

# Analizando o engajamento do Instagram

## 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

```
# Importando o pandas
import pandas as pd
```

```
# Importar a base em excel
# Base: engajamento_instagram.xlsx
base = pd.read_excel('engajamento_instagram.xlsx')
```

```
# Visualizando as 5 primeiras linhas
base.head()
```

border="1" class="dataframe">

	Tipo	Data	Curtidas	Comentários	Visualizações	Tags	Pessoas	Campanhas	Carrossel	Interações
0	Foto	2021-09-11	2858	16	NaN	Loja	N	N	NaN	2874
1	Foto	2021-09-11	2930	28	NaN	Loja/Produtos	N	N	NaN	2958
2	Foto	2021-09-11	2807	9	NaN	Loja	N	N	NaN	2816
3	Vídeo	2021-09-12	5115	49	82878.0	Produtos	N	N	NaN	5164
4	Foto	2021-09-13	4392	45	NaN	Produtos	S	N	NaN	4437

Como a empresa pediu para ignorarmos a coluna `visualizações`, nós podemos apagar ela da nossa base de dados.

O motivo é simples, estamos trabalhando com uma base de dados consideravelmente pequena, porém, quando estivermos trabalhando com uma grande, quanto mais colunas/linhas desnecessárias tivermos, pior é para nossa análise, pois ficará consumindo mais processamento e memória.

### Em python, o que não te ajuda, te atrapalha ~ Lira da Hashtag

Para apagar uma coluna, utilizamos a seguinte sintaxe: `nome_base.drop('nome_coluna', axis=1)`

- O `axis=1` se refere a coluna, enquanto `axis=0` se refere a linha;
- Devemos passar o nome da coluna que queremos apagar a base;
- Em caso de mais de 1 coluna, passamos a lista entre colchetes.
- OBS: o `inplace=True` faz com que as modificações realizadas fiquem gravadas na base original.

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

```
base.head()
```

border="1" class="dataframe">

	Tipo	Data	Curtidas	Comentários	Tags	Pessoas	Campanhas	Carrossel	Interações
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

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

border="1" class="dataframe">

	Tipo	Data	Curtidas	Comentários	Tags	Pessoas	Campanhas	Carrossel	Interações
47	IGTV	2022-03-12	5489	77	Dicas de como usar/Novos Produtos	S	N	NaN	5566
48	Foto	2022-03-20	29084	479	Datas comemorativas/Promoções	S	S	NaN	29563
49	Foto	2022-03-22	9087	106	NaN	S	S	NaN	9193
50	Foto	2022-03-26	16551	186	NaN	S	N	NaN	16737
51	IGTV	2022-03-27	4934	65	Dicas de como usar/Produtos	S	N	NaN	4999

```
# Tamanho da base
base.shape
```

```
(52, 9)
```

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

border="1" class="dataframe">

	Tipo	Data	Curtidas	Comentários	Tags	Pessoas	Campanhas	Carrossel	Interações
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
5	Foto	2021-09-17	5359	62	Novos Produtos	N	S	S	5421
6	Foto	2021-09-19	21597	852	Novos Produtos	S	S	NaN	22449
7	Foto	2021-09-25	6346	33	Produtos	N	S	NaN	6379
8	Foto	2021-09-27	6355	89	Produtos	S	N	S	6444
9	Vídeo	2021-09-28	4056	81	Produtos	N	N	NaN	4137
10	Reels	2021-10-10	12894	249	Trends/Produtos	S	N	NaN	13143
11	Foto	2021-10-12	17831	391	NaN	S	S	NaN	18222
12	Foto	2021-10-21	6166	55	Novos Produtos	S	S	S	6221
13	Foto	2021-10-24	15940	612	Promoções	S	N	NaN	16552
14	Foto	2021-10-29	14121	184	Datas comemorativas	S	S	NaN	14305
15	Vídeo	2021-11-04	3646	71	Produtos	N	N	NaN	3717
16	Reels	2021-11-06	6376	137	Produtos	N	N	NaN	6513
17	Vídeo	2021-11-09	3213	60	Produtos	N	N	NaN	3273
18	Reels	2021-11-12	5493	59	Produtos	N	N	NaN	5552
19	Foto	2021-12-12	16086	268	NaN	S	S	NaN	16354
20	Foto	2021-12-16	2881	29	Produtos	N	N	NaN	2910

	Tipo	Data	Curtidas	Comentários	Tags	Pessoas	Campanhas	Carrossel	Interações
21	Foto	2021-12-23	8328	93	Produtos	S	N	S	8421
22	Foto	2021-12-26	18097	225	Influenciadores	S	S	NaN	18322
23	Foto	2021-12-28	8191	94	Influenciadores	S	N	NaN	8285
24	Vídeo	2021-12-30	17600	383	Trends	S	S	NaN	17983
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
27	Foto	2022-01-11	16067	265	Novos Produtos	S	N	NaN	16332
28	Foto	2022-01-15	9936	119	Novos Produtos	S	N	S	10055
29	Foto	2022-01-19	8612	142	NaN	S	N	NaN	8754
30	Reels	2022-01-24	29981	502	Trends	S	S	NaN	30483
31	Foto	2022-01-27	10019	103	Novos Produtos	S	N	NaN	10122
32	IGTV	2022-02-04	9270	222	Dicas de como usar/Produtos	S	N	NaN	9492
33	Foto	2022-02-06	24655	186	Influenciadores	S	S	NaN	24841
34	Foto	2022-02-09	20660	292	Influenciadores	S	S	NaN	20952
35	IGTV	2022-02-10	8556	188	Dicas de como usar/Produtos	S	N	NaN	8744
36	Foto	2022-02-12	11802	102	Produtos	S	S	NaN	11904
37	Vídeo	2022-02-13	15219	357	Datas comemorativas	S	S	NaN	15576
38	Foto	2022-02-15	17687	213	NaN	S	N	NaN	17900
39	Foto	2022-02-17	37351	502	Promoções	S	S	NaN	37853

	Tipo	Data	Curtidas	Comentários	Tags	Pessoas	Campanhas	Carrossel	Interações
40	Foto	2022-02-21	21621	213	Influenciadores	S	S	S	21834
41	Foto	2022-02-22	12530	90	NaN	S	N	NaN	12620
42	Foto	2022-02-26	8544	72	Influenciadores	S	S	NaN	8616
43	Foto	2022-03-04	24399	266	NaN	S	S	NaN	24665
44	IGTV	2022-03-07	5918	116	Dicas de como usar/Novos Produtos	S	N	NaN	6034
45	Reels	2022-03-08	19621	275	Trends	S	S	NaN	19896
46	Foto	2022-03-09	4613	50	Influenciadores	S	N	NaN	4663
47	IGTV	2022-03-12	5489	77	Dicas de como usar/Novos Produtos	S	N	NaN	5566
48	Foto	2022-03-20	29084	479	Datas comemorativas /Promoções	S	S	NaN	29563
49	Foto	2022-03-22	9087	106	NaN	S	S	NaN	9193
50	Foto	2022-03-26	16551	186	NaN	S	N	NaN	16737
51	IGTV	2022-03-27	4934	65	Dicas de como usar/Produtos	S	N	NaN	4999

```
# Verificando as informações
base.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 52 entries, 0 to 51
Data columns (total 9 columns):
#   Column      Non-Null Count  Dtype
---  -
0   Tipo        52 non-null     object
1   Data        52 non-null     datetime64[ns]
```

```

2    Curtidas      52 non-null    int64
3    Comentários  52 non-null    int64
4    Tags          44 non-null    object
5    Pessoas       52 non-null    object
6    Campanhas     52 non-null    object
7    Carrossel     8 non-null     object
8    Interacoes   52 non-null    int64
dtypes: datetime64[ns](1), int64(3), object(5)
memory usage: 3.8+ KB

```

## A coluna **Carrossel** possui apenas 8 valores não nulos

- Vamos entender os valores de carrossel:

```

# Contando os valores que aparecem na coluna Carrossel
base['Carrossel'].value_counts()

```

```

Carrossel
S      8
Name: count, dtype: int64

```

- Na verdade, os valores nulos são de postagens que não são carrossel, ou seja, 'S' para sendo postagens e o nulo deveria ser "N".

## Tratando os valores nulos

```

# Filtrando os valores em que carrossel é nulo
base.loc[base['Carrossel'].isnull()].head()

```

border="1" class="dataframe">

	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

```

# Filtrando os valores em que carrossel não é nulo
base.loc[base['Carrossel'].notnull()].head()

```

border="1" class="dataframe">

	Tipo	Data	Curtidas	Comentários	Visualizações	Tags	Pessoas	Campanhas	Carrossel	Interações
5	Foto	2021-09-17	5359	62	NaN	Novos Produtos	N	S	S	5421
8	Foto	2021-09-27	6355	89	NaN	Produtos	S	N	S	6444
12	Foto	2021-10-21	6166	55	NaN	Novos Produtos	S	S	S	6221
21	Foto	2021-12-23	8328	93	NaN	Produtos	S	N	S	8421
25	Foto	2022-01-02	12193	138	NaN	Novos Produtos	S	N	S	12331

```
# Selecionando apenas a coluna Carrossel
base.loc[base['Carrossel'].isnull(), "Carrossel"].head()
```

```
0    NaN
1    NaN
2    NaN
3    NaN
4    NaN
Name: Carrossel, dtype: object
```

```
# Agora vamos atribuir o valor N para essa coluna
base.loc[base['Carrossel'].isnull(), "Carrossel"] = "N"
```

**Verificando novamente os valores de Carrossel**

```
base.head()
```

border="1" class="dataframe">

	Tipo	Data	Curtidas	Comentários	Visualizações	Tags	Pessoas	Campanhas	Carrossel	Interações
0	Foto	2021-09-11	2858	16	NaN	Loja	N	N	N	2874
1	Foto	2021-09-11	2930	28	NaN	Loja/Produtos	N	N	N	2958
2	Foto	2021-09-11	2807	9	NaN	Loja	N	N	N	2816
3	Vídeo	2021-09-12	5115	49	82,878.00	Produtos	N	N	N	5164
4	Foto	2021-09-13	4392	45	NaN	Produtos	S	N	N	4437



```
base['Carrossel'].value_counts()
```

```
Carrossel
N      44
S       8
Name: count, dtype: int64
```

## Mostrando as informações estatísticas

```
# Descrição estatística da base
base.describe()
```

```
border="1" class="dataframe">
```

	Data	Curtidas	Comentários	Interacoes
<b>count</b>	52	52.000000	52.000000	52.000000
<b>mean</b>	2021-12-23 04:36:55.384615 424	12262.730769	189.500000	12452.230769
<b>min</b>	2021-09-11 00:00:00	2807.000000	9.000000	2816.000000
<b>25%</b>	2021-10-23 06:00:00	5492.000000	69.500000	5562.500000
<b>50%</b>	2022-01-05 00:00:00	9603.000000	128.000000	9773.500000
<b>75%</b>	2022-02-15 12:00:00	17621.750000	265.250000	17920.750000
<b>max</b>	2022-03-27 00:00:00	37351.000000	852.000000	37853.000000
<b>std</b>	NaN	8165.875326	170.687709	8299.390088

Para termos mais ou menos uma ideia e uma análise dos dados iniciais (ou até mesmo descobrir um padrão), a primeira coisa que podemos fazer é **pegar os 5 primeiros registros com mais e menos curtidas**.

```
# Ordenando os valores
base.sort_values(by="Curtidas", ascending=False).head()
```

border="1" class="dataframe">

	Tipo	Data	Curtidas	Comentários	Tags	Pessoas	Campanhas	Carrossel	Interações
39	Foto	2022-02-17	37351	502	Promoções	S	S	N	37853
30	Reels	2022-01-24	29981	502	Trends	S	S	N	30483
48	Foto	2022-03-20	29084	479	Datas comemorativas / Promoções	S	S	N	29563
33	Foto	2022-02-06	24655	186	Influenciadores	S	S	N	24841
26	Foto	2022-01-08	24585	354	Datas comemorativas	S	S	S	24939

```
# Ordenando os valores de forma ascendente
base.sort_values(by="Curtidas", ascending=True).head()

# OU

# base.sort_values(by="Curtidas", ascending=False).tail()
```

border="1" class="dataframe">

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

- 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**

# groupby

Vamos usar o **groupby** para entender melhor os dados

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

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

```
# Agrupando as informações por tipo
base.groupby("Tipo")["Curtidas"].mean()
```

Tipo	
Foto	13,341.14
IGTV	6,833.40
Reels	14,873.00
Vídeo	8,141.50

Name: Curtidas, dtype: float64

- 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()
```

- Entre parenteses - **coluna1** - é a informação que queremos continuar analisando;
- Já o que está entre colchetes - **[coluna2]** - é a coluna que eu quero aplicar a minha função de agregação

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

**Inclusive, é possível agregar por mais de uma 1 coluna:**

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

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:

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

border="1" class="dataframe">

		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

```
# Somente para pessoas
```

```
base.groupby("Pessoas")[["Curtidas", "Comentários"]].mean()
```

border="1" class="dataframe">

	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

```
# Somente para campanhas
```

```
base.groupby("Campanhas")[["Curtidas", "Comentários"]].mean()
```

border="1" class="dataframe">

	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!

Outra coisa interessante é que também podemos filtrar a base

Vamos fazer a seguinte pergunta: Quais os carrosseis que são apenas de fotos? E o engajamento deles, como são?

```
base[base['Tipo'] == 'Foto'].groupby(['Carrossel'])[['Curtidas',
"Comentários"]].mean()
```

border="1" class="dataframe">

	Carrossel	Curtidas	Comentários
	N	13,776.36	208.57
	S	11,817.88	140.38

- 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.
  - 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

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

border="1" class="dataframe">

	Pessoas	Campanhas	Curtidas	Comentários
	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.

## Conclusões

- Em uma análise inicial, postagens incluindo pessoas engajam muito mais que aquelas que não possui ninguém;
- Postagens em épocas de campanha também possuem um melhor engajamento;
- Nessa base, o carrossel não foi um diferencial para melhorar o engajamento da marca.