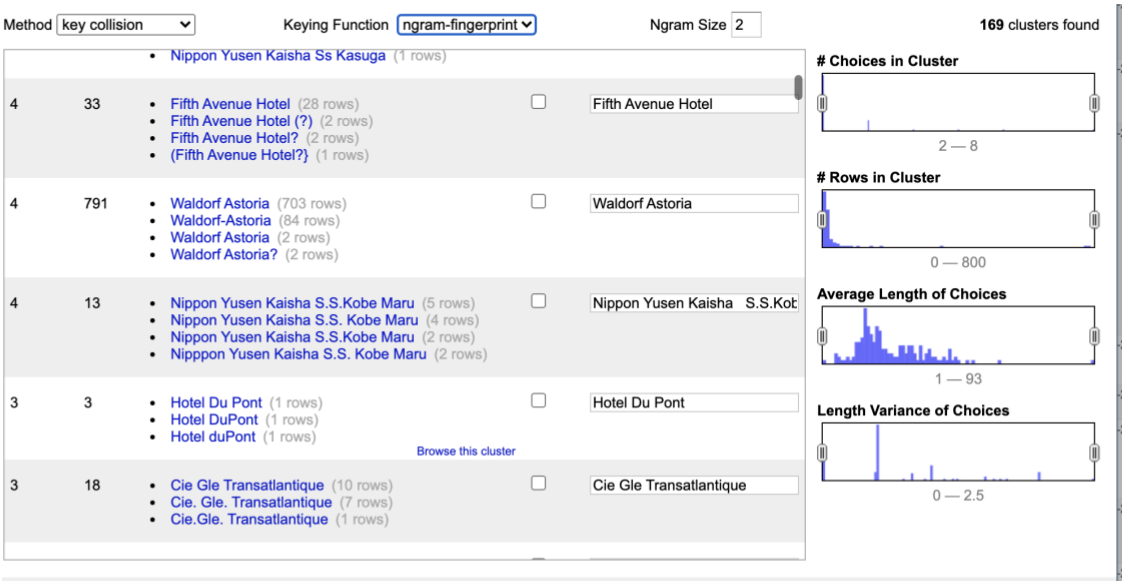
1. ***Cleaning up Dishes name***: Below is the screenshot of few examples name of the dishes we will need to clean.



Method key collision Keying Function fingerprint

		??? (1 rows) ???? (1 rows)		
11	11	(2) Boiled Eggs (1 rows) (2) Eggs Boiled (1 rows) 2 Boiled Eggs (1 rows) 2 Boiled Eggs (1 rows) 2 Boiled eggs (1 rows) 2 Eggs Boiled (1 rows) 2 Eggs, boiled (1 rows) 2 eggs boiled (1 rows) 2 eggs, boiled (1 rows) 2 eggs: boiled (1 rows) [2] BOILED EGGS (1 rows)	□	(2) Boiled Eggs
10	11	American Cheese Sandwich (2 rows) AMERICAN CHEESE SANDWICH (1 rows) AMERICAN CHEESE Sandwich (1 rows) AMERICAN CHEESE sandwich (1 rows) American Cheese Sandwich (1 rows) American Cheese [Sandwich] (1 rows) American Cheese sandwich (1 rows) American cheese (sandwich) (1 rows) American cheese sandwich (1 rows) american cheese sandwich (1 rows)	□	American Cheese Sandwich
9	9	(2) Eggs Fried (1 rows) (2) Fried Eggs (1 rows)	□	(2) Eggs Fried

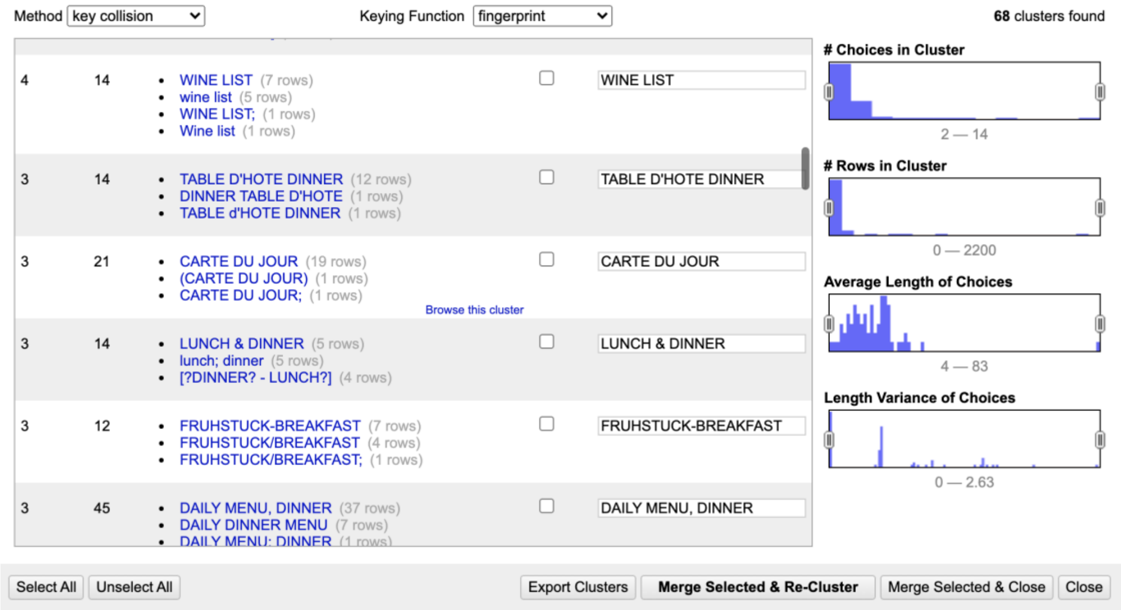
2. ***Cleaning up Sponsor/Location Details***: Below is the screenshot of few examples Sponsor/Location names we need to clean up



Method key collision Keying Function ngram-fingerprint Ngram Size 2 **169** clusters found

		Nippon Yusen Kaisha Ss Kasuga (1 rows)		
4	33	Fifth Avenue Hotel (28 rows) Fifth Avenue Hotel (?) (2 rows) Fifth Avenue Hotel? (2 rows) (Fifth Avenue Hotel?} (1 rows)	□	Fifth Avenue Hotel
4	791	Waldorf Astoria (703 rows) Waldorf-Astoria (84 rows) Waldorf Astoria (2 rows) Waldorf Astoria? (2 rows)	□	Waldorf Astoria
4	13	Nippon Yusen Kaisha S.S.Kobe Maru (5 rows) Nippon Yusen Kaisha S.S. Kobe Maru (4 rows) Nippon Yusen Kaisha S.S.Kobe Maru (2 rows) Nipppon Yusen Kaisha S.S. Kobe Maru (2 rows)	□	Nippon Yusen Kaisha S.S.Kob
3	3	Hotel Du Pont (1 rows) Hotel DuPont (1 rows) Hotel duPont (1 rows) Browse this cluster	□	Hotel Du Pont
3	18	Cie Gle Transatlantique (10 rows) Cie. Gle. Transatlantique (7 rows) Cie.Gle. Transatlantique (1 rows)	□	Cie Gle Transatlantique

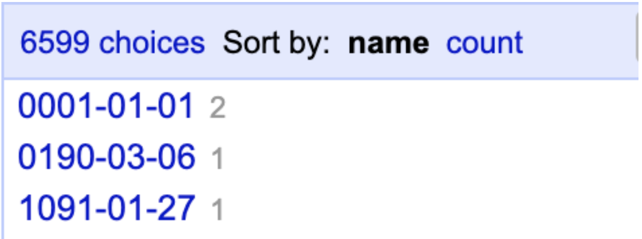
3. ***Clean Event Types***: Below is the screenshots for sample event types names required cleaning



Method key collision Keying Function fingerprint **68** clusters found

4	14	• WINE LIST (7 rows) • wine list (5 rows) • WINE LIST; (1 rows) • Wine list (1 rows)	☐	WINE LIST
3	14	• TABLE D'HOTE DINNER (12 rows) • DINNER TABLE D'HOTE (1 rows) • TABLE d'HOTE DINNER (1 rows)	☐	TABLE D'HOTE DINNER
3	21	• CARTE DU JOUR (19 rows) • (CARTE DU JOUR) (1 rows) • CARTE DU JOUR; (1 rows) Browse this cluster	☐	CARTE DU JOUR
3	14	• LUNCH & DINNER (5 rows) • lunch; dinner (5 rows) • [?DINNER? - LUNCH?] (4 rows)	☐	LUNCH & DINNER
3	12	• FRUHSTUCK-BREAKFAST (7 rows) • FRUHSTUCK/BREAKFAST (4 rows) • FRUHSTUCK/BREAKFAST; (1 rows)	☐	FRUHSTUCK-BREAKFAST
3	45	• DAILY MENU, DINNER (37 rows) • DAILY DINNER MENU (7 rows) • DAILY MENU: DINNER (1 rows)	☐	DAILY MENU, DINNER

4. ***Date Values***: There seems to some date values outside range will require some cleaning, like following

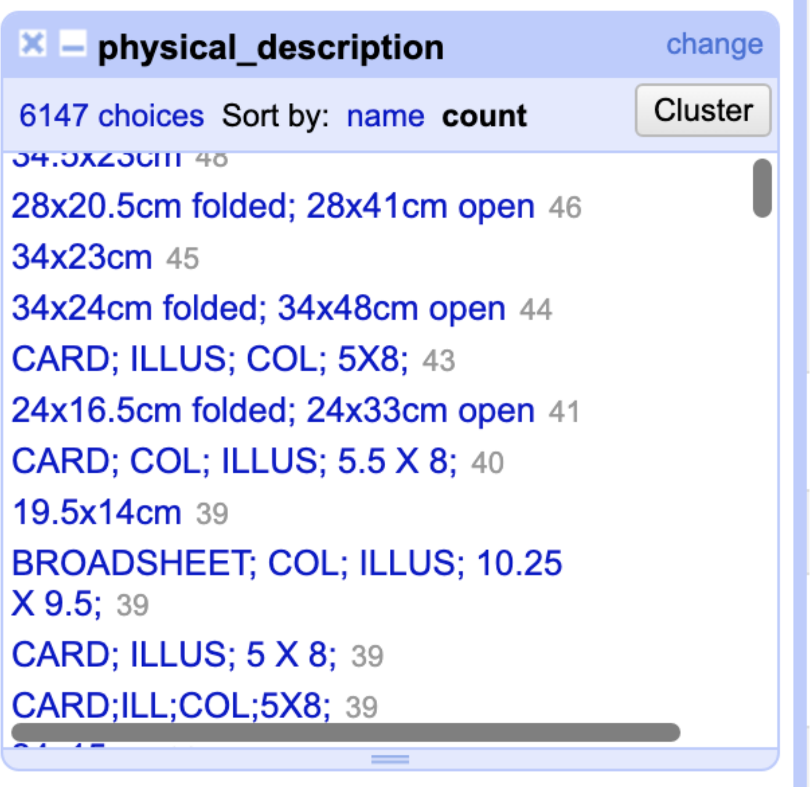


6599 choices Sort by: **name** count

0001-01-01	2
0190-03-06	1
1091-01-27	1

5. ***Additional Cleaning***: Apart from fields needed for usecase U1. There are some more fields can be cleaned and provide more details about menus. Like following

    1. *Physical Details of the Menus*: Seems like following can be cleaned and we can at-least get some information about Menu build like Card, Folder, etc.



physical_description change

6147 choices Sort by: name **count** Cluster

34.5x23cm 46

28x20.5cm folded; 28x41cm open 46

34x23cm 45

34x24cm folded; 34x48cm open 44

CARD; ILLUS; COL; 5X8; 43

24x16.5cm folded; 24x33cm open 41

CARD; COL; ILLUS; 5.5 X 8; 40

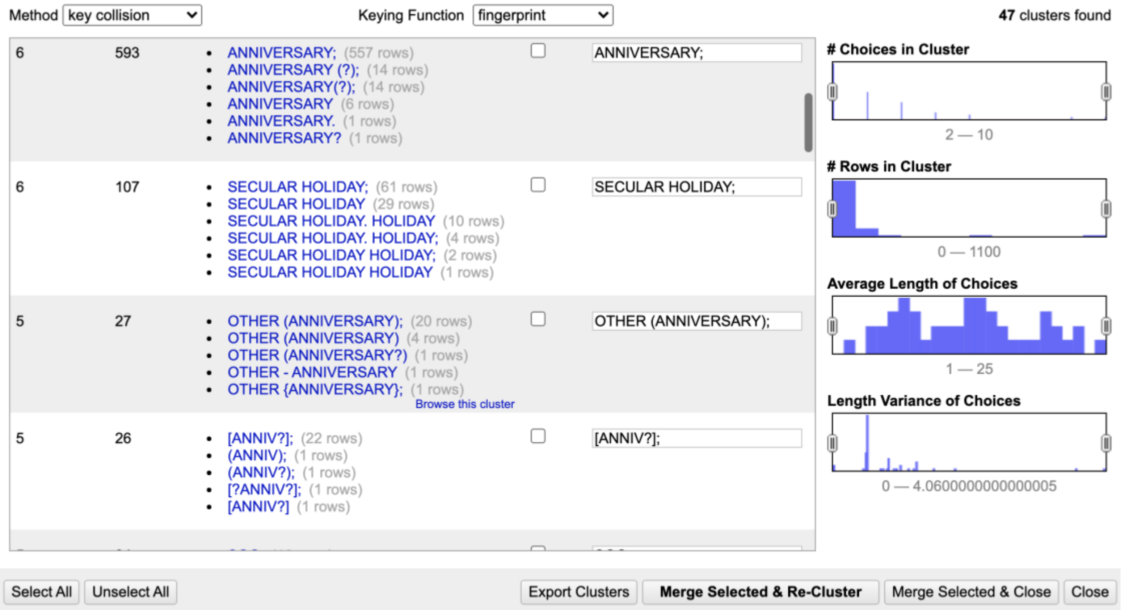
19.5x14cm 39

BROADSHEET; COL; ILLUS; 10.25 X 9.5; 39

CARD; ILLUS; 5 X 8; 39

CARD;ILL;COL;5X8; 39

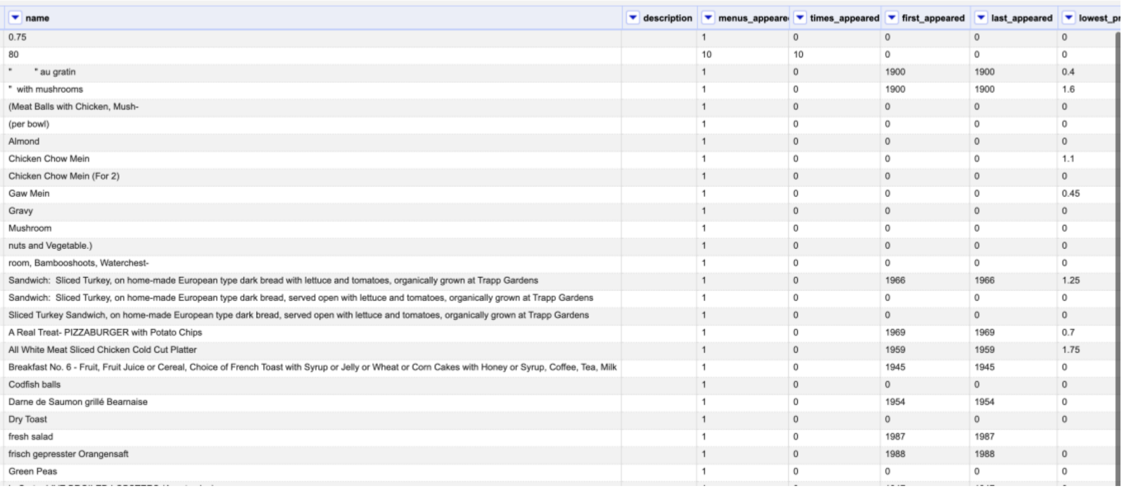
2. *Occasion*: Occasion details can also be cleaned to get the details if Menu was used for certain Occasion



Method key collision Keying Function fingerprint **47** clusters found

6	593	ANNIVERSARY; (557 rows) ANNIVERSARY (?); (14 rows) ANNIVERSARY(?); (14 rows) ANNIVERSARY (6 rows) ANNIVERSARY. (1 rows) ANNIVERSARY? (1 rows)	□	ANNIVERSARY;
6	107	SECULAR HOLIDAY; (61 rows) SECULAR HOLIDAY (29 rows) SECULAR HOLIDAY. HOLIDAY (10 rows) SECULAR HOLIDAY. HOLIDAY; (4 rows) SECULAR HOLIDAY HOLIDAY; (2 rows) SECULAR HOLIDAY HOLIDAY (1 rows)	□	SECULAR HOLIDAY;
5	27	OTHER (ANNIVERSARY); (20 rows) OTHER (ANNIVERSARY) (4 rows) OTHER (ANNIVERSARY?) (1 rows) OTHER - ANNIVERSARY (1 rows) OTHER {ANNIVERSARY}; (1 rows) Browse this cluster	□	OTHER (ANNIVERSARY);
5	26	[ANNIV?]; (22 rows) (ANNIV); (1 rows) (ANNIV?); (1 rows) [?ANNIV?]; (1 rows) [ANNIV?] (1 rows)	□	[ANNIV?];

3. *Additional Cleaning for Dishes*: There are certain dishes having 0 count for Menu occurrence and not year information. We should be able to drop those from our analysis



name	description	menus_appeare	times_appeared	first_appeared	last_appeared	lowest_pr
0.75		1	0	0	0	0
80		10	10	0	0	0
" " au gratin		1	0	1900	1900	0.4
" with mushrooms		1	0	1900	1900	1.6
(Meat Balls with Chicken, Mush-		1	0	0	0	0
(per bowl)		1	0	0	0	0
Almond		1	0	0	0	0
Chicken Chow Mein		1	0	0	0	1.1
Chicken Chow Mein (For 2)		1	0	0	0	0
Gaw Mein		1	0	0	0	0.45
Gravy		1	0	0	0	0
Mushroom		1	0	0	0	0
nuts and Vegetable.)		1	0	0	0	0
room, Bambooshoots, Waterchest-		1	0	0	0	0
Sandwich: Sliced Turkey, on home-made European type dark bread with lettuce and tomatoes, organically grown at Trapp Gardens		1	0	1966	1966	1.25
Sandwich: Sliced Turkey, on home-made European type dark bread, served open with lettuce and tomatoes, organically grown at Trapp Gardens		1	0	0	0	0
Sliced Turkey Sandwich, on home-made European type dark bread, served open with lettuce and tomatoes, organically grown at Trapp Gardens		1	0	0	0	0
A Real Treat- PIZZABURGER with Potato Chips		1	0	1969	1969	0.7
All White Meat Sliced Chicken Cold Cut Platter		1	0	1959	1959	1.75
Breakfast No. 6 - Fruit, Fruit Juice or Cereal, Choice of French Toast with Syrup or Jelly or Wheat or Corn Cakes with Honey or Syrup, Coffee, Tea, Milk		1	0	1945	1945	0
Codfish balls		1	0	0	0	0
Darne de Saumon grillé Bearnaise		1	0	1954	1954	0
Dry Toast		1	0	0	0	0
fresh salad		1	0	1987	1987	
frisch gepresster Orangensaft		1	0	1988	1988	0
Green Peas		1	0	0	0	0

Initial Plan

Cleaning Steps

1. Convert data to appropriate formats like Date, number or text.
2. Using OpenRefine clean the text fields for leading and trailing spaces and also clean for consecutive spaces
3. Remove any special characters using OpenRefine and Regular Expression
4. Using OpenRefine use clustering methods to clean the fields like Dishes name, Sponsor, Location, Event, Occasion
5. Trying cleaning Physical Description of the Menu using Regular Expression to find Build Type of the Menu
6. Using SQLite build the database schema for the dataset to check for ICs like:
    1. All ids should be unique for Dishes, Menus, Menu Pages
    2. ids should not be null for Dishes, Menus, Menu Pages
    3. Check for the dishes where last appeared or first appeared is 0
    4. Lowest Price should not be greater than Highest Price
    5. First Appeared should not be greater than last appeared
    6. Page count in a Menu should not be null or 0