Assignment 2

February 22, 2020

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

```
[2]: df = pd.read_csv('vgsales2019.csv', header= 0)
```

```
[3]: df.head()
```

```
[3]:     ID                          Name     Genre PLATFORM         PUBLISHER  \
     0  ga1                    Wii Sports    Sports      Wii          Nintendo
     1  ga2             Super Mario Bros.  Platform      NES          Nintendo
     2  ga3                Mario Kart Wii    Racing      Wii          Nintendo
     3  ga4  PlayerUnknown's Battlegrounds   Shooter       PC  PUBG Corporation
     4  ga5             Wii Sports Resort    Sports      Wii          Nintendo

               DEVELOPER  CRITIC_SCORE  USER_SCORE  Total_TOTAL_SHIPPED  \
     0      Nintendo EAD           7.7         NaN                82.86
     1      Nintendo EAD          10.0         NaN                40.24
     2      Nintendo EAD           8.2         9.1                37.14
     3  PUBG Corporation           NaN         NaN                36.60
     4      Nintendo EAD           8.0         8.8                33.09

        GLOBAL_SALES  NA_SALES  PAL_SALES  JP_SALES  OTHER_SALES     YRS
     0           NaN       NaN        NaN       NaN          NaN  2006.0
     1           NaN       NaN        NaN       NaN          NaN  1985.0
     2           NaN       NaN        NaN       NaN          NaN  2008.0
     3           NaN       NaN        NaN       NaN          NaN  2017.0
     4           NaN       NaN        NaN       NaN          NaN  2009.0
```

0.0.1 We can see from the dataset that there is a lot of missing values in SALES columns. Since this will affect the quality of the tables, we will be dropping these tables.

```
[4]: df = df.
      ↪drop(['GLOBAL_SALES','NA_SALES','PAL_SALES','JP_SALES','OTHER_SALES'],axis =␣
      ↪1)
```

0.0.2 To increase the quality of the data, we will be dropping the records having any null values

[5]:
```
df = df.dropna()
df
```

[5]:
```
            ID                       Name         Genre PLATFORM  \
2          ga3             Mario Kart Wii        Racing      Wii
4          ga5          Wii Sports Resort        Sports      Wii
6          ga7      New Super Mario Bros.      Platform       DS
8          ga9  New Super Mario Bros. Wii      Platform      Wii
11        ga12                   Wii Play          Misc      Wii
…        …                        …           …      …
2546    ga2547                  Red Steel       Shooter      Wii
3633    ga3634     Phantasy Star Portable  Role-Playing      PSP
5781    ga5782            Bionic Commando     Adventure     X360
7278    ga7279                The Conduit       Shooter      Wii
16763  ga16764           Submarine Titans      Strategy       PC

            PUBLISHER              DEVELOPER  CRITIC_SCORE  USER_SCORE  \
2            Nintendo           Nintendo EAD           8.2         9.1
4            Nintendo           Nintendo EAD           8.0         8.8
6            Nintendo           Nintendo EAD           9.1         8.1
8            Nintendo           Nintendo EAD           8.6         9.2
11           Nintendo           Nintendo EAD           5.9         4.5
…               …                    …           …         …
2546          Ubisoft          Ubisoft Paris           5.9         7.8
3633             Sega            Alfa System           7.1         9.1
5781           Capcom                   GRIN           7.1         8.0
7278             Sega  High Voltage Software           6.6         8.2
16763  Strategy First        Ellipse Studios           6.1         9.0

       Total_TOTAL_SHIPPED     YRS
2                    37.14  2008.0
4                    33.09  2009.0
6                    30.80  2006.0
8                    30.22  2009.0
11                   28.02  2007.0
…                    …     …
2546                  0.95  2006.0
3633                  0.64  2009.0
5781                  0.38  2009.0
7278                  0.27  2009.0
16763                 0.03  2000.0

[64 rows x 10 columns]
```

0.0.3 Creating the tables as DataFrames required

[6]:
```
game_inventory = pd.DataFrame(columns = ['G_ID', 'G_NAME','GENRE','DEVELOPER',␣
 ↪'PLATFORM','TOTAL_SHIPPED','CRITIC_SCORE'
                                          ,'USER_SCORE','YEAR'])
```

0.0.4 Storing the appropriate values to proposed Tables

[7]:
```
game_inventory['G_NAME'] = df['Name']
game_inventory['PLATFORM'] = df['PLATFORM']
years = []
for i in df['YRS']:
    years.append(int(i)) # Converting to integer
game_inventory['YEAR'] = years
game_inventory['TOTAL_SHIPPED'] = df['Total_TOTAL_SHIPPED']
game_inventory['CRITIC_SCORE'] = df['CRITIC_SCORE']
game_inventory['USER_SCORE'] = df['USER_SCORE']
game_inventory['GENRE'] = df['Genre']
game_inventory['DEVELOPER'] = df['DEVELOPER']
```

1 1 NF Normalization:

2 Now we shall try to achive 1NF from on the Game Inventory by adding Primary Key. The same shall be done for all the tables to achieve the 1NF across the tables in the database that we are building.

2.1 Generating Primary Keys for Game_Inventory

[8]:
```
j=1
g_id = []
for i in game_inventory['G_ID']:
    g_id.append(j)
    j+=1
game_inventory['G_ID']=g_id
```

[9]:
```
game_inventory['G_ID']=g_id
```

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

[10]:
```
    G_ID                       G_NAME     GENRE     DEVELOPER  \
2      1                Mario Kart Wii    Racing  Nintendo EAD
4      2             Wii Sports Resort    Sports  Nintendo EAD
6      3         New Super Mario Bros.  Platform  Nintendo EAD
8      4     New Super Mario Bros. Wii  Platform  Nintendo EAD
```

```
11     5                       Wii Play          Misc    Nintendo EAD
14     6                  Mario Kart DS        Racing    Nintendo EAD
28     7                    Pokemon X/Y  Role-Playing      Game Freak
33     8  Pokemon Black / White Version  Role-Playing      Game Freak
43     9                         Halo 3       Shooter  Bungie Studios
52    10        Super Smash Bros. Brawl      Fighting    Project Sora

   PLATFORM  TOTAL_SHIPPED  CRITIC_SCORE  USER_SCORE  YEAR
2       Wii          37.14           8.2         9.1  2008
4       Wii          33.09           8.0         8.8  2009
6        DS          30.80           9.1         8.1  2006
8       Wii          30.22           8.6         9.2  2009
11      Wii          28.02           5.9         4.5  2007
14       DS          23.60           9.1         9.4  2005
28      3DS          16.37           8.9         9.7  2013
33       DS          15.64           8.6         9.0  2011
43     X360          14.50           9.6         9.5  2007
52      Wii          13.29           9.2         9.7  2008
```

2.2 Creating separate tables for Developers, Publishers and Platforms

[17]:

```
developer = pd.DataFrame(columns = ['DEV_ID', 'DEV_NAME', 'PUB_ID'])
publisher = pd.DataFrame(columns = ['PUB_ID','PUB_NAME'])
platform = pd.DataFrame(columns = ['PLAT_ID','PLTF_NAME'])
genre = pd.DataFrame(columns = ['GENRE_ID','GEN_NAME'])
developer['DEV_NAME'] = df.DEVELOPER.unique()
publisher['PUB_NAME'] = df.PUBLISHER.unique()
platform.PLTF_NAME = game_inventory['PLATFORM'].unique()
genre.GEN_NAME = game_inventory['GENRE'].unique()
```

3 2 Normalization:

4 Since there is no calculated columns in the tables and no partial dependencies we have achieved the 2NF.

5 3 Normalization:

6 To achieve the 3NF each column in the table should have direct relation to the PK. We have ID columns of other tables which are kept to establish a relationship to another table by adding the foreign key constraints to the table.

6.1 Generating Foreign Keys for Developers, Publishers and Platforms

[18]:

```
game_inventory.rename(columns = {'DEVELOPER':'DEV_ID'}, inplace = True) # Renaming to reflect the meaningful changes
j = 101
temp_pubid = []
for i in publisher['PUB_NAME']:
    temp_pubid.append(j)
    j+=1
j= 201
temp_devid = []
for i in developer['DEV_NAME']:
    temp_devid.append(j)
    j+=1
j= 301
temp_platid = []
for i in platform['PLTF_NAME']:
    temp_platid.append(j)
    j+=1
j=401
temp_genid=[]
for i in genre['GEN_NAME']:
    temp_genid.append(j)
    j+=1
publisher['PUB_ID'] = temp_pubid
developer['DEV_ID'] = temp_devid
platform.PLAT_ID = temp_platid
genre.GENRE_ID = temp_genid
```

[19]:

```
publisher.head(10)
```

[19]:

```
   PUB_ID                      PUB_NAME
0     101                      Nintendo
1     102        Microsoft Game Studios
```

```
2     103  Sony Interactive Entertainment
3     104     Sony Computer Entertainment
4     105                          Square
5     106                     Square Enix
6     107          Blizzard Entertainment
7     108                         Capcom
8     109                 Electronic Arts
9     110                            Sega
```

[20]:

```
genre.head(10)
```

[20]:

```
   GENRE_ID          GEN_NAME
0       401            Racing
1       402            Sports
2       403          Platform
3       404              Misc
4       405      Role-Playing
5       406           Shooter
6       407          Fighting
7       408  Action-Adventure
8       409            Action
9       410         Adventure
```

6.1.1 Now, we will be adding the corresponding PUB_ID from the Publisher table to the Developer Table

[21]:

```
df2 = df.drop_duplicates('DEVELOPER', keep='first', inplace = False)
pubtemp = []
for i in developer['DEV_NAME']:
    value = df2.loc[(df2.DEVELOPER == i),'PUBLISHER'].tolist()[0]
    pubtemp.append(value)
pub_id = []
for i in pubtemp:
    pub_id.append(publisher.loc[publisher.PUB_NAME == i,'PUB_ID'].tolist()[0])
developer.PUB_ID = pub_id
```

6.1.2 Now, we will be adding the corresponding PLAT_ID and DEV_ID from the Platform table and Developer Table to the Game_Inventory Table

[22]:

```
df2 = df.drop_duplicates('DEVELOPER', keep='first', inplace = False)
pubtemp = []
for i in game_inventory['PLATFORM']:
    value = df.loc[(game_inventory.PLATFORM == i),'PLATFORM'].tolist()[0]
    pubtemp.append(value)
pub_id = []
for i in pubtemp:
    pub_id.append(platform.loc[platform.PLTF_NAME == i,'PLAT_ID'].tolist()[0])
game_inventory['PLATFORM']=pub_id
```

[23]:

```
devtemp = []
for i in df['Name']:
    value = df.loc[(df.Name == i),'DEVELOPER'].tolist()[0]
    devtemp.append(value)
dev_id = []
for i in devtemp:
    dev_id.append(developer.loc[developer.DEV_NAME == i,'DEV_ID'].tolist()[0])
game_inventory['DEV_ID'] = dev_id
```

[24]:

```
genret = []
for i in df['Genre']:
    value = df.loc[(df.Genre == i),'Genre'].tolist()[0]
    genret.append(value)
gen_id = []
for i in genret:
    gen_id.append(genre.loc[genre.GEN_NAME == i,'GENRE_ID'].tolist()[0])
game_inventory['GENRE'] = gen_id
game_inventory.rename(columns = {'GENRE':'GENRE_ID'}, inplace = True)
```

6.2 Sample Table Representation

[25]:

```
game_inventory.head(10)
```

[25]:

```
    G_ID                         G_NAME  GENRE_ID  DEV_ID  PLATFORM  \
2      1                 Mario Kart Wii       401     201       301
4      2              Wii Sports Resort       402     201       301
6      3          New Super Mario Bros.       403     201       302
8      4      New Super Mario Bros. Wii       403     201       301
11     5                       Wii Play       404     201       301
14     6                  Mario Kart DS       401     201       302
28     7                    Pokemon X/Y       405     202       303
33     8  Pokemon Black / White Version       405     202       302
43     9                         Halo 3       406     203       304
52    10        Super Smash Bros. Brawl       407     204       301

    TOTAL_SHIPPED  CRITIC_SCORE  USER_SCORE  YEAR
2           37.14           8.2         9.1  2008
4           33.09           8.0         8.8  2009
6           30.80           9.1         8.1  2006
8           30.22           8.6         9.2  2009
11          28.02           5.9         4.5  2007
14          23.60           9.1         9.4  2005
28          16.37           8.9         9.7  2013
33          15.64           8.6         9.0  2011
```

```
      43          14.50           9.6         9.5  2007
      52          13.29           9.2         9.7  2008
```

```
[26]: platform.head(10)
```

```
[26]:    PLAT_ID PLTF_NAME
      0      301       Wii
      1      302        DS
      2      303       3DS
      3      304      X360
      4      305        NS
      5      306       PS4
      6      307        PS
      7      308       PS2
      8      309       N64
      9      310       PS3
```

```
[27]: developer.head(10)
```

```
[27]:    DEV_ID            DEV_NAME  PUB_ID
      0     201        Nintendo EAD     101
      1     202          Game Freak     101
      2     203      Bungie Studios     102
      3     204        Project Sora     101
      4     205  Nintendo EAD Tokyo     101
      5     206  Bandai Namco Games     101
      6     207        Nintendo EPD     101
      7     208     Guerrilla Games     103
      8     209          SquareSoft     104
      9     210         Square Enix     106
```

```
[28]: publisher.head(10)
```

```
[28]:    PUB_ID                        PUB_NAME
      0     101                        Nintendo
      1     102          Microsoft Game Studios
      2     103  Sony Interactive Entertainment
      3     104     Sony Computer Entertainment
      4     105                          Square
      5     106                     Square Enix
      6     107           Blizzard Entertainment
      7     108                          Capcom
      8     109                 Electronic Arts
      9     110                            Sega
```

```
[29]: genre.head(10)
```

```
[29]:    GENRE_ID          GEN_NAME
      0       401            Racing
      1       402            Sports
      2       403          Platform
      3       404              Misc
      4       405      Role-Playing
      5       406           Shooter
      6       407          Fighting
      7       408  Action-Adventure
      8       409            Action
      9       410         Adventure
```

6.3 Social Media Account

```
[ ]: import twitter
```

```
[ ]: CONSUMER_KEY = 'Ulg0aOEDEy09m90eQ1WjgY8ax'
     CONSUMER_SECRET = ''
     OAUTH_TOKEN = '715063474020159488-TWPJ4GzyfYJ94Iy4ItvIu4c7Qd065nD'
     OAUTH_TOKEN_SECRET = ''
     auth = twitter.oauth.OAuth(OAUTH_TOKEN, OAUTH_TOKEN_SECRET,
                                CONSUMER_KEY, CONSUMER_SECRET)

     twitter_api = twitter.Twitter(auth=auth)
     # For security purposes we won't be sharing the secret ID and password.
```

6.3.1 CONTRIBUTION

Your contribution towards project. How much code did you write and how much you took from other site or some other source.

Our Own : 50%

By External source: 50%