DATA PREPROCESSING / transformation

```
from google.colab import files
uploaded = files.upload()
```

Choose Files Cleaned_Insta_Data.csv

- **Cleaned_Insta_Data.csv**(text/csv) - 13230 bytes, last modified: 7/16/2025 - 100% done

```
Saving Cleaned_Insta_Data.csv to Cleaned_Insta_Data (7).csv
```

```
import pandas as pd
df = pd.read_csv('Cleaned_Insta_Data.csv')
df.head()
```

	rank	channel_info	influence_score	posts	followers	avg_likes	60_day_eng_rate	new_post_avg_like	total_likes	country
0	1	cristiano	92	3.3k	475.8m	8.7m	1.39%	6.5m	29.0b	Spain
1	2	kyliejenner	91	6.9k	366.2m	8.3m	1.62%	5.9m	57.4b	United States
2	3	leomessi	90	0.89k	357.3m	6.8m	1.24%	4.4m	6.0b	United States
3	4	selenagomez	93	1.8k	342.7m	6.2m	0.97%	3.3m	11.5b	United States
4	5	therock	91	6.8k	334.1m	1.9m	0.20%	665.3k	12.5b	United States

Next steps: Generate code with df | View recommended plots | New interactive sheet

```
df.dtypes
```

	0
rank	int64
channel_info	object
influence_score	int64
posts	object
followers	object
avg_likes	object
60_day_eng_rate	object
new_post_avg_like	object
total_likes	object
country	object

dtype: object

```
df['posts']=df['posts'].str.lower()
df['posts']
```

	posts
0	3.3k
1	6.9k
2	0.89k
3	1.8k
4	6.8k
...	...
195	2.3k
196	3.8k
197	0.77k
198	2.3k
199	4.2k

200 rows × 1 columns

dtype: object

```
df['posts'] = df['posts'].str.replace('k','', regex = False)
df['posts']
```

	posts
0	3.3
1	6.9
2	0.89
3	1.8
4	6.8
...	...
195	2.3
196	3.8
197	0.77
198	2.3
199	4.2

200 rows × 1 columns

dtype: object

```
df['posts'] = df['posts'].astype('float')*1000
df['posts']
```

	posts
0	3300.0
1	6900.0
2	890.0
3	1800.0
4	6800.0
...	...
195	2300.0
196	3800.0
197	770.0
198	2300.0
199	4200.0

200 rows × 1 columns

dtype: float64

```
df['posts'] = df['posts'].astype('int')
df['posts']
```

	posts
0	3300
1	6900
2	890
3	1800
4	6800
...	...
195	2300
196	3800
197	770
198	2300
199	4200

200 rows × 1 columns

dtype: int64

```
df.head()
```

	rank	channel_info	influence_score	posts	followers	avg_likes	60_day_eng_rate	new_post_avg_like	total_likes	country
0	1	cristiano	92	3300	475.8m	8.7m	1.39%	6.5m	29.0b	Spain
1	2	kyliejenner	91	6900	366.2m	8.3m	1.62%	5.9m	57.4b	United States
2	3	leomessi	90	890	357.3m	6.8m	1.24%	4.4m	6.0b	United States
3	4	selenagomez	93	1800	342.7m	6.2m	0.97%	3.3m	11.5b	United States
4	5	therock	91	6800	334.1m	1.9m	0.20%	665.3k	12.5b	United States

Next steps: Generate code with df | View recommended plots | New interactive sheet

```
df['followers'] = df['followers'].str.lower()
df['followers'] = df['followers'].str.replace('m','', regex = False)
df['followers'] = df['followers'].astype(float)*1000000
df['followers'] = df['followers'].astype(int)
df['followers']
```

	followers
0	475800000
1	366200000
2	357300000
3	342700000
4	334100000
...	...
195	33200000
196	33200000
197	33200000
198	33000000
199	32799999

200 rows × 1 columns

dtype: int64

```
df.head()
```

	rank	channel_info	influence_score	posts	followers	avg_likes	60_day_eng_rate	new_post_avg_like	total_likes	country
0	1	cristiano	92	3300	475800000	8.7m	1.39%	6.5m	29.0b	Spain
1	2	kyliejenner	91	6900	366200000	8.3m	1.62%	5.9m	57.4b	United States
2	3	leomessi	90	890	357300000	6.8m	1.24%	4.4m	6.0b	United States
3	4	selenagomez	93	1800	342700000	6.2m	0.97%	3.3m	11.5b	United States
4	5	therock	91	6800	334100000	1.9m	0.20%	665.3k	12.5b	United States

Next steps: Generate code with df | View recommended plots | New interactive sheet

```
df.dtypes
```

	0
rank	int64
channel_info	object
influence_score	int64
posts	int64
followers	int64
avg_likes	object
60_day_eng_rate	object
new_post_avg_like	object
total_likes	object
country	object

dtype: object

```
df['avg_likes'] = df['avg_likes'].str.lower()
df['avg_likes'] = df['avg_likes'].str.replace('m','', regex = False)
df['avg_likes'] = df['avg_likes'].str.replace('k','', regex = False)
df['avg_likes'] = df['avg_likes'].astype(float)*1000000
df['avg_likes'] = df['avg_likes'].astype(int)
df['avg_likes']
```

	avg_likes
0	8700000
1	8300000
2	6800000
3	6200000
4	1900000
...	...
195	623800000
196	390400000
197	193300000
198	719600000
199	232200000

200 rows × 1 columns

dtype: int64

```
df.head()
```

	rank	channel_info	influence_score	posts	followers	avg_likes	60_day_eng_rate	new_post_avg_like	total_likes	country
0	1	cristiano	92	3300	475800000	8700000	1.39%	6.5m	29.0b	Spain
1	2	kyliejenner	91	6900	366200000	8300000	1.62%	5.9m	57.4b	United States
2	3	leomessi	90	890	357300000	6800000	1.24%	4.4m	6.0b	United States
3	4	selenagomez	93	1800	342700000	6200000	0.97%	3.3m	11.5b	United States
4	5	therock	91	6800	334100000	1900000	0.20%	665.3k	12.5b	United States

Next steps: Generate code with df | View recommended plots | New interactive sheet

```
df.dtypes
```

	0
rank	int64
channel_info	object
influence_score	int64
posts	int64
followers	int64
avg_likes	int64
60_day_eng_rate	object
new_post_avg_like	object
total_likes	object
country	object

dtype: object

```
df['60_day_eng_rate'] = df['60_day_eng_rate'].str.replace('%','',regex = False)
df['60_day_eng_rate'] = df['60_day_eng_rate'].astype(float)
df['60_day_eng_rate']
```

	60_day_eng_rate
0	1.39
1	1.62
2	1.24
3	0.97
4	0.20
...	...
195	1.40
196	0.64
197	0.26
198	1.42
199	0.30

200 rows × 1 columns

dtype: float64

```
df['60_day_eng_rate'].isnull().sum()
```

```
np.int64(1)
```

```
nan_rows = df[df['60_day_eng_rate'].isnull()]
print(nan_rows)
```

```
     rank channel_info  influence_score  posts  followers  avg_likes  \
167   168        rkive               83    110   37000000   10900000

     60_day_eng_rate new_post_avg_like total_likes        country
167              NaN                 0        1.2b  United States
```

```
df.head(168)
```

	rank	channel_info	influence_score	posts	followers	avg_likes	60_day_eng_rate	new_post_avg_like	total_likes	country
0	1	cristiano	92	3300	475800000	8700000	1.39	6.5m	29.0b	Spain
1	2	kyliejenner	91	6900	366200000	8300000	1.62	5.9m	57.4b	United States
2	3	leomessi	90	890	357300000	6800000	1.24	4.4m	6.0b	United States
3	4	selenagomez	93	1800	342700000	6200000	0.97	3.3m	11.5b	United States
4	5	therock	91	6800	334100000	1900000	0.20	665.3k	12.5b	United States
...	...	...	...	...	...	...	...	...	...	...
163	164	prattprattpratt	84	730	38300000	813500000	1.00	363.8k	594.7m	United States
164	165	marvelstudios	83	2700	38100000	594200000	1.58	555.9k	1.6b	United States
										United

Next steps: Generate code with df | View recommended plots | New interactive sheet

```
mean  =  df['60_day_eng_rate'].mean()
mean
```

```
np.float64(1.9020100502512562)
```

```
median = df['60_day_eng_rate'].median()
median
```

```
0.88
```

```
mode = df['60_day_eng_rate'].mode()
mode
```

	60_day_eng_rate
0	0.02

dtype: float64

```
df['60_day_eng_rate'] = df['60_day_eng_rate'].fillna(median)
```

```
df['60_day_eng_rate'].isna().sum()
```

```
np.int64(0)
```

```
df['60_day_eng_rate'] = df['60_day_eng_rate'].astype('float')*100
df['60_day_eng_rate']
```

	60_day_eng_rate
0	139.0
1	162.0
2	124.0
3	97.0
4	20.0
...	...
195	140.0
196	64.0
197	26.0
198	142.0
199	30.0

200 rows × 1 columns

dtype: float64

```
df['60_day_eng_rate'] = df['60_day_eng_rate'].astype('int')
df['60_day_eng_rate']
```

	60_day_eng_rate
0	139
1	162
2	124
3	97
4	20
...	...
195	140
196	64
197	26
198	142
199	30

200 rows × 1 columns

dtype: int64

```
df['60_day_eng_rate'].isna().sum()
```

```
np.int64(0)
```

```
df.head()
```

	rank	channel_info	influence_score	posts	followers	avg_likes	60_day_eng_rate	new_post_avg_like	total_likes	country
0	1	cristiano	92	3300	475800000	8700000	139	6.5m	29.0b	Spain
1	2	kyliejenner	91	6900	366200000	8300000	162	5.9m	57.4b	United States
2	3	leomessi	90	890	357300000	6800000	124	4.4m	6.0b	United States
3	4	selenagomez	93	1800	342700000	6200000	97	3.3m	11.5b	United States
4	5	therock	91	6800	334100000	1900000	20	665.3k	12.5b	United States

Next steps: Generate code with df | View recommended plots | New interactive sheet

```
df.dtypes
```

	0
rank	int64
channel_info	object
influence_score	int64
posts	int64
followers	int64
avg_likes	int64
60_day_eng_rate	int64
new_post_avg_like	object
total_likes	object
country	object

dtype: object

```
def convert_abbreviated_number(x):
    if isinstance(x, str):
        x = x.strip().lower()
        if 'm' in x:
            return float(x.replace('m', '')) * 1_000_000
        elif 'k' in x:
            return float(x.replace('k', '')) * 1_000
        return x
```

```
df['new_post_avg_like']= df['new_post_avg_like'].apply(convert_abbreviated_number)
df['new_post_avg_like']
```

	new_post_avg_like
0	6500000.0
1	5900000.0
2	4400000.0
3	3300000.0
4	665300.0
...	...
195	464700.0
196	208000.0
197	82600.0
198	467700.0
199	97400.0

200 rows × 1 columns

dtype: object