**Comments.py**

import pandas as pd

import numpy as np

import warnings

warnings.filterwarnings("ignore")

def read\_data\_from\_csv():

comments=pd.read\_csv('comments.csv')

return comments

def data\_cleaning():

#DO NOT REMOVE FOLLOWING LINE

#call remove\_unwanted\_columns() function to get dataframe

comments=read\_data\_from\_csv()

#Remove Unwanted columns

comments.drop(["posted date","emoji used","Hashtags used count"],inplace=True,axis=1)

#rename columns, only these columns are allowed in the dataset

# 1. id

# 2. comment\_text

# 3. user\_id

# 4. photo\_id

# 5. created\_at

comments.rename(columns={'comment':'comment\_text' ,'User id':'user\_id',

'Photo id':'photo\_id','created Timestamp':'created\_at' },inplace=True)

#export cleaned Dataset to newcsv file named "comments\_cleaned.csv"

comments.to\_csv('comments\_cleaned.csv')

return comments

#Do not Delete the Following function

def task\_runner():

data\_cleaning()

task\_runner()

**follows.py**

import pandas as pd

import numpy as np

import warnings

warnings.filterwarnings("ignore")

def read\_data\_from\_csv():

follows=pd.read\_csv('follows.csv')

return follows

def data\_cleaning():

#DO NOT REMOVE FOLLOWING LINE

#call remove\_unwanted\_columns() function to get dataframe

follows=read\_data\_from\_csv()

#Remove Unwanted columns

follows.drop(columns=["is follower active","followee Acc status"],axis=1,inplace=True)

#rename columns, only these columns are allowed in the dataset

# 1. follower\_id

# 2. followee\_id

# 3. created\_at

follows.rename(columns={"follower":"follower\_id","followee ":"followee\_id","created time":"created\_at"},inplace=True)

#export cleaned Dataset to newcsv file named "follows\_cleaned.csv"

follows.to\_csv('follows\_cleaned.csv')

return follows

#Do not Delete the Following function

def task\_runner():

data\_cleaning()

task\_runner()

**likes.py**

import pandas as pd

import numpy as np

import warnings

warnings.filterwarnings("ignore")

def read\_data\_from\_csv():

likes=pd.read\_csv('likes.csv')

return likes

def data\_cleaning():

#DO NOT REMOVE FOLLOWING LINE

#call remove\_unwanted\_columns() function to get dataframe

likes=read\_data\_from\_csv()

#Remove Unwanted columns

likes.drop(columns=["following or not","like type"],axis=1,inplace=True)

#rename columns, only these columns are allowed in the dataset

# 1. user\_id

# 2. photo\_id

# 3. created\_at

likes.rename(columns={"user ":"user\_id","photo":"photo\_id","created time":"created\_at"},inplace=True)

#export cleaned Dataset to newcsv file named "likes\_cleaned.csv"

likes.to\_csv('likes\_cleaned.csv')

return likes

#Do not Delete the Following function

def task\_runner():

data\_cleaning()

task\_runner()

**photos\_tags.py**

import pandas as pd

import numpy as np

import warnings

warnings.filterwarnings("ignore")

def read\_data\_from\_csv():

photo\_tags=pd.read\_csv('photo\_tags.csv')

return photo\_tags

def data\_cleaning():

#DO NOT REMOVE FOLLOWING LINE

#call remove\_unwanted\_columns() function to get dataframe

photo\_tags=read\_data\_from\_csv()

#Remove Unwanted columns

photo\_tags.drop(columns=["user id"],axis=1,inplace=True)

photo\_tags.rename(columns={"photo":"photo\_id","tag ID":"tag\_id"},inplace=True)

#rename columns, only these columns are allowed in the dataset

# 1. photo\_id

# 2. tag\_id

#photo\_tags.rename(columns={"photo":"photo\_id","tag ID":"tag\_id"},inplace=True)

#export cleaned Dataset to newcsv file named "photo\_tags\_cleaned.csv"

photo\_tags.to\_csv('photo\_tags\_cleaned.csv')

return photo\_tags

#Do not Delete the Following function

def task\_runner():

data\_cleaning()

task\_runner()

**photos.py**

import pandas as pd

import numpy as np

import warnings

warnings.filterwarnings("ignore")

def read\_data\_from\_csv():

photos=pd.read\_csv('photos.csv')

return photos

def data\_cleaning():

#DO NOT REMOVE FOLLOWING LINE

#call remove\_unwanted\_columns() function to get dataframe

photos=read\_data\_from\_csv()

#Remove Unwanted columns

#photos.drop(columns=["Insta filter used","photo type"],axis=1,inplace=True)

#photos.rename(columns={"image link":"image\_url","user ID":"user\_id","created dat":"created\_date"},inplace=True)

photos.drop(columns=["Insta filter used","photo type"],axis=1,inplace=True)

photos.rename(columns={"image link":"image\_url","user ID":"user\_id","created dat":"created\_date"},inplace=True)

#rename columns, only these columns are allowed in the dataset

# 1. id

# 2. image\_url

# 3. user\_id

# 4. created\_date

#export cleaned Dataset to newcsv file named "photos\_cleaned.csv"

photos.to\_csv('photos\_cleaned.csv')

return photos

#Do not Delete the Following function

def task\_runner():

data\_cleaning()

task\_runner()

**tags.py**

import pandas as pd

import numpy as np

import warnings

warnings.filterwarnings("ignore")

def read\_data\_from\_csv():

tags=pd.read\_csv('tags.csv')

return tags

def data\_cleaning():

#DO NOT REMOVE FOLLOWING LINE

#call remove\_unwanted\_columns() function to get dataframe

tags=read\_data\_from\_csv()

#Remove Unwanted columns

tags.drop(columns=["location"],axis=1,inplace=True)

tags.rename(columns={"tag text":"tag\_name","created time":"created\_at"},inplace=True)

#rename columns, only these columns are allowed in the dataset

# 1. id

# 2. tag\_name

# 3. created\_at

#export cleaned Dataset to newcsv file named "tags\_cleaned.csv"

tags.to\_csv('tags\_cleaned.csv')

return tags

#Do not Delete the Following function

def task\_runner():

data\_cleaning()

task\_runner()

**user.py**

import pandas as pd

import numpy as np

import warnings

warnings.filterwarnings("ignore")

def read\_data\_from\_csv():

users=pd.read\_csv('users.csv')

return users

def data\_cleaning():

#DO NOT REMOVE FOLLOWING LINE

#call remove\_unwanted\_columns() function to get dataframe

users=read\_data\_from\_csv()

#Remove Unwanted columns

users.drop(columns=["private/public","post count","Verified status"],axis=1,inplace=True)

users.rename(columns={"name":"username","created time":"created\_at"},inplace=True)

#rename columns, only these columns are allowed in the dataset

# 1. id

# 2. username

# 3. created\_at

#export cleaned Dataset to newcsv file named "users\_cleaned.csv"

users.to\_csv('users\_cleaned.csv')

return users

#Do not Delete the Following function

def task\_runner():

data\_cleaning()

task\_runner()