In [6]:
```
pip install pymysql
```

```
Collecting pymysqlNote: you may need to restart the kernel to use updated packa
ges.

  Downloading PyMySQL-1.0.2-py3-none-any.whl (43 kB)
     ---------------------------------------- 43.8/43.8 kB 143.5 kB/s eta 0:00:00
Installing collected packages: pymysql
Successfully installed pymysql-1.0.2
```

In [ ]:
```
Project - Instagram User Analytics
```

In [ ]:
```

```

In [1]:
```
import pymysql
import pandas as pd
```

In [2]:
```
db_name="ig_clone"
db_host="localhost"
db_username="root"
db_passwor="***"
```

In [3]:
```
try:
    conn=pymysql.connect(host=db_host,
                         port=int(3306),
                         user="root",
                         passwd=db_passwor,
                         db=db_name)
except e:
  print(E)
```

In [4]:
```
if conn:

    df_users = pd.read_sql_query("SELECT * from users ",conn)
    print(df_users)
    df_photos = pd.read_sql_query("SELECT * from photos",conn)
    print(df_photos)
    df_comments = pd.read_sql_query("SELECT * from comments",conn)
    print(df_comments)
    df_follows = pd.read_sql_query("SELECT * from follows",conn)
    print(df_follows)
    df_likes = pd.read_sql_query("SELECT * from likes",conn)
    print(df_likes)
    df_phototags = pd.read_sql_query("SELECT * from photo_tags",conn)
    print(df_tags)
    df_tags = pd.read_sql_query("SELECT * from tags",conn)
    print(df_tags)
    print("Database Table connected successfully")
    df_users.to_csv(r'C:\Users\santhosh\Music\seleniumproject\users.csv')
    df_photos.to_csv(r'C:\Users\santhosh\Music\seleniumproject\photos.csv')
    df_comments.to_csv(r'C:\Users\santhosh\Music\seleniumproject\comments.csv')
    df_follows.to_csv(r'C:\Users\santhosh\Music\seleniumproject\follows.csv')
    df_likes.to_csv(r'C:\Users\santhosh\Music\seleniumproject\likes.csv')
    df_phototags.to_csv(r'C:\Users\santhosh\Music\seleniumproject\photo_tags.csv'
    df_tags.to_csv(r'C:\Users\santhosh\Music\seleniumproject\tags.csv')

else :
    print("error")
```

```
C:\Users\santhosh\anaconda3\lib\site-packages\pandas\io\sql.py:762: UserWarni
ng: pandas only support SQLAlchemy connectable(engine/connection) ordatabase
string URI or sqlite3 DBAPI2 connectionother DBAPI2 objects are not tested, p
lease consider using SQLAlchemy
  warnings.warn(
```

In [ ]:

In [5]:
```
df_users = pd.read_sql_query("SELECT * from users ",conn)
print(df_users)
```

```
      id             username          created_at
0      1         Kenton_Kirlin 2017-02-16 18:22:11
1      2         Andre_Purdy85 2017-04-02 17:11:21
2      3         Harley_Lind18 2017-02-21 11:12:33
3      4         Arely_Bogan63 2016-08-13 01:28:43
4      5         Aniya_Hackett 2016-12-07 01:04:39
..   ...                   ...                 ...
195  196  Keenan.Schamberger60 2016-08-28 14:57:28
196  197        Tomas.Beatty93 2017-02-11 11:38:55
197  198       Imani_Nicolas17 2017-01-31 22:59:34
198  199          Alek_Watsica 2016-12-10 07:43:58
199  200             Javonte83 2017-03-27 22:06:37

[200 rows x 3 columns]


C:\Users\santhosh\anaconda3\lib\site-packages\pandas\io\sql.py:762: UserWarnin
g: pandas only support SQLAlchemy connectable(engine/connection) ordatabase str
ing URI or sqlite3 DBAPI2 connectionother DBAPI2 objects are not tested, please
consider using SQLAlchemy
  warnings.warn(
```

In [ ]:
```
users=df_users.sort_values(['created_at','username'],ascending=True)
```

1) Rewarding Most Loyal Users: People who have been using the platform for the longest time.
Your Task: Find the 5 oldest users of the Instagram from the database provided

In [7]:
```
top5=users.drop_duplicates(subset=["username"])
top5['username'].head(5)
```

Out[7]:
```
79        Darby_Herzog
66    Emilio_Bernier52
62            Elenor88
94            Nicole71
37    Jordyn.Jacobson2
Name: username, dtype: object
```

In [ ]:
```

```

In [8]:
```
df_photos = pd.read_sql_query("SELECT * from photos",conn)
print(df_photos)
```

```
      id             image_url  user_id         created_dat
0      1     http://elijah.biz (http://elijah.biz)        1 2022-10-29 12:18:20
1      2    https://shanon.org (https://shanon.org)        1 2022-10-29 12:18:2
0
2      3      http://vicky.biz (http://vicky.biz)        1 2022-10-29 12:18:20
3      4      http://oleta.net (http://oleta.net)        1 2022-10-29 12:18:20
4      5  https://jennings.biz (https://jennings.biz)        1 2022-10-29 12:1
8:20
..   ...                   ...      ...                 ...
509  510   http://ryleigh.info (http://ryleigh.info)       99 2022-10-29 12:33:
23
510  511   https://darien.name (https://darien.name)       99 2022-10-29 12:33:
23
511  512   https://xzavier.org (https://xzavier.org)       99 2022-10-29 12:33:
23
512  513    https://kaela.name (https://kaela.name)      100 2022-10-29 12:33:2
3
513  514   http://dedrick.info (http://dedrick.info)      100 2022-10-29 12:33:
23

[514 rows x 4 columns]


C:\Users\santhosh\anaconda3\lib\site-packages\pandas\io\sql.py:762: UserWarnin
g: pandas only support SQLAlchemy connectable(engine/connection) ordatabase str
ing URI or sqlite3 DBAPI2 connectionother DBAPI2 objects are not tested, please
consider using SQLAlchemy
  warnings.warn(
```

In [ ]:
```
```

In [9]:
```
id=df_photos.drop_duplicates(subset=["user_id"])
id
```

Out[9]:

	id	image_url	user_id	created_dat
0	1	http://elijah.biz	1	2022-10-29 12:18:20
5	6	https://quinn.biz	2	2022-10-29 12:18:20
9	10	https://elenor.name	3	2022-10-29 12:18:20
13	14	https://gerhard.biz	4	2022-10-29 12:18:20
16	17	http://annamae.name	6	2022-10-29 12:18:20
...	...	...	...	...
246	247	https://helmer.org	96	2022-10-29 12:18:20
249	250	http://ayla.org	97	2022-10-29 12:18:20
251	252	http://jennie.com	98	2022-10-29 12:18:20
252	253	http://ryleigh.info	99	2022-10-29 12:18:20
255	256	https://kaela.name	100	2022-10-29 12:18:20

74 rows × 4 columns

In [10]:
```
id.rename(columns={'id':'image_id','user_id':'id'},inplace=True)
id
```

```
C:\Users\santhosh\AppData\Local\Temp\ipykernel_2840\2669915829.py:1: SettingWit
hCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/sta
ble/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pandas.pyd
ata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-c
opy)
  id.rename(columns={'id':'image_id','user_id':'id'},inplace=True)
```

Out[10]:

	image_id	image_url	id	created_dat
0	1	http://elijah.biz	1	2022-10-29 12:18:20
5	6	https://quinn.biz	2	2022-10-29 12:18:20
9	10	https://elenor.name	3	2022-10-29 12:18:20
13	14	https://gerhard.biz	4	2022-10-29 12:18:20
16	17	http://annamae.name	6	2022-10-29 12:18:20
...	...	...	...	...
246	247	https://helmer.org	96	2022-10-29 12:18:20
249	250	http://ayla.org	97	2022-10-29 12:18:20
251	252	http://jennie.com	98	2022-10-29 12:18:20
252	253	http://ryleigh.info	99	2022-10-29 12:18:20
255	256	https://kaela.name	100	2022-10-29 12:18:20

74 rows × 4 columns

In [11]:
```
df_cd=pd.merge(df_users,id,how='left',on='id')
df_cd
```

Out[11]:

	id	username	created_at	image_id	image_url	created_dat
0	1	Kenton_Kirlin	2017-02-16 18:22:11	1.0	http://elijah.biz	2022-10-29 12:18:20
1	2	Andre_Purdy85	2017-04-02 17:11:21	6.0	https://quinn.biz	2022-10-29 12:18:20
2	3	Harley_Lind18	2017-02-21 11:12:33	10.0	https://elenor.name	2022-10-29 12:18:20
3	4	Arely_Bogan63	2016-08-13 01:28:43	14.0	https://gerhard.biz	2022-10-29 12:18:20
4	5	Aniya_Hackett	2016-12-07 01:04:39	NaN	NaN	NaT
...	...	...	...	...	...	...
195	196	Keenan.Schamberger60	2016-08-28 14:57:28	NaN	NaN	NaT
196	197	Tomas.Beatty93	2017-02-11 11:38:55	NaN	NaN	NaT
197	198	Imani_Nicolas17	2017-01-31 22:59:34	NaN	NaN	NaT
198	199	Alek_Watsica	2016-12-10 07:43:58	NaN	NaN	NaT
199	200	Javonte83	2017-03-27 22:06:37	NaN	NaN	NaT

200 rows × 6 columns

In [12]:
```
df_cd=df_cd[df_cd['image_url'].isna()]
df_cd
```

Out[12]:

	id	username	created_at	image_id	image_url	created_dat
4	5	Aniya_Hackett	2016-12-07 01:04:39	NaN	NaN	NaT
6	7	Kasandra_Homenick	2016-12-12 06:50:08	NaN	NaN	NaT
13	14	Jaclyn81	2017-02-06 23:29:16	NaN	NaN	NaT
20	21	Rocio33	2017-01-23 11:51:15	NaN	NaN	NaT
23	24	Maxwell.Halvorson	2017-04-18 02:32:44	NaN	NaN	NaT
...	...	...	...	...	...	...
195	196	Keenan.Schamberger60	2016-08-28 14:57:28	NaN	NaN	NaT
196	197	Tomas.Beatty93	2017-02-11 11:38:55	NaN	NaN	NaT
197	198	Imani_Nicolas17	2017-01-31 22:59:34	NaN	NaN	NaT
198	199	Alek_Watsica	2016-12-10 07:43:58	NaN	NaN	NaT
199	200	Javonte83	2017-03-27 22:06:37	NaN	NaN	NaT

126 rows × 6 columns

2) Remind Inactive Users to Start Posting: By sending them promotional emails to post their 1st photo. Your Task: Find the users who have never posted a single photo on Instagram

In [13]:
```
name_list=df_cd['username']
name_list
```

Out[13]:
```
4              Aniya_Hackett
6         Kasandra_Homenick
13                Jaclyn81
20                 Rocio33
23        Maxwell.Halvorson
                 ...
195    Keenan.Schamberger60
196          Tomas.Beatty93
197         Imani_Nicolas17
198            Alek_Watsica
199               Javonte83
Name: username, Length: 126, dtype: object
```

In [ ]:
```

```

In [14]:
```
df_likes = pd.read_sql_query("SELECT * from likes",conn)
print(df_likes)
```

```
C:\Users\santhosh\anaconda3\lib\site-packages\pandas\io\sql.py:762: UserWarnin
g: pandas only support SQLAlchemy connectable(engine/connection) ordatabase str
ing URI or sqlite3 DBAPI2 connectionother DBAPI2 objects are not tested, please
consider using SQLAlchemy
  warnings.warn(

      user_id  photo_id          created_at
0           2         1 2022-10-29 12:18:24
1           2         4 2022-10-29 12:18:24
2           2         8 2022-10-29 12:18:24
3           2         9 2022-10-29 12:18:24
4           2        10 2022-10-29 12:18:24
...       ...       ...                 ...
8777      100       245 2022-10-29 12:18:24
8778      100       246 2022-10-29 12:18:24
8779      100       248 2022-10-29 12:18:24
8780      100       249 2022-10-29 12:18:24
8781      100       255 2022-10-29 12:18:24

[8782 rows x 3 columns]
```

3) Declaring Contest Winner: The team started a contest and the user who gets the most likes on a single photo will win the contest now they wish to declare the winner. Your Task: Identify the winner of the contest and provide their details to the team

In [15]:
```
print(df_likes['photo_id'].value_counts())
```

```
145    48
127    43
182    43
123    42
30     41
       ..
50     27
139    27
195    26
223    25
1      25
Name: photo_id, Length: 257, dtype: int64
```

In [16]:
```
a=df_likes[df_likes['photo_id']==145]
print(a)
```

```
      user_id  photo_id          created_at
141         3       145 2022-10-29 12:18:24
230         4       145 2022-10-29 12:18:24
410         5       145 2022-10-29 12:18:24
577         6       145 2022-10-29 12:18:24
912        11       145 2022-10-29 12:18:24
1067       13       145 2022-10-29 12:18:24
1259       14       145 2022-10-29 12:18:24
1518       16       145 2022-10-29 12:18:24
1609       17       145 2022-10-29 12:18:24
1678       18       145 2022-10-29 12:18:24
1856       20       145 2022-10-29 12:18:24
2040       21       145 2022-10-29 12:18:24
2202       22       145 2022-10-29 12:18:24
2388       24       145 2022-10-29 12:18:24
2557       26       145 2022-10-29 12:18:24
2639       27       145 2022-10-29 12:18:24
2800       30       145 2022-10-29 12:18:24
3324       36       145 2022-10-29 12:18:24
3482       37       145 2022-10-29 12:18:24
3651       39       145 2022-10-29 12:18:24
3745       40       145 2022-10-29 12:18:24
3924       41       145 2022-10-29 12:18:24
4427       47       145 2022-10-29 12:18:24
4498       48       145 2022-10-29 12:18:24
4584       50       145 2022-10-29 12:18:24
4662       52       145 2022-10-29 12:18:24
4845       54       145 2022-10-29 12:18:24
5000       55       145 2022-10-29 12:18:24
5080       56       145 2022-10-29 12:18:24
5261       57       145 2022-10-29 12:18:24
5420       60       145 2022-10-29 12:18:24
5516       61       145 2022-10-29 12:18:24
5593       62       145 2022-10-29 12:18:24
5674       63       145 2022-10-29 12:18:24
5954       66       145 2022-10-29 12:18:24
6116       67       145 2022-10-29 12:18:24
6201       69       145 2022-10-29 12:18:24
6302       70       145 2022-10-29 12:18:24
6482       71       145 2022-10-29 12:18:24
6640       72       145 2022-10-29 12:18:24
6910       75       145 2022-10-29 12:18:24
7167       76       145 2022-10-29 12:18:24
7325       78       145 2022-10-29 12:18:24
7646       85       145 2022-10-29 12:18:24
7920       91       145 2022-10-29 12:18:24
8087       92       145 2022-10-29 12:18:24
8342       95       145 2022-10-29 12:18:24
8592       98       145 2022-10-29 12:18:24
```

In [ ]:
```
```

In [18]:
```
df_comments = pd.read_sql_query("SELECT * from comments",conn)
print(df_comments)
```

```
C:\Users\santhosh\anaconda3\lib\site-packages\pandas\io\sql.py:762: UserWarnin
g: pandas only support SQLAlchemy connectable(engine/connection) ordatabase str
ing URI or sqlite3 DBAPI2 connectionother DBAPI2 objects are not tested, please
consider using SQLAlchemy
  warnings.warn(
```

	id	comment_text	user_id	photo_id	created_at
0	1	unde at dolorem	2	1	2022-10-29 12:18:22
1	2	quae ea ducimus	3	1	2022-10-29 12:18:22
2	3	alias a voluptatum	5	1	2022-10-29 12:18:22
3	4	facere suscipit sunt	14	1	2022-10-29 12:18:22
4	5	totam eligendi quaerat	17	1	2022-10-29 12:18:22
...	...	...	...	...	...
7483	7484	accusamus vel est	82	257	2022-10-29 12:18:22
7484	7485	sit nulla qui	91	257	2022-10-29 12:18:22
7485	7486	sed quidem vitae	93	257	2022-10-29 12:18:22
7486	7487	dolorem eveniet rerum	95	257	2022-10-29 12:18:22
7487	7488	dolores nihil voluptas	96	257	2022-10-29 12:18:22

[7488 rows x 5 columns]

4)Hashtag Researching: A partner brand wants to know, which hashtags to use in the post to reach the most people on the platform. Your Task: Identify and suggest the top 5 most commonly used hashtags on the platform

In [19]:
```
a=df_comments['comment_text'].value_counts().nlargest(1)
print(a)
```

```
et et et    3
Name: comment_text, dtype: int64
```

In [ ]:
```

```

In [21]:
```
df_follows = pd.read_sql_query("SELECT * from follows",conn)
print(df_follows)
```

```
C:\Users\santhosh\anaconda3\lib\site-packages\pandas\io\sql.py:762: UserWarnin
g: pandas only support SQLAlchemy connectable(engine/connection) ordatabase str
ing URI or sqlite3 DBAPI2 connectionother DBAPI2 objects are not tested, please
consider using SQLAlchemy
  warnings.warn(
```

	follower_id	followee_id	created_at
0	2	1	2022-10-29 12:18:21
1	2	3	2022-10-29 12:18:21
2	2	4	2022-10-29 12:18:21
3	2	5	2022-10-29 12:18:21
4	2	6	2022-10-29 12:18:21
...	...	...	...
7618	100	95	2022-10-29 12:18:21
7619	100	96	2022-10-29 12:18:21
7620	100	97	2022-10-29 12:18:21
7621	100	98	2022-10-29 12:18:21
7622	100	99	2022-10-29 12:18:21

[7623 rows x 3 columns]

In [ ]:
```
```

In [22]:
```
df_photo_tags = pd.read_sql_query("SELECT * from photo_tags",conn)
print(df_photo_tags)
```

	photo_id	tag_id
0	14	1
1	21	1
2	45	1
3	75	1
4	83	1
..	...	...
496	221	21
497	226	21
498	230	21
499	232	21
500	239	21

[501 rows x 2 columns]

```
C:\Users\santhosh\anaconda3\lib\site-packages\pandas\io\sql.py:762: UserWarnin
g: pandas only support SQLAlchemy connectable(engine/connection) ordatabase str
ing URI or sqlite3 DBAPI2 connectionother DBAPI2 objects are not tested, please
consider using SQLAlchemy
  warnings.warn(
```

In [23]:
```
df_photo_tags.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 501 entries, 0 to 500
Data columns (total 2 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   photo_id  501 non-null    int64
 1   tag_id    501 non-null    int64
dtypes: int64(2)
memory usage: 8.0 KB
```

In [24]:
```
df_tags = pd.read_sql_query("SELECT * from tags",conn)
print(df_tags)
```

```
C:\Users\santhosh\anaconda3\lib\site-packages\pandas\io\sql.py:762: UserWarnin
g: pandas only support SQLAlchemy connectable(engine/connection) ordatabase str
ing URI or sqlite3 DBAPI2 connectionother DBAPI2 objects are not tested, please
consider using SQLAlchemy
  warnings.warn(
```

	id	tag_name	created_at
0	1	sunset	2022-10-29 12:18:26
1	2	photography	2022-10-29 12:18:26
2	3	sunrise	2022-10-29 12:18:26
3	4	landscape	2022-10-29 12:18:26
4	5	food	2022-10-29 12:18:26
5	6	foodie	2022-10-29 12:18:26
6	7	delicious	2022-10-29 12:18:26
7	8	beauty	2022-10-29 12:18:26
8	9	stunning	2022-10-29 12:18:26
9	10	dreamy	2022-10-29 12:18:26
10	11	lol	2022-10-29 12:18:26
11	12	happy	2022-10-29 12:18:26
12	13	fun	2022-10-29 12:18:26
13	14	style	2022-10-29 12:18:26
14	15	hair	2022-10-29 12:18:26
15	16	fashion	2022-10-29 12:18:26
16	17	party	2022-10-29 12:18:26
17	18	concert	2022-10-29 12:18:26
18	19	drunk	2022-10-29 12:18:26
19	20	beach	2022-10-29 12:18:26
20	21	smile	2022-10-29 12:18:26

In [25]:
```
a=df_tags['tag_name'].value_counts()
print(a)
```

```
sunset         1
happy          1
beach          1
drunk          1
concert        1
party          1
fashion        1
hair           1
style          1
fun            1
lol            1
photography    1
dreamy         1
stunning       1
beauty         1
delicious      1
foodie         1
food           1
landscape      1
sunrise        1
smile          1
Name: tag_name, dtype: int64
```

In [ ]:

In [ ]: