Kimball Lifecycle Proposal Template

CIS 9440 - Data Warehousing for Analytics
Final Project Milestone 1
Group Number - **17**
Student(s) – **KWANG HEUM YEON**

This Proposal is the beginning of your semester-long Final Project. The goal of the project is to develop a working Data Warehouse using a commercial database management system. Your project will use data from a public source(s), transform the data into a dimensional model inside your Data Warehouse, and connect to a Business Intelligence application to produce valuable, actionable insights.

For motivation on project ideas, **think about interesting problems, opportunities, or insights that could be shown, solved, or highlighted with data about New York City**. Search for datasets on NYC Open Data (https://opendata.cityofnewyork.us/) that interest you and your group. You may need to combine datasets from NYC Open Data to address your desired problem or opportunity. Below are just a few **examples** of Project Ideas:

1. NYC Motor Vehicle Collision Reduction Project
2. NYC Parking Ticket Transparency Project
3. NYC Transportation Optimization Project

Your ideas may be far more creative and combine multiple datasets to achieve a goal.

To complete this Milestone, please fill in all bolded sections below:

Data Warehouse Project Title:

New York City Bicycle Operation Data Warehouse

Motivation for Project idea:

I'd love my neighbors to have a healthy life and save time by having the current bicycle operating system optimized.

Description of the issues or opportunities the project will address:

New York City is one of the most densely populated areas having lots of traffic congestion at almost every street over the year. To alleviate the traffic jam and promote an eco-friendly policy to residences, the city has been installed and executed a bicycle self-check-in and out system. However, it is questionable if the number of available bicycles is demographically optimized considering my previous experience.
To answer this question, New York City Bicycle Operation Data Warehouse will combine bicycle count, location, zip code, and census to uncover insights regarding bicycle count per a New York City resident.

Business Justification:

High-level Business Initiative:

Determine the number of bicycles registered in the system at each site in New York City and compare it with the number of residents. BPR(Bicycle per Person Ratio) information will be shared for the sake of maintenance and development of the bicycle operation.

BI Sponsors and Stakeholders (who will own this project?)

New York City Department of Transportation (NYC DOT)

What's the Business Value?

The primary value of this project is to reduce the city's budget through optimization of the current bicycle management system. It is expected to save the city's budget by decreasing carbon dioxide emissions and alleviating traffic congestion and establishing a healthy bicycle-loving culture in New York City.

How long will this take? How much will this cost?

There are multiple data sources in different locations to get expected insights. We also need to get permission to access each database site among New York City agencies. Hence, to build the Data Warehouse for this project, we need to have sufficient storage, high-speed processors such as GPU and CPU, internet connection, and a team of three data engineers and two data analysts for 3 months. The approximate cost for this project is in the range of $80,000 to $100,000/year to maintain and update.

Technical Justification:

Which data sources do we already have for this project?

Dataset 1: Bicycle count conducted around New York City at key locations.

Dataset 2: New York State census data

What new data sources do we need (if any)?

Dataset 1: Address information per each location

Is the data we have conformed, consistent, and current? (data quality)

The data is neither consistent nor cleaned along with dates. Some data has null, zero, or duplicated values that might require secondary verification processes from the other sources of data.

What technical skills will we need to complete this project?

1. Data Gathering
2. Data Exploration
3. Data Cleaning
4. Data Verification
5. Data Visualization
6. Data Modeling
7. ETL Creation
8. BI Application Design and Implementation
9. Data Warehouse engineering
10. Standardized Report development

Will we need any new types of technologies?

1. Data Mining
2. ETL tool or custom development
3. Cloud data storage
4. Cloud data warehouse

Key Performance Indicators (KPI's) your Data Warehouse will display:

1. Bicycle count at key locations
2. Bicycle service information – zip code, latitude, longitude, and address
3. Bicycle per Person Ratio

Data Warehouse Project

March 3, 2022

1 Data Warehouse Project - Kwang Heum Yeon

To find the feasibility of this project, I tried to gather some of the focal data addressed on the previous pages and executed preliminary data processing by using Python coding language.

1.1 Bicycle Count per Location by latitude and longitude

[1]:
```
import pandas as pd
import numpy as np
```

https://data.cityofnewyork.us/Transportation/Bicycle-Counts/uczf-rk3c

[2]:
```
df1 = pd.read_csv('#1_Bicycle_Counts.csv')
df1.head()
```

[2]:

	id	counts	date	status	site
0	0	41.0	08/31/2012 12:00:00 AM	4.0	100005020
1	1	52.0	08/31/2012 12:15:00 AM	4.0	100005020
2	2	38.0	08/31/2012 12:30:00 AM	4.0	100005020
3	3	36.0	08/31/2012 12:45:00 AM	4.0	100005020
4	4	40.0	08/31/2012 01:00:00 AM	4.0	100005020

[3]:
```
df1[df1.site == 100005020].counts.sum()
```

[3]:
```
416543.0
```

[4]:
```
df1 = df1.groupby('site').sum().reset_index()[['site','counts']]
df1.head()
```

[4]:

	site	counts
0	100005020	416543.0
1	100009424	2988769.0
2	100009425	4470824.0
3	100009426	468686.0
4	100009427	14326653.0

[5]:
```
drp = df1[df1.counts == 0].index.values
df1 = df1.drop(index = drp)
df1.head()
```

[5]:
```
        site      counts
0  100005020    416543.0
1  100009424   2988769.0
2  100009425   4470824.0
3  100009426    468686.0
4  100009427  14326653.0
```

[6]:
```
df1.shape
```

[6]:
```
(24, 2)
```

https://data.cityofnewyork.us/Transportation/Bicycle-Counters/smn3-rzf9

[7]:
```
df2 = pd.read_csv('#2_Bicycle_Counters.csv')
df2.head()
```

[7]:
```
   id                                          name   latitude  longitude  \
0   0        Manhattan Bridge 2012 Test Bike Counter  40.699810 -73.985890
1   5          Ed Koch Queensboro Bridge Shared Path  40.751038 -73.940820
2  10  1st Avenue - 26th St N - Interference testing  40.738830 -73.977165
3  24                                          Test  40.707381 -73.998845
4  25          Comprehensive Brooklyn Bridge Counter  40.711644 -74.004109

              domain       site                    timezone  interval  \
0  New York City DOT  100005020  (UTC-05:00) US/Eastern;DST        15
1  New York City DOT  100009428  (UTC-05:00) US/Eastern;DST        15
2  New York City DOT  100010020  (UTC-05:00) US/Eastern;DST        15
3  New York City DOT  300020692  (UTC-05:00) US/Eastern;DST         0
4  New York City DOT  300020904  (UTC-05:00) US/Eastern;DST        15

       counter
0          NaN
1  Y2H19111445
2  Y2H18044984
3          NaN
4          NaN
```

[8]:
```
df12 = pd.merge(df1, df2, how = 'left', on = 'site')
df12 = df12[['name','site', 'counts', 'latitude', 'longitude']]
df12.head()
```

[8]:
```
                                   name       site      counts   latitude  \
0  Manhattan Bridge 2012 Test Bike Counter  100005020    416543.0  40.699810
1                   2nd Avenue - 26th St S  100009424   2988769.0  40.739710
2                       Prospect Park West  100009425   4470824.0  40.671288
3                Manhattan Bridge Ped Path  100009426    468686.0  40.714573
4            Williamsburg Bridge Bike Path  100009427  14326653.0  40.710530
```

```
   longitude
0 -73.985890
1 -73.979540
2 -73.971382
3 -73.994950
4 -73.961450
```

[9]:
```
rnd = []
for i in df12.latitude.values:
    rnd.append(np.round(i, 2))
df12.latitude = rnd

rnd = []
for i in df12.longitude.values:
    rnd.append(np.round(i, 2))
df12.longitude = rnd
```

[10]:
```
df12.head()
```

[10]:
```
                                      name       site      counts  latitude  \
0  Manhattan Bridge 2012 Test Bike Counter  100005020    416543.0     40.70
1                   2nd Avenue - 26th St S  100009424   2988769.0     40.74
2                       Prospect Park West  100009425   4470824.0     40.67
3                Manhattan Bridge Ped Path  100009426    468686.0     40.71
4            Williamsburg Bridge Bike Path  100009427  14326653.0     40.71

   longitude
0     -73.99
1     -73.98
2     -73.97
3     -73.99
4     -73.96
```

[11]:
```
df12.shape
```

[11]:
```
(24, 5)
```

1.2 Bicycle Count per Zip code

https://www.unitedstateszipcodes.org/ny/

[12]:
```
df3 = pd.read_csv('#3_zip_code_database.csv')
df3.head()
```

[12]:
```
     zip    type  decommissioned primary_city acceptable_cities  \
0  99723  PO BOX               0       Barrow               NaN
1  99782  PO BOX               0   Wainwright               NaN
2  99791  PO BOX               0      Atqasuk            Barrow
```

```
3  99734  PO BOX              0  Prudhoe Bay               NaN
4  99747  PO BOX              0     Kaktovik               NaN

  unacceptable_cities state               county           timezone  \
0                 NaN    AK  North Slope Borough  America/Anchorage
1                 NaN    AK  North Slope Borough  America/Anchorage
2                 NaN    AK  North Slope Borough  America/Anchorage
3                 NaN    AK  North Slope Borough  America/Anchorage
4                 NaN    AK  North Slope Borough  America/Anchorage

  area_codes world_region country  latitude  longitude  \
0        907          NaN      US     71.28    -156.78
1        907          NaN      US     70.63    -159.96
2        907          NaN      US     70.48    -157.39
3        907          NaN      US     70.43    -149.29
4        907          NaN      US     70.12    -143.66

   irs_estimated_population
0                      3630
1                       435
2                       214
3                        96
4                       281
```

[13]:
```
df3 = df3[['latitude', 'longitude', 'zip']]
df3.head()
```

[13]:
```
   latitude  longitude    zip
0     71.28    -156.78  99723
1     70.63    -159.96  99782
2     70.48    -157.39  99791
3     70.43    -149.29  99734
4     70.12    -143.66  99747
```

[14]:
```
df3 = df3.groupby(['latitude', 'longitude']).median().reset_index()
df3.head()
```

[14]:
```
   latitude  longitude      zip
0    -44.25      33.53   9323.0
1    -14.27    -170.70  96799.0
2      0.00       0.00   9743.5
3      5.29     162.97  96944.0
4      6.85     158.26  96941.0
```

[19]:
```
df123 = pd.merge(df12, df3, how = 'left', on = ['latitude','longitude'])
df123
```

[19]:

	name	site	counts
0	Manhattan Bridge 2012 Test Bike Counter	100005020	416543.0
1	2nd Avenue - 26th St S	100009424	2988769.0
2	Prospect Park West	100009425	4470824.0
3	Manhattan Bridge Ped Path	100009426	468686.0
4	Williamsburg Bridge Bike Path	100009427	14326653.0
5	Ed Koch Queensboro Bridge Shared Path	100009428	10862827.0
6	Manhattan Bridge 2013 to 2018 Bike Counter	100009429	6394256.0
7	Staten Island Ferry	100010017	825979.0
8	Pulaski Bridge	100010018	3107263.0
9	Kent Ave btw North 8th St and North 9th St	100010019	4964859.0
10	1st Avenue - 26th St N - Interference testing	100010020	8193890.0
11	Brooklyn Bridge Bike Path	100010022	5114851.0
12	Forsyth Plaza	100039064	18764.0
13	Manhattan Bridge Display Bike Counter	100047029	11791947.0
14	Manhattan Bridge 2012 to 2019 Bike Counter	100051865	7645129.0
15	Manhattan Bridge Interference Calibration 2019…	100055175	918672.0
16	8th Ave at 50th St.	100057316	2926986.0
17	Broadway at 50th St	100057318	158147.0
18	Amsterdam Ave at 86th St.	100057319	2107765.0
19	Columbus Ave at 86th St.	100057320	1151883.0
20	Kent Ave btw South 6th St. and Broadway	100058279	1735733.0
21	Manhattan Bridge Bike Comprehensive	100062893	11791947.0
22	Brooklyn Bridge Bicycle Path (Roadway)	300020241	325611.0
23	Comprehensive Brooklyn Bridge Counter	300020904	5435241.0

	latitude	longitude	zip
0	40.70	-73.99	NaN
1	40.74	-73.98	10010.0
2	40.67	-73.97	NaN
3	40.71	-73.99	10155.0
4	40.71	-73.96	NaN
5	40.75	-73.94	NaN
6	40.70	-73.99	NaN
7	40.64	-74.07	NaN
8	40.74	-73.95	NaN
9	40.72	-73.96	NaN
10	40.74	-73.98	10010.0
11	40.71	-74.00	10096.0
12	40.72	-73.99	NaN
13	40.72	-73.99	NaN
14	40.70	-73.99	NaN
15	0.00	0.00	9743.5
16	40.76	-73.99	10027.0
17	40.76	-73.98	10111.5
18	40.79	-73.98	NaN
19	40.79	-73.98	NaN

```
20     40.71     -73.97      NaN
21     40.72     -73.99      NaN
22     40.71     -74.00  10096.0
23     40.71     -74.00  10096.0
```

Manually verify the zip code by lookup data, and then combine it with df4.

[16]:
```
df123.counts.sum() == df1.counts.sum()
```

[16]:
```
True
```

1.3 Population per Zip code

https://www.newyork-demographics.com/zip_codes_by_population

[17]:
```
df4 = pd.read_csv('#4_zip per population(option).csv')
df4.head()
```

[17]:
```
   Rank Zip Code Population
0     1    11368    112,088
1     2    11385    107,796
2     3    11211    103,123
3     4    11208    101,313
4     5    10467    101,255
```

[18]:
```
sum(df4['Zip Code'].value_counts() == 1)
```

[18]:
```
1584
```

TBD by SQL

1.4 Bicycle Count per Person

TBD by SQL