

Analisando o engajamento do Instagram

Equipes: 2 alunos

Entrega: disponível no moodle

Base: disponível no moodle

Valor: 10

O que queremos responder?

- Qual tipo de conteúdo **mais engaja** no Instagram da minha empresa?
- Temos a base de dados do Instagram **desde que o usuário começou a postar na marca até o dia 27/março**
- Ele também dá alguns direcionamentos:
 - Podem ignorar a coluna visualizações, queremos entender apenas curtidas, comentários e interações
 - Tags vazias é que realmente não possuem tag (favor tratar como vazio)

Vamos importar e visualizar a nossa base

```
In [8]: # Importando o pandas  
import pandas as pd
```

```
In [164... # Importar a base em excel disponível no moodle  
# - Base: 08. Analisando o engajamento no Instagram.xlsx  
df = pd.read_excel('BD Analisando o engajamento no Instagram.xlsx')
```

```
In [165... # Visualizando as 5 primeiras linhas  
df.head()
```

Out[165]:

	Tipo	Data	Curtidas	Comentários	Visualizações	Tags	Pessoas	Campanhas	Carrossel
0	Foto	2021-09-11	2858	16	NaN	Loja	N	N	NaN
1	Foto	2021-09-11	2930	28	NaN	Loja/Produtos	N	N	NaN
2	Foto	2021-09-11	2807	9	NaN	Loja	N	N	NaN
3	Vídeo	2021-09-12	5115	49	82,878.00	Produtos	N	N	NaN
4	Foto	2021-09-13	4392	45	NaN	Produtos	S	N	NaN

Não considerar a coluna visualizações, devemos retirar essa coluna da base

In [166]:

```
# Apagando a coluna "Visualizações"
df.drop('Visualizações', axis = 1, inplace = True)
```

In [167]:

```
# Visualizando novamente as 5 primeiras linhas
df.head()
```

Out[167]:

	Tipo	Data	Curtidas	Comentários	Tags	Pessoas	Campanhas	Carrossel	Interacoes
0	Foto	2021-09-11	2858	16	Loja	N	N	NaN	2874
1	Foto	2021-09-11	2930	28	Loja/Produtos	N	N	NaN	2958
2	Foto	2021-09-11	2807	9	Loja	N	N	NaN	2816
3	Vídeo	2021-09-12	5115	49	Produtos	N	N	NaN	5164
4	Foto	2021-09-13	4392	45	Produtos	S	N	NaN	4437

In [168]:

```
# Visualizando as 5 últimas linhas
df.tail()
```

Out[168]:

	Tipo	Data	Curtidas	Comentários	Tags	Pessoas	Campanhas	Carrossel	Id
47	IGTV	2022-03-12	5489	77	Dicas de como usar/Novos Produtos	S	N	NaN	
48	Foto	2022-03-20	29084	479	Datas comemorativas/Promoções	S	S	NaN	
49	Foto	2022-03-22	9087	106	NaN	S	S	NaN	
50	Foto	2022-03-26	16551	186	NaN	S	N	NaN	
51	IGTV	2022-03-27	4934	65	Dicas de como usar/Produtos	S	N	NaN	



In [169]:

```
# Tamanho da base
df[df.columns[0]].count()
```

Out[169]:

52

In [170]:

```
# Se a base for pequena, o display mostra a base completa
df
```

Out[170]:

	Tipo	Data	Curtidas	Comentários	Tags	Pessoas	Campanhas	Carrossel
0	Foto	2021-09-11	2858	16	Loja	N	N	NaN
1	Foto	2021-09-11	2930	28	Loja/Produtos	N	N	NaN
2	Foto	2021-09-11	2807	9	Loja	N	N	NaN
3	Vídeo	2021-09-12	5115	49	Produtos	N	N	NaN
4	Foto	2021-09-13	4392	45	Produtos	S	N	NaN
5	Foto	2021-09-17	5359	62	Novos Produtos	N	S	S
6	Foto	2021-09-19	21597	852	Novos Produtos	S	S	NaN
7	Foto	2021-09-25	6346	33	Produtos	N	S	NaN
8	Foto	2021-09-27	6355	89	Produtos	S	N	S
9	Vídeo	2021-09-28	4056	81	Produtos	N	N	NaN
10	Reels	2021-10-10	12894	249	Trends/Produtos	S	N	NaN
11	Foto	2021-10-12	17831	391	NaN	S	S	NaN
12	Foto	2021-10-21	6166	55	Novos Produtos	S	S	S
13	Foto	2021-10-24	15940	612	Promoções	S	N	NaN
14	Foto	2021-10-29	14121	184	Datas comemorativas	S	S	NaN
15	Vídeo	2021-11-04	3646	71	Produtos	N	N	NaN
16	Reels	2021-11-06	6376	137	Produtos	N	N	NaN
17	Vídeo	2021-11-09	3213	60	Produtos	N	N	NaN
18	Reels	2021-11-12	5493	59	Produtos	N	N	NaN
19	Foto	2021-12-12	16086	268	NaN	S	S	NaN
20	Foto	2021-12-16	2881	29	Produtos	N	N	NaN

	Tipo	Data	Curtidas	Comentários	Tags	Pessoas	Campanhas	Carrossel
21	Foto	2021-12-23	8328	93	Produtos	S	N	S
22	Foto	2021-12-26	18097	225	Influenciadores	S	S	NaN
23	Foto	2021-12-28	8191	94	Influenciadores	S	N	NaN
24	Vídeo	2021-12-30	17600	383	Trends	S	S	NaN
25	Foto	2022-01-02	12193	138	Novos Produtos	S	N	S
26	Foto	2022-01-08	24585	354	Datas comemorativas	S	S	S
27	Foto	2022-01-11	16067	265	Novos Produtos	S	N	NaN
28	Foto	2022-01-15	9936	119	Novos Produtos	S	N	S
29	Foto	2022-01-19	8612	142	NaN	S	N	NaN
30	Reels	2022-01-24	29981	502	Trends	S	S	NaN
31	Foto	2022-01-27	10019	103	Novos Produtos	S	N	NaN
32	IGTV	2022-02-04	9270	222	Dicas de como usar/Produtos	S	N	NaN
33	Foto	2022-02-06	24655	186	Influenciadores	S	S	NaN
34	Foto	2022-02-09	20660	292	Influenciadores	S	S	NaN
35	IGTV	2022-02-10	8556	188	Dicas de como usar/Produtos	S	N	NaN
36	Foto	2022-02-12	11802	102	Produtos	S	S	NaN
37	Vídeo	2022-02-13	15219	357	Datas comemorativas	S	S	NaN
38	Foto	2022-02-15	17687	213	NaN	S	N	NaN
39	Foto	2022-02-17	37351	502	Promoções	S	S	NaN
40	Foto	2022-02-21	21621	213	Influenciadores	S	S	S
41	Foto	2022-02-22	12530	90	NaN	S	N	NaN

	Tipo	Data	Curtidas	Comentários	Tags	Pessoas	Campanhas	Carrossel
42	Foto	2022-02-26	8544	72	Influenciadores	S	S	NaN
43	Foto	2022-03-04	24399	266	NaN	S	S	NaN
44	IGTV	2022-03-07	5918	116	Dicas de como usar/Novos Produtos	S	N	NaN
45	Reels	2022-03-08	19621	275	Trends	S	S	NaN
46	Foto	2022-03-09	4613	50	Influenciadores	S	N	NaN
47	IGTV	2022-03-12	5489	77	Dicas de como usar/Novos Produtos	S	N	NaN
48	Foto	2022-03-20	29084	479	Datas comemorativas/Promoções	S	S	NaN
49	Foto	2022-03-22	9087	106	NaN	S	S	NaN
50	Foto	2022-03-26	16551	186	NaN	S	N	NaN
51	IGTV	2022-03-27	4934	65	Dicas de como usar/Produtos	S	N	NaN

In []: *# Verificando as informações*

Carrossel possui apenas 8 valores não nulos

- Vamos entender os valores de carrossel

In [171]: *# Contando os valores que aparecem na coluna Carrossel*
`df['Carrossel'].notnull().sum()`

Out[171]: 8

- Na verdade, os valores nulos são de postagens que não são carrossel. Sendo assim o nulo deveria ser "N"

Tratando os valores nulos

In [172]: *# Filtrando os valores em que carrossel é nulo*
`databool = pd.isnull(df["Carrossel"])`
`df[databool]`

Out[172]:

	Tipo	Data	Curtidas	Comentários	Tags	Pessoas	Campanhas	Carrossel
0	Foto	2021-09-11	2858	16	Loja	N	N	NaN
1	Foto	2021-09-11	2930	28	Loja/Produtos	N	N	NaN
2	Foto	2021-09-11	2807	9	Loja	N	N	NaN
3	Vídeo	2021-09-12	5115	49	Produtos	N	N	NaN
4	Foto	2021-09-13	4392	45	Produtos	S	N	NaN
6	Foto	2021-09-19	21597	852	Novos Produtos	S	S	NaN
7	Foto	2021-09-25	6346	33	Produtos	N	S	NaN
9	Vídeo	2021-09-28	4056	81	Produtos	N	N	NaN
10	Reels	2021-10-10	12894	249	Trends/Produtos	S	N	NaN
11	Foto	2021-10-12	17831	391	NaN	S	S	NaN
13	Foto	2021-10-24	15940	612	Promoções	S	N	NaN
14	Foto	2021-10-29	14121	184	Datas comemorativas	S	S	NaN
15	Vídeo	2021-11-04	3646	71	Produtos	N	N	NaN
16	Reels	2021-11-06	6376	137	Produtos	N	N	NaN
17	Vídeo	2021-11-09	3213	60	Produtos	N	N	NaN
18	Reels	2021-11-12	5493	59	Produtos	N	N	NaN
19	Foto	2021-12-12	16086	268	NaN	S	S	NaN
20	Foto	2021-12-16	2881	29	Produtos	N	N	NaN
22	Foto	2021-12-26	18097	225	Influenciadores	S	S	NaN
23	Foto	2021-12-28	8191	94	Influenciadores	S	N	NaN
24	Vídeo	2021-12-30	17600	383	Trends	S	S	NaN

	Tipo	Data	Curtidas	Comentários	Tags	Pessoas	Campanhas	Carrossel
27	Foto	2022-01-11	16067	265	Novos Produtos	S	N	NaN
29	Foto	2022-01-19	8612	142	NaN	S	N	NaN
30	Reels	2022-01-24	29981	502	Trends	S	S	NaN
31	Foto	2022-01-27	10019	103	Novos Produtos	S	N	NaN
32	IGTV	2022-02-04	9270	222	Dicas de como usar/Produtos	S	N	NaN
33	Foto	2022-02-06	24655	186	Influenciadores	S	S	NaN
34	Foto	2022-02-09	20660	292	Influenciadores	S	S	NaN
35	IGTV	2022-02-10	8556	188	Dicas de como usar/Produtos	S	N	NaN
36	Foto	2022-02-12	11802	102	Produtos	S	S	NaN
37	Vídeo	2022-02-13	15219	357	Datas comemorativas	S	S	NaN
38	Foto	2022-02-15	17687	213	NaN	S	N	NaN
39	Foto	2022-02-17	37351	502	Promoções	S	S	NaN
41	Foto	2022-02-22	12530	90	NaN	S	N	NaN
42	Foto	2022-02-26	8544	72	Influenciadores	S	S	NaN
43	Foto	2022-03-04	24399	266	NaN	S	S	NaN
44	IGTV	2022-03-07	5918	116	Dicas de como usar/Novos Produtos	S	N	NaN
45	Reels	2022-03-08	19621	275	Trends	S	S	NaN
46	Foto	2022-03-09	4613	50	Influenciadores	S	N	NaN
47	IGTV	2022-03-12	5489	77	Dicas de como usar/Novos Produtos	S	N	NaN
48	Foto	2022-03-20	29084	479	Datas comemorativas/Promoções	S	S	NaN
49	Foto	2022-03-22	9087	106	NaN	S	S	NaN

	Tipo	Data	Curtidas	Comentários	Tags	Pessoas	Campanhas	Carrossel
50	Foto	2022-03-26	16551	186	NaN	S	N	NaN
51	IGTV	2022-03-27	4934	65	Dicas de como usar/Produtos	S	N	NaN

```
In [173]: # Buscando valores que NAO são nulos
databool = pd.notnull(df["Carrossel"])
df[databool]
```

Out[173]:

	Tipo	Data	Curtidas	Comentários	Tags	Pessoas	Campanhas	Carrossel	Interacoes
5	Foto	2021-09-17	5359	62	Novos Produtos	N	S	S	5421
8	Foto	2021-09-27	6355	89	Produtos	S	N	S	6444
12	Foto	2021-10-21	6166	55	Novos Produtos	S	S	S	6221
21	Foto	2021-12-23	8328	93	Produtos	S	N	S	8421
25	Foto	2022-01-02	12193	138	Novos Produtos	S	N	S	12331
26	Foto	2022-01-08	24585	354	Datas comemorativas	S	S	S	24939
28	Foto	2022-01-15	9936	119	Novos Produtos	S	N	S	10055
40	Foto	2022-02-21	21621	213	Influenciadores	S	S	S	21834

```
In [174]: # Selecionando apenas a coluna Carrossel
df['Carrossel']
```

```
Out[174]: 0      NaN
           1      NaN
           2      NaN
           3      NaN
           4      NaN
           5       S
           6      NaN
           7      NaN
           8       S
           9      NaN
          10      NaN
          11      NaN
          12       S
          13      NaN
          14      NaN
          15      NaN
          16      NaN
          17      NaN
          18      NaN
          19      NaN
          20      NaN
          21       S
          22      NaN
          23      NaN
          24      NaN
          25       S
          26       S
          27      NaN
          28       S
          29      NaN
          30      NaN
          31      NaN
          32      NaN
          33      NaN
          34      NaN
          35      NaN
          36      NaN
          37      NaN
          38      NaN
          39      NaN
          40       S
          41      NaN
          42      NaN
          43      NaN
          44      NaN
          45      NaN
          46      NaN
          47      NaN
          48      NaN
          49      NaN
          50      NaN
          51      NaN
Name: Carrossel, dtype: object
```

```
In [175... # Agora vamos atribuir o valor N para essa coluna
base = df.assign(Carrossel='N')
```

Verificando novamente os valores de Carrossel:

```
In [176]: # Verificando novamente os valores dessa coluna  
base['Carrossel']
```

```
Out[176]: 0      N  
1      N  
2      N  
3      N  
4      N  
5      N  
6      N  
7      N  
8      N  
9      N  
10     N  
11     N  
12     N  
13     N  
14     N  
15     N  
16     N  
17     N  
18     N  
19     N  
20     N  
21     N  
22     N  
23     N  
24     N  
25     N  
26     N  
27     N  
28     N  
29     N  
30     N  
31     N  
32     N  
33     N  
34     N  
35     N  
36     N  
37     N  
38     N  
39     N  
40     N  
41     N  
42     N  
43     N  
44     N  
45     N  
46     N  
47     N  
48     N  
49     N  
50     N  
51     N
```

Name: Carrossel, dtype: object

Mostrando as informações estatísticas

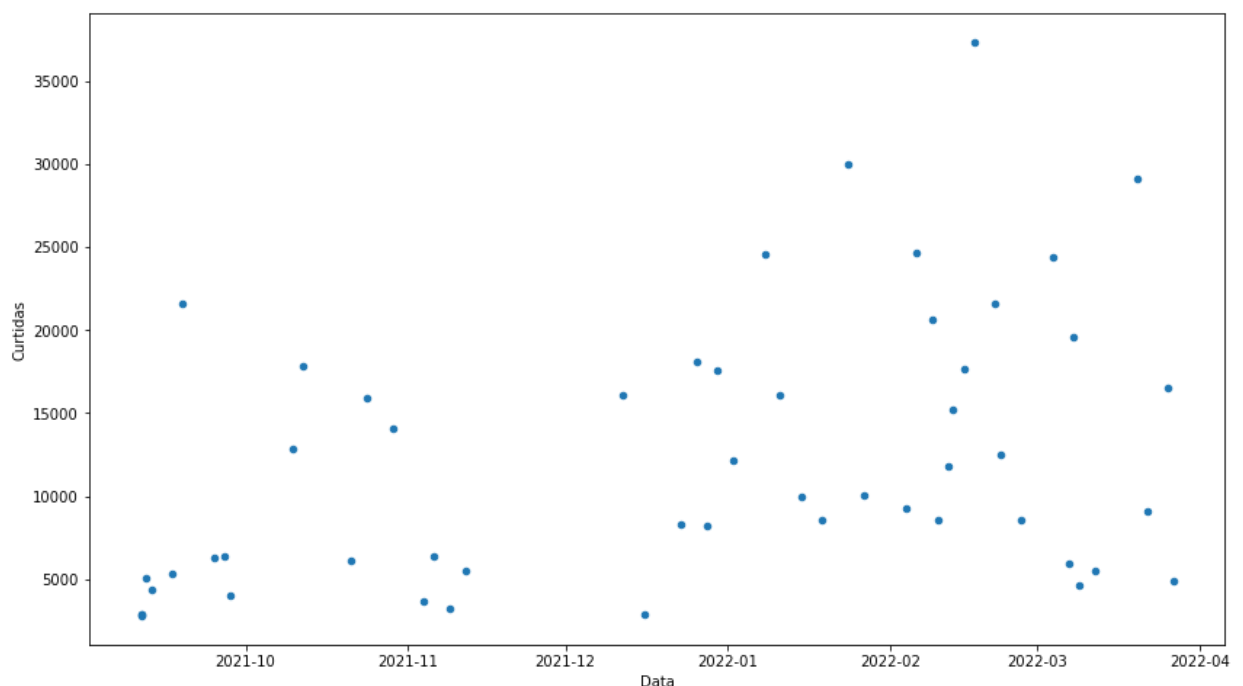
```
In [177... # Descrição estatística da base
base.describe(include = 'all', datetime_is_numeric = True)
```

Out[177]:

	Tipo	Data	Curtidas	Comentários	Tags	Pessoas	Campanhas	Carrossel
count	52	52	52.00	52.00	44	52	52	52
unique	4	NaN	NaN	NaN	12	2	2	1
top	Foto	NaN	NaN	NaN	Produtos	S	N	N
freq	36	NaN	NaN	NaN	12	40	30	52
mean	NaN	2021-12-23 04:36:55.384615424	12,262.73	189.50	NaN	NaN	NaN	NaN
min	NaN	2021-09-11 00:00:00	2,807.00	9.00	NaN	NaN	NaN	NaN
25%	NaN	2021-10-23 06:00:00	5,492.00	69.50	NaN	NaN	NaN	NaN
50%	NaN	2022-01-05 00:00:00	9,603.00	128.00	NaN	NaN	NaN	NaN
75%	NaN	2022-02-15 12:00:00	17,621.75	265.25	NaN	NaN	NaN	NaN
max	NaN	2022-03-27 00:00:00	37,351.00	852.00	NaN	NaN	NaN	NaN
std	NaN	NaN	8,165.88	170.69	NaN	NaN	NaN	NaN

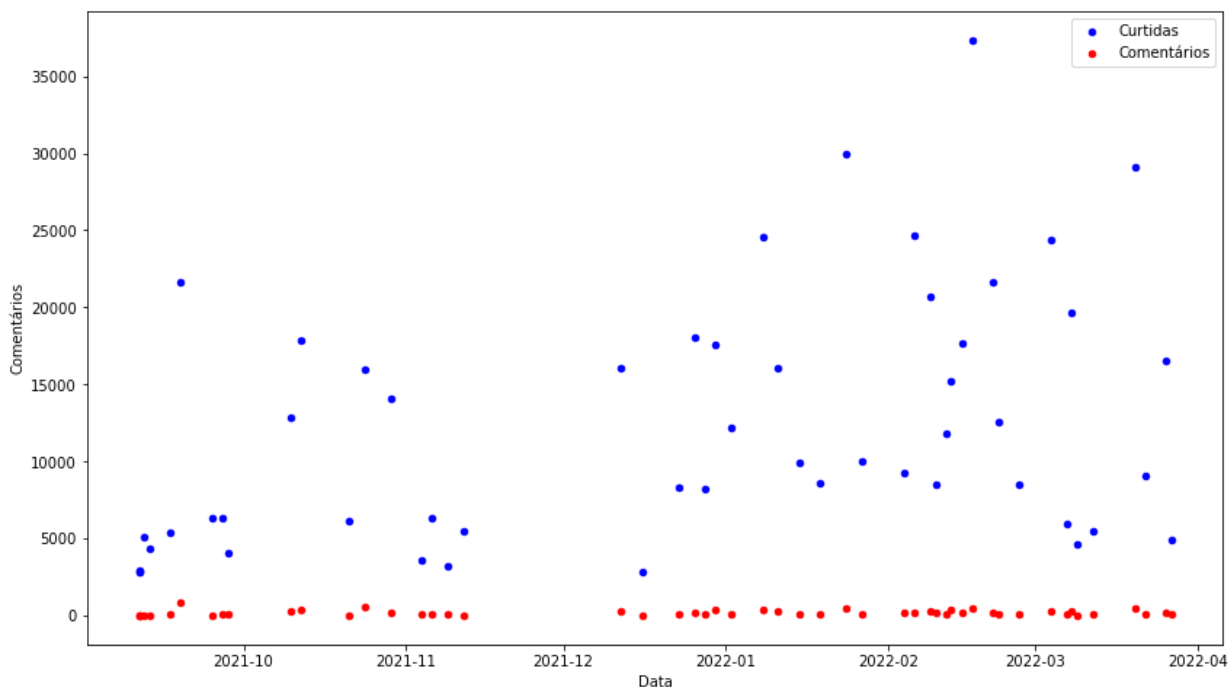
Visualizando essas informações de maneira gráfica

```
In [178... # Um gráfico de dispersão ajudaria a entender melhor curtidas e comentários
base.plot(kind="scatter",x="Data",y="Curtidas",figsize=(14,8));
```



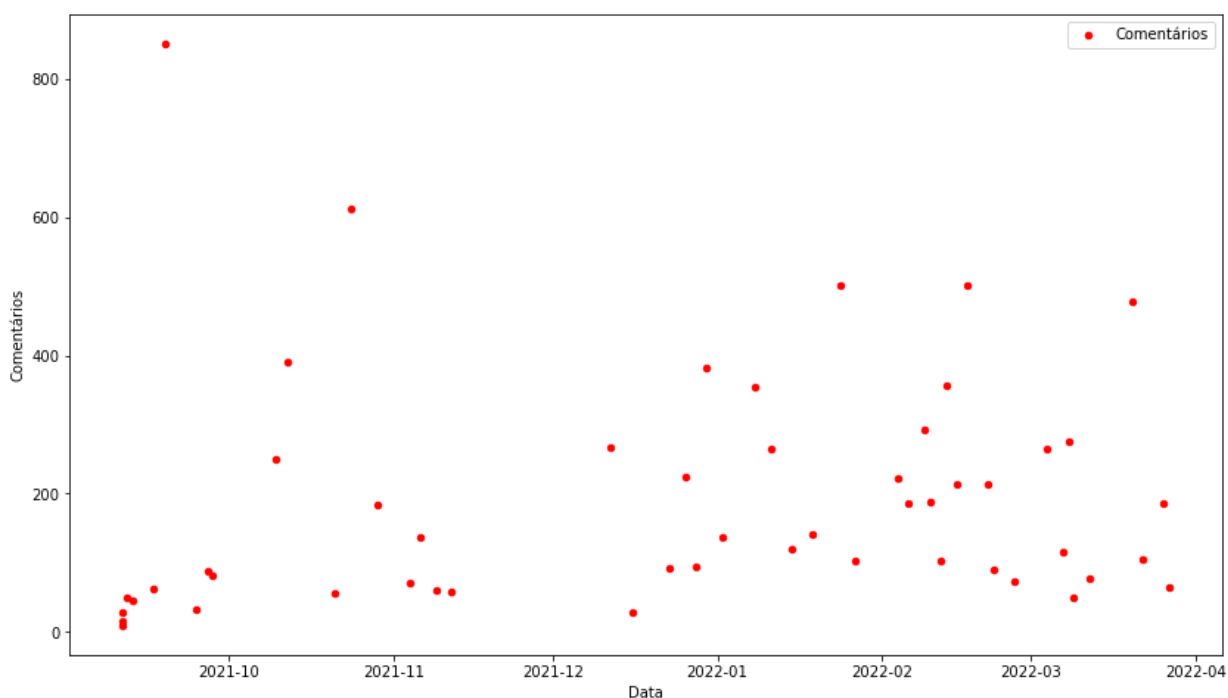
In [179...

```
# Podemos colocar curtidas e comentários no mesmo gráfico
ax = df.plot(kind="scatter",x="Data",y="Curtidas",color="blue",label="Curtidas",figsiz
base.plot(kind="scatter",x="Data",y="Comentários",color="red",label="Comentários",figs
```



In [180...

```
# A escala de curtidas pode estar atrapalhando a visualização, por isso vamos deixar c
base.plot(kind="scatter",x="Data",y="Comentários",color="red",label="Comentários",figs
```



- O gráfico e as informações estatísticas não estão dizendo muita coisa pois existe uma grande dispersão entre curtidas e comentários
- Precisamos verificar se existe um padrão usando as outras colunas de informações

A primeira coisa que podemos fazer é pegar os 5 primeiros

registros com mais e menos curtidas

```
In [181]: # Ordenando os valores
base.sort_values(by="Curtidas",ascending=False).head()
```

Out[181]:

	Tipo	Data	Curtidas	Comentários	Tags	Pessoas	Campanhas	Carrossel	I
39	Foto	2022-02-17	37351	502	Promoções	S	S	N	
30	Reels	2022-01-24	29981	502	Trends	S	S	N	
48	Foto	2022-03-20	29084	479	Datas comemorativas/Promoções	S	S	N	
33	Foto	2022-02-06	24655	186	Influenciadores	S	S	N	
26	Foto	2022-01-08	24585	354	Datas comemorativas	S	S	N	

```
In [182]: # Selecionando os 5 últimos valores
base.sort_values(by="Curtidas",ascending=False).tail()
```

Out[182]:

	Tipo	Data	Curtidas	Comentários	Tags	Pessoas	Campanhas	Carrossel	Interacoes
17	Vídeo	2021-11-09	3213	60	Produtos	N	N	N	3273
1	Foto	2021-09-11	2930	28	Loja/Produtos	N	N	N	2958
20	Foto	2021-12-16	2881	29	Produtos	N	N	N	2910
0	Foto	2021-09-11	2858	16	Loja	N	N	N	2874
2	Foto	2021-09-11	2807	9	Loja	N	N	N	2816

- Podemos observar que no top 5 todas as postagens tinham pessoas e eram fotos de campanha
- Nas 5 piores postagens, não haviam pessoas e nem eram postagens de campanhas
- Isso pode ser um indicador que pessoas e campanhas tem relação com as curtidas

Vamos usar o **group by** para entender melhor os dados

O **group by** é muito análogo a tabela dinâmica que fazemos no Excel

Rótulos de Linha ▾	Média de Curtidas
Foto	13.341,14
IGTV	6.833,40
Reels	14.873,00
Vídeo	8.141,50
Total Geral	12.262,73

```
In [130... # Para melhorar a visualização, vamos criar um padrão no formato dos valores
pd.options.display.float_format = '{:,.2f}'.format
```

```
In [183... # Agrupando as informações por tipo
base.groupby("Tipo")["Comentários"].count()
```

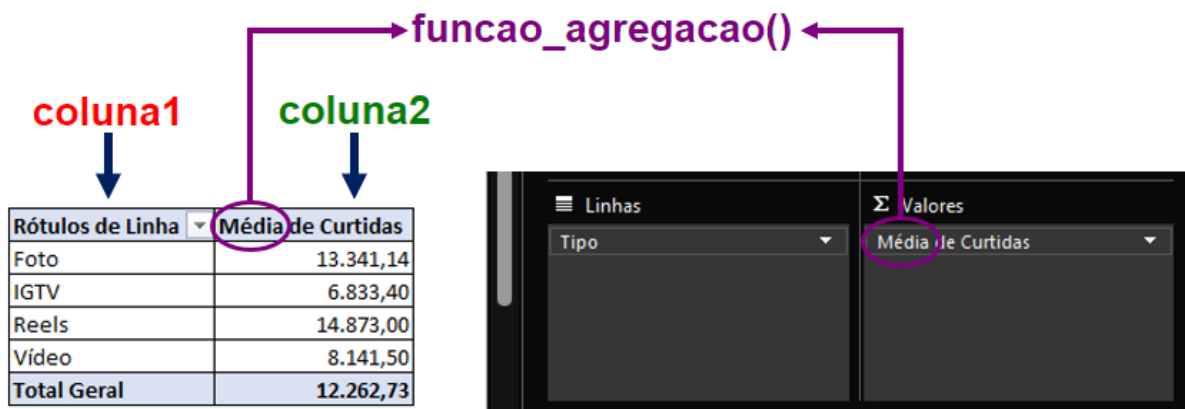
```
Out[183]: Tipo
Foto      36
IGTV      5
Reels     5
Vídeo     6
Name: Comentários, dtype: int64
```

- Nesse caso pode parecer que vídeo e IGTV são estratégias ruins que não devem ser usadas

O **groupby** nos permite agregar os dados, o que facilita muito a visualização

```
base.groupby(**coluna1**)[**coluna2**].**funcao_agregacao**
```

- O **groupby** vai agregar pela **coluna1**
- A **função de agregação** será aplicada na **coluna2**



Inclusive é possível agregar por mais de 1 coluna:

```
In [184... # Agrupando por Tipo e Pessoas
base.groupby(["Tipo", "Pessoas"])["Curtidas"].mean()
```

```
Out[184]:
```

	Tipo	Pessoas
Foto	N	3,863.50
	S	15,236.67
IGTV	S	6,833.40
Reels	N	5,934.50
	S	20,832.00
Vídeo	N	4,007.50
	S	16,409.50

Name: Curtidas, dtype: float64

E também aplicar a função de agregação em várias outras colunas:

```
In [185... base.groupby(["Tipo", "Pessoas"])[["Curtidas", 'Comentários']].mean()
```

```
Out[185]:
```

		Curtidas	Comentários
Tipo	Pessoas		
Foto	N	3,863.50	29.50
	S	15,236.67	226.20
IGTV	S	6,833.40	133.60
Reels	N	5,934.50	98.00
	S	20,832.00	342.00
Vídeo	N	4,007.50	65.25
	S	16,409.50	370.00

```
In [186... # Incluindo a coluna de campanhas
base.groupby(["Tipo", "Pessoas", "Campanhas"])[["Curtidas", 'Comentários']].mean()
```

```
Out[186]:
```

			Curtidas	Comentários
Tipo	Pessoas	Campanhas		
Foto	N	N	2,869.00	20.50
		S	5,852.50	47.50
	S	N	10,815.29	159.93
		S	19,105.38	284.19
IGTV	S	N	6,833.40	133.60
Reels	N	N	5,934.50	98.00
		S	12,894.00	249.00
	S	S	24,801.00	388.50
Vídeo	N	N	4,007.50	65.25
	S	S	16,409.50	370.00

- O groupby já permite ver que publicações de campanha tem um grande engajamento e com foto de pessoas também

Podemos então fazer os agrupamentos que achamos melhor para entender os nossos dados

```
In [187... # Somente para pessoas
base.groupby(["Pessoas"])[["Curtidas", 'Comentários']].mean()
```

Out[187]:

	Curtidas	Comentários
Pessoas		
N	4,256.67	52.83
S	14,664.55	230.50

- **Postagens com pessoas engajam muito mais para essa marca, sendo 3 vezes maior de quando não tem pessoas**

```
In [188... # Somente para campanhas
base.groupby(["Campanhas"])[["Curtidas", 'Comentários']].mean()
```

Out[188]:

	Curtidas	Comentários
Campanhas		
N	7,928.33	123.17
S	18,173.27	279.95

- **Quando é uma postagem de campanha, o engajamento também é melhor!**

```
In [189... # Carrossel (sem filtrar a base)
df.groupby(["Carrossel"])[["Curtidas", 'Comentários']].mean()
```

Out[189]:

	Curtidas	Comentários
Carrossel		
S	11,817.88	140.38

```
In [192... # Podemos também filtrar a base
base.groupby(["Carrossel"])[["Curtidas", 'Comentários']].mean()
```

Out[192]:

	Curtidas	Comentários
Carrossel		
N	12,262.73	189.50

- **A média sem usar carrossel é melhor do que quando usamos, então não é algo que possa impactar tanto no resultado das mídias dessa empresa olhando inicialmente**

- Nesse caso devemos filtrar apenas as fotos pois só temos carrossel em fotos. Sem esse filtro estaríamos comparando coisas erradas

Colocando pessoas e campanhas juntos podemos ver como se dá essa diferença

```
In [193]: # Agregando por pessoas e campanhas
base.groupby(["Pessoas", "Campanhas"])[["Curtidas", 'Comentários']].mean()
```

Out[193]:

		Curtidas	Comentários
Pessoas	Campanhas		
N	N	3,937.50	53.90
	S	5,852.50	47.50
S	N	9,923.75	157.80
	S	19,405.35	303.20

- A média quando tem pessoas E é publicação de campanhas é de cerca de 19,4 mil curtidas, já quando é apenas pessoas (sem campanha passa para quase 10 mil e se não tiver pessoas chega no máximo a 5,9 mil mesmo em campanhas
- Nesse caso a gente já consegue mostrar para a empresa a importância de incluir pessoas usando os seus produtos, o que gera um aumento considerável no engajamento

```
In [194]: # Agregando por pessoas, campanhas e tipo
base.groupby(["Pessoas", "Campanhas", "Tipo"])[["Curtidas", 'Comentários']].mean()
```

Out[194]:

			Curtidas	Comentários
Pessoas	Campanhas	Tipo		
N	N	Foto	2,869.00	20.50
		Reels	5,934.50	98.00
		Vídeo	4,007.50	65.25
S	S	Foto	5,852.50	47.50
	N	Foto	10,815.29	159.93
		IGTV	6,833.40	133.60
S	S	Reels	12,894.00	249.00
		Foto	19,105.38	284.19
		Reels	24,801.00	388.50
S	S	Vídeo	16,409.50	370.00

- Analisando novamente a questão do vídeo, ele não parece mais tão ruim assim.

Quando feito em campanha e usando pessoas ele teve um resultado bom, inclusive próximo a foto

- O que poderia ter levado a média baixa é que só temos vídeo ou COM pessoa e COM campanha ou sem nenhum dos dois. Não temos nenhum vídeo com apenas um dos dois (pessoa ou campanha)
- Já IGTV, mesmo tendo pessoa, não teve um resultado tão bom

Inclusive podemos entender o que havia gerado a média baixo no vídeo

```
In [206]: # Vamos filtrar a base apenas onde o tipo é Vídeo
video = base['Tipo'] == 'Vídeo'
base[video].groupby(['Tipo'])[['Curtidas', 'Comentários']].mean()
```

```
Out[206]:
```

	Curtidas	Comentários
Tipo		
Vídeo	8,141.50	166.83

Conclusões

Cite 3 conclusões que se chegou após a realização da análise da base em questão

```
In [ ]: A média de publicações envolvendo pessoas é sempre superior, logo, é sempre bom incluir
IGTV não possui um resultado bom, mesmo em campanha e com pessoas;
Vídeos possuem menos engajamentos do que fotos, porém se o mesmo possuir pessoas e uma
```