

# Desafio Convenia

---

Este é um desafio proposto pela Convenia, com a finalidade de retirar insights de uma base de dados disponibilizada

## Base de Dados

A base de dados são duas planilhas:

- celularessubtraidos\_2024\_1\_6\_(1)
- celularessubtraidos\_2024\_7\_9\_(1)

Temos também a planilha *dicionario\_bd\_(1)* que explica melhor sobre as bases de dados, quais informações temos o que cada campo significa.

Com as bases disponibilizadas, utilizei o Python para poder realizar o primeiro tratamento de dados

---

## Python

Primeiramente, importei a biblioteca *Pandas* que será necessária para realizar a importação das planilhas e realizar os tratamentos dos dados.

```
import pandas as pd
```

Após a importação do *Pandas*, importei a primeira planilha e utilizei a função *info()* para poder verificar melhor os dados dessa primeira planilha, como nome das colunas, tipo de dados e quantos dados não nulos existentes:

```
celulares_subtraidos1 = pd.read_excel('celularessubtraidos_2024_1_6_(1).xlsb')
```

```
celulares_subtraidos1.info()
```

#	Column	Non-Null Count	Dtype
0	ID_DELEGACIA	175204	non-null int64
1	NOME_DEPARTAMENTO	175204	non-null object
2	NOME_SECCIONAL	175204	non-null object
3	NOME_DELEGACIA	175204	non-null object
4	NOME_MUNICIPIO	175204	non-null object
5	ANO_BO	175204	non-null int64
6	NUM_BO	175204	non-null object
7	VERSAO	175204	non-null int64
8	NOME_DEPARTAMENTO_CIRC	175204	non-null object
9	NOME_SECCIONAL_CIRC	175204	non-null object
10	NOME_DELEGACIA_CIRC	175204	non-null object
11	NOME_MUNICIPIO_CIRC	175204	non-null object
12	DATA_OCORRENCIA_BO	175204	non-null int64
13	HORA_OCORRENCIA	99970	non-null float64
14	DESCRICAO_APRESENTACAO	175204	non-null object
15	DATAHORA_REGISTRO_BO	175204	non-null int64
16	DATA_COMUNICACAO_BO	175204	non-null int64
17	DATAHORA_IMPRESSAO_BO	175106	non-null float64
18	DESCR_PERIODO	75234	non-null object
19	AUTORIA_BO	175204	non-null object
20	FLAG_INTOLERANCIA	175204	non-null object
21	TIPO_INTOLERANCIA	13	non-null object
22	FLAG_FLAGRANTE	175204	non-null object
23	FLAG_STATUS	175204	non-null object
24	DESC_LEI	175204	non-null object
25	FLAG_ATO_INFRAACIONAL	175204	non-null object
26	RUBRICA	175204	non-null object
27	DESCR_CONDUTA	145276	non-null object
28	DESDOBRAMENTO	5041	non-null object
29	CIRCUNSTANCIA	34015	non-null object
30	DESCR_TIPOLOCAL	167458	non-null object
31	DESCR_SUBTIPOLOCAL	170346	non-null object
32	CIDADE	175204	non-null object
33	BAIRRO	173442	non-null object
34	CEP	155494	non-null float64
35	LOGRADOURO_VERSAO	175204	non-null int64

Fiz o mesmo processo com a segunda planilha

```
celulares_subtraidos2 = pd.read_excel('celularessubtraidos_2024_7_9_(1).xlsb')
```

```
celulares_subtraidos2.info()
```

Data columns (total 51 columns):			
#	Column	Non-Null Count	Dtype
0	ID_DELEGACIA	91129 non-null	int64
1	NOME_DEPARTAMENTO	91129 non-null	object
2	NOME_SECCIONAL	91129 non-null	object
3	NOME_DELEGACIA	91129 non-null	object
4	NOME_MUNICIPIO	91129 non-null	object
5	ANO_BO	91129 non-null	int64
6	NUM_BO	91129 non-null	object
7	VERSAO	91129 non-null	int64
8	NOME_DEPARTAMENTO_CIRC	91129 non-null	object
9	NOME_SECCIONAL_CIRC	91129 non-null	object
10	NOME_DELEGACIA_CIRC	91129 non-null	object
11	NOME_MUNICIPIO_CIRC	91129 non-null	object
12	DATA_OCORRENCIA_BO	91129 non-null	int64
13	HORA_OCORRENCIA	49446 non-null	object
14	DESCRICAO_APRESENTACAO	91129 non-null	object
15	DATAHORA_REGISTRO_BO	91129 non-null	int64
16	DATA_COMUNICACAO_BO	91129 non-null	int64
17	DATAHORA_IMPRESSAO_BO	91090 non-null	float64
18	DESCR_PERIODO	41683 non-null	object
19	AUTORIA_BO	91129 non-null	object
20	FLAG_INTOLERANCIA	91129 non-null	object
21	TIPO_INTOLERANCIA	2 non-null	object
22	FLAG_FLAGRANTE	91129 non-null	object
23	FLAG_STATUS	91129 non-null	object
24	DESC_LEI	91129 non-null	object
25	FLAG_ATO_INFRACIONAL	91129 non-null	object
26	RUBRICA	91129 non-null	object
27	DESCR_CONDUTA	74506 non-null	object
28	DESDOBRAMENTO	2478 non-null	object
29	CIRCUNSTANCIA	16398 non-null	object
30	DESCR_TIPOLOCAL	91129 non-null	object
31	DESCR_SUBTIPOLOCAL	91129 non-null	object
32	CIDADE	91129 non-null	object
33	BAIRRO	90198 non-null	object
34	CEP	82762 non-null	float64
35	LOGRADOURO_VERSAO	91129 non-null	int64
36	LOGRADOURO	91129 non-null	object
37	NUMERO_LOGRADOURO	86377 non-null	float64
38	LATITUDE	77957 non-null	float64

Após o processo de importação das planilhas, realizei a união das duas planilhas, pois, as duas possuem a mesma estrutura de dados (mesma quantidade de colunas com o mesmo nome).

```
df = pd.concat([celulares_subtraidos1, celulares_subtraidos2])
```

Agora temos um DataFrame com o nome de *df* que possui os dados das duas planilhas juntas.

Data columns (total 51 columns):			
#	Column	Non-Null Count	Dtype
0	ID_DELEGACIA	266333	non-null int64
1	NOME_DEPARTAMENTO	266333	non-null object
2	NOME_SECCIONAL	266333	non-null object
3	NOME_DELEGACIA	266333	non-null object
4	NOME_MUNICIPIO	266333	non-null object
5	ANO_BO	266333	non-null int64
6	NUM_BO	266333	non-null object
7	VERSAO	266333	non-null int64
8	NOME_DEPARTAMENTO_CIRC	266333	non-null object
9	NOME_SECCIONAL_CIRC	266333	non-null object
10	NOME_DELEGACIA_CIRC	266333	non-null object
11	NOME_MUNICIPIO_CIRC	266333	non-null object
12	DATA_OCORRENCIA_BO	266333	non-null int64
13	HORA_OCORRENCIA	149416	non-null object
14	DESCRICAO_APRESENTACAO	266333	non-null object
15	DATAHORA_REGISTRO_BO	266333	non-null int64
16	DATA_COMUNICACAO_BO	266333	non-null int64
17	DATAHORA_IMPRESSAO_BO	266196	non-null float64
18	DESCR_PERIODO	116917	non-null object
19	AUTORIA_BO	266333	non-null object
20	FLAG_INTOLERANCIA	266333	non-null object
21	TIPO_INTOLERANCIA	15	non-null object
22	FLAG_FLAGRANTE	266333	non-null object
23	FLAG_STATUS	266333	non-null object
24	DESC_LEI	266333	non-null object
25	FLAG_ATO_INFRACIONAL	266333	non-null object
26	RUBRICA	266333	non-null object
27	DESCR_CONDUTA	219782	non-null object
28	DESCR_CONDUTA	7519	non-null object
29	CIRCUNSTANCIA	50413	non-null object
30	DESCR_TIPOLOCAL	258587	non-null object

Algumas colunas apresentam valores nulos com tipo *object*. Para esses casos, irei utilizar a função *unique()* para poder visualizar os valores distintos em determinadas colunas.

Neste caso, irei realizar o mesmo processo abaixo nas colunas:

- DESDOBRAMENTO
- TIPO\_INTOLERANCIA
- DESCR\_CONDUTA
- DESCR\_UNIDADE
- CIRCUNSTANCIA
- DESCR\_TIPOLOCAL
- DESCR\_SUBTIPOLOCAL
- FLAG\_BLOQUEIO
- FLAG\_DESBLOQUEIO
- DESCR\_PERIODO

As demais colunas de Data, Hora ou Número, serão tratadas no Power BI.

```
df['TIPO_INTOLERANCIA'].unique()
```

No caso da coluna TIPO\_INTOLERANCIA, temos os seguintes valores abaixo:

```
array([nan, 'Homofobia/Transfobia', 'Racial/Etnia/Cor'], dtype=object)
```

Como podemos visualizar, temos o valor *nan* que seriam os valores nulos, juntamente com outros dois valores preenchidos.

Para poder tratar os valores nulos, iremos utilizar a função *fillna()* que realizar o tratamento de todos os valores nulos dentro de determinado campo. No caso, iremos preencher dentro dessa função o valor *Não Informado* para que os valores nulos sejam alterados para esse valor.

```
df['TIPO_INTOLERANCIA'] = df['TIPO_INTOLERANCIA'].fillna('Não Informado')
```

Validando novamente a coluna, podemos verificar que todos os valores não nulos foram preenchidos, ficando da seguinte forma:

**Antes:**

```
21 TIPO_INTOLERANCIA      15 non-null    object
```

**Depois**

```
21 TIPO_INTOLERANCIA      266333 non-null    object
```

Finalizado o processo de tratamento dos dados, iremos exportar os dados para um arquivo *celulares\_subtraidos\_consolidado* no qual iremos carregar no Power BI, finalizar o tratamento dos dados e realizar a construção dos painéis.

```
df.to_excel('celulares_subtraidos_consolidado.xlsx')
```

## Power BI

Após a extração da planilha, iremos subir o arquivo no Power BI para ser tratado no Power Query.

Iremos chamar essa base de *Base de Dados*.

Primeiramente, ocorre o processo padrão utilizar a primeira linha como cabeçalho da coluna e também iremos identificar o tipo da coluna de forma automática:

**ETAPAS APLICADAS**

- Fonte
- Navegação
- Cabeçalhos Promovidos
- Tipo Alterado

Após esse processo, iremos retirar as duplicatas da base de dados.

As duplicatas, conforme informado dentro da planilha que explica sobre a base de dados, devem ser removidas utilizando 3 colunas:

- NOME\_DELEGACIA
- ANO\_BO
- NUM\_BO

Table.Distinct(#"Tipo Alterado", {"NOME\_DELEGACIA", "ANO\_BO", "NUM\_BO"})

ID_DELEGACIA	NOME_DEPARTAMENTO	NOME_SECCIONAL	NOME_DELEGACIA	NOME_MUNICIPIO	ANO_BO	NUM_BO	VERSAO
0	900020 DIPOL - DEPTO DE INTELIGENCIA	DELEGACIA ELETRONICA	DELEGACIA ELETRONICA	S.PAULO	2024	AA0187	
1	900020 DIPOL - DEPTO DE INTELIGENCIA	DELEGACIA ELETRONICA	DELEGACIA ELETRONICA	S.PAULO	2024	AA0310	
2	900020 DIPOL - DEPTO DE INTELIGENCIA	DELEGACIA ELETRONICA	DELEGACIA ELETRONICA	S.PAULO	2024	AA0488	
3	900020 DIPOL - DEPTO DE INTELIGENCIA	DELEGACIA ELETRONICA	DELEGACIA ELETRONICA	S.PAULO	2024	AA0650	
4	900020 DIPOL - DEPTO DE INTELIGENCIA	DELEGACIA ELETRONICA	DELEGACIA ELETRONICA	S.PAULO	2024	AA0983	
6	20113 DECAP	DEL SEC.4P NORTE	13º D.P. CASA VERDE	S.PAULO	2024	AA1075	
7	900021 DIPOL - DEPTO DE INTELIGENCIA	DELEGACIA ELETRONICA	DELEGACIA ELETRONICA 1	S.PAULO	2024	AA1444	
8	900021 DIPOL - DEPTO DE INTELIGENCIA	DELEGACIA ELETRONICA	DELEGACIA ELETRONICA 1	S.PAULO	2024	AA1648	
9	900020 DIPOL - DEPTO DE INTELIGENCIA	DELEGACIA ELETRONICA	DELEGACIA ELETRONICA	S.PAULO	2024	AA1911	
10	900021 DIPOL - DEPTO DE INTELIGENCIA	DELEGACIA ELETRONICA	DELEGACIA ELETRONICA 1	S.PAULO	2024	AA2161	

A coluna HORA\_OCORRENCIA possui um caso específico. Algumas linhas, após a identificação da hora exata preenchida, possui um campo .0000000 após a hora, que ocorre um erro quando a coluna é convertida para hora.

Para solucionar esse caso, iremos dividir a coluna utilizando o delimitador ponto "." para separarmos esses valores.

Table.SplitColumn(Table.TransformColumnTypes(#"Duplicatas Removidas", {"HORA\_OCORRENCIA", type text}), "pt-BR", "HORA\_OCORRENCIA", Splitter.SplitTextByDelimiter(".", QuoteStyle.Csv), {"HORA\_OCORRENCIA.1", "HORA\_OCORRENCIA.2"})

ONAL_CIRC	NOME_DELEGACIA	NOME_MUNICIPIO	DATA_OCORRENCIA	HORA_OCORRENCIA.1	HORA_OCORRENCIA.2	DESCRICAO_APRESENTAC...	DATAHORA_REGISTRO	DATA_COMUNI
IO	01º D.P. SE	S.PAULO	30/12/2023	22:00	null	Pela Parte Interessada	01/01/2024	
IO	01º D.P. SE	S.PAULO	30/12/2023	null	null	Pela Parte Interessada	01/01/2024	
IO	01º D.P. SE	S.PAULO	30/12/2023	22:30	null	Pela Parte Interessada	01/01/2024	
IO	01º D.P. SE	S.PAULO	01/01/2024	null	null	Pela Parte Interessada	01/01/2024	
IO	01º D.P. SE	S.PAULO	30/12/2023	null	null	Pela Parte Interessada	01/01/2024	
IO	01º D.P. SE	S.PAULO	31/12/2023	23:00	null	Pela Parte Interessada	01/01/2024	
IO	01º D.P. SE	S.PAULO	31/12/2023	23:00	null	Pela Parte Interessada	01/01/2024	

Por padrão, o Power Query transformou essas colunas para o tipo texto. Basta retomar a primeira coluna que contém a hora para o formato de *tempo*, renomear a coluna de volta para HORA\_OCORRENCIA e apagar a segunda coluna criada.

Table.RenameColumns(#"Tipo Alterado1", {"HORA\_OCORRENCIA.1", "HORA\_OCORRENCIA"})

ONAL_CIRC	NOME_DELEGACIA	NOME_MUNICIPIO	DATA_OCORRENCIA	HORA_OCORRENCIA	HORA_OCORRENCIA.2	DESCRICAO_APRESENTAC...	DATAHORA_REGISTRO	DATA_COMUNI
JO	01º D.P. SE	S.PAULO	30/12/2023	22:00:00	null	Pela Parte Interessada	01/01/2024	
JO	01º D.P. SE	S.PAULO	30/12/2023	22:30:00	null	Pela Parte Interessada	01/01/2024	
JO	01º D.P. SE	S.PAULO	01/01/2024	00:20:00	null	Pela Parte Interessada	01/01/2024	
JO	01º D.P. SE	S.PAULO	30/12/2023	23:00:00	null	Pela Parte Interessada	01/01/2024	
JO	01º D.P. SE	S.PAULO	31/12/2023	23:00:00	null	Pela Parte Interessada	01/01/2024	
JO	01º D.P. SE	S.PAULO	01/01/2024	00:20:00	null	Pela Parte Interessada	01/01/2024	
JO	01º D.P. SE	S.PAULO	31/12/2023	15:30:00	null	Pela Parte Interessada	01/01/2024	

No próximo passo, iremos remover as seguintes colunas que não irão fazer parte da nossa análise. Alguns campos serão removidos por motivos de não fazerem sentido para o nosso tipo de análise, outros porque o campo possui somente 1 valor preenchido:

- Column1
- MES
- ANO
- LONGITUDE
- LATITUDE
- CEP
- LOGRADOURO\_VERSAO
- LOGRADOURO
- NUMERO\_LOGRADOURO
- CONT\_OBJETO
- DESCR\_MODO\_OBJETO
- DESCR\_TIPO\_OBJETO
- DESCR\_SUBTIPO\_OBJETO
- DESCR\_UNIDADE
- HORA\_OCORRENCIA.2
- VERSAO
- DATAHORA\_REGISTRO\_BO
- DATA\_COMUNICACAO\_BO
- DATAHORA\_IMPRESSAO\_BO
- FLAG\_STATUS

Ao analisarmos mais a fundo os dados, podemos perceber que as colunas HORA\_OCORRENCIA e DESCR\_PERIODO possuem algumas divergências. A coluna DESCR\_PERIODO quando está marcada como "Não Informado", a coluna HORA\_OCORRENCIA está preenchida com o horário do registro do BO, enquanto quando a coluna de hora está em branco, a coluna de descrição do período está preenchido.

Para ajustar esses dados, iremos criar uma coluna condicional com o nome de \*Periodo da Ocorrencia\*, no qual iremos criar uma condicional a partir da coluna das horas para ficar de acordo com a coluna DESCR\_PERIODO.

```
if Time.From([HORA_OCORRENCIA]) = null then [DESCR_PERIODO]
else if Time.From([HORA_OCORRENCIA]) < #time(6,0,0) then "De madrugada"
else if Time.From([HORA_OCORRENCIA]) < #time(12,0,0) then "Pela manhã"
else if Time.From([HORA_OCORRENCIA]) < #time(18,0,0) then "A tarde"
else if Time.From([HORA_OCORRENCIA]) < #time(23,59,59) then "A noite"
else [DESCR_PERIODO]
```

Basicamente, o código acima irá verificar se a coluna do horário não está preenchido. Se caso não estiver, ele irá preencher de acordo com a coluna do periodo, mas se caso tiver preenchido, iremos tratar da seguinte forma:

- 00:00 -> 05:59 = "De madrugada"
- 06:00 -> 11:59 = "Pela manhã"
- 12:00 -> 17:59 = "A tarde"
- 18:00 -> 23:59 = "A noite"

Dessa forma, conseguiremos fazer uma análise dos períodos.

Posteriormente, criei a coluna condicional "Ordem do Periodo da Ocorrência" que iremos utilizar para ordenar os períodos nos gráficos dentro do Power BI.

As colunas ficaram com o tipo de Texto e Número Inteiro, respectivamente.

Personalização Adicionada	⚙
Tipo Alterado2	⚙
Coluna Condisional Adicionada	⚙
Tipo Alterado3	⚙

Posteriormente, iremos tratar as seguintes colunas para termos os valores de acordo com a documentação oficial:

- AUTORIA\_BO - D para Desconhecida e C para Conhecida
- FLAG\_INTOLERANCIA - N para Não e S para Sim
- FLAG\_FLAGRANTE - N para Não e S para Sim
- FLAG\_ATO\_INFRACIONAL - N para Não e S para Sim

Valor Substituído	⚙
Valor Substituído1	⚙
Valor Substituído2	⚙
Valor Substituído3	⚙
Valor Substituído4	⚙
Valor Substituído5	⚙
Valor Substituído6	⚙

Após essas mudanças, iremos alterar dois valores na coluna QUANTIDADE\_OBJETO. Temos dois valores que provavelmente foram inseridos de forma errada pois fogem completamente do restante dos números.

Os números são o 1111111 e o 1351989, que iremos alterar para quantidade 1 para podermos aproveitar o BO registrado.

A última alteração que iremos realizar, será o nome S. PAULO para SÃO PAULO na coluna CIDADE, pois dessa forma fica mais fácil a leitura e mais fácil de filtrar a cidade a depender da análise.

Para finalizarmos, iremos criar mais dois campos personalizados, chamados de Dia da Semana e Dia da Semana - Ordem.

Dia da semana será para podermos validar qual dia da semana que o BO foi registrado e a ordem será para podermos ordenar de acordo com o dia da semana, iniciando no domingo.

```
Date.DayOfWeekName([DATA_OCORRENCIA_BO], "pt-BR")
```

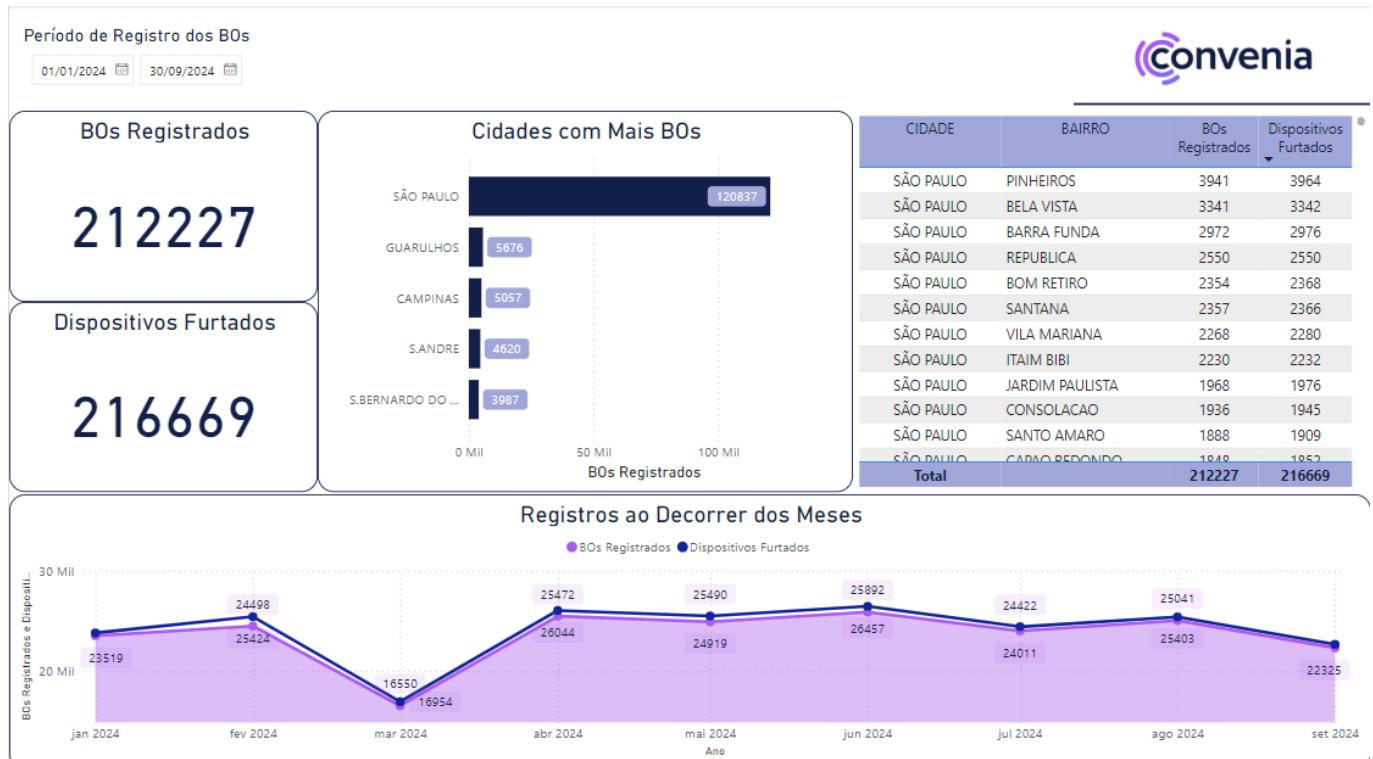
```
Date.DayOfWeek([DATA_OCORRENCIA_BO], Day.Sunday)
```

## Insights

Agora iremos mostrar alguns insights que podemos obter com os painéis criados.

*Obs: Iremos utilizar o período do dia 01/01/2024 ao dia 30/09/2024, período onde tivemos maior volume de registros.*

O primeiro painel abaixo, ficou da seguinte forma:



Nesse painel, podemos visualizar a quantidade de BOs que tivemos registrados no determinado período, juntamente com a quantidade de Dispositivos Furtados.

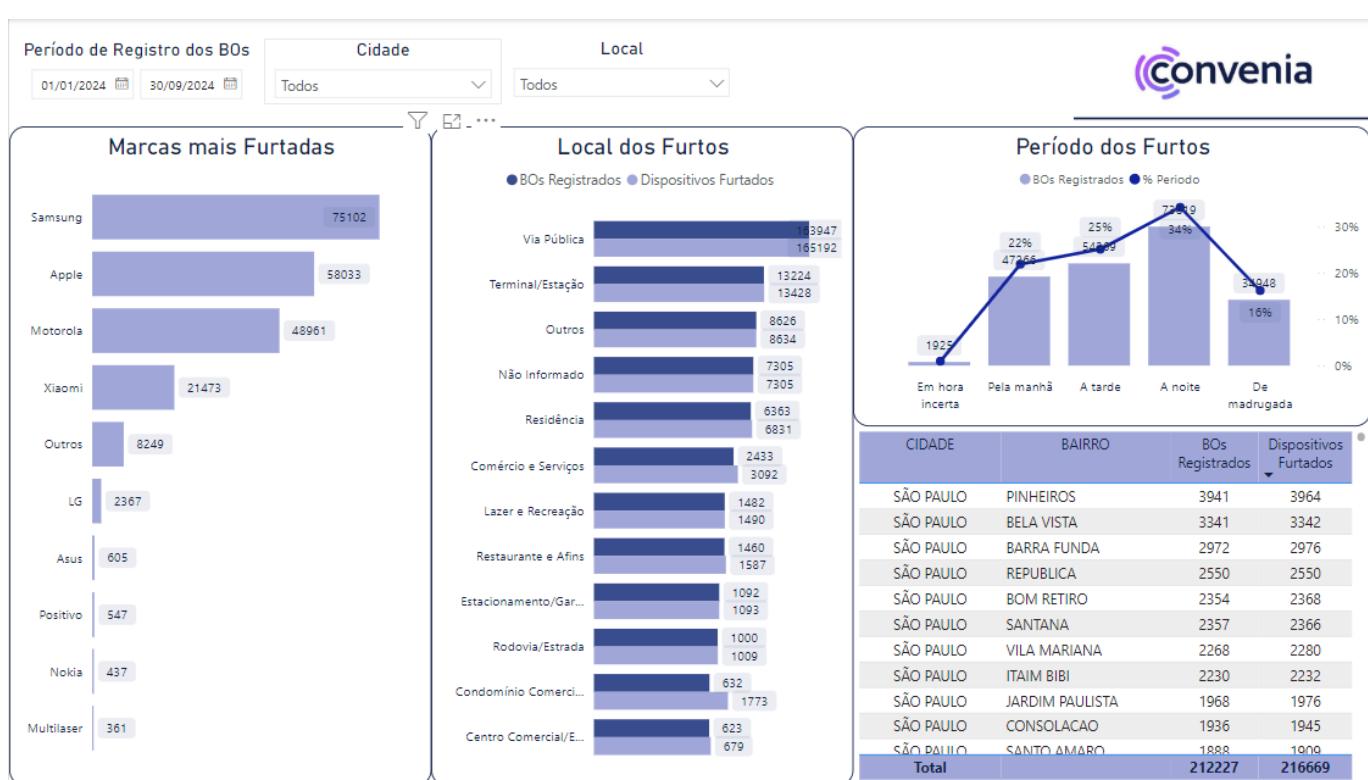
Podemos ver que a cidade de São Paulo é a cidade com mais BOs registrados. Cerca de 56% do registros que possuímos na nossa base de dados.

A direita, conseguimos ver visualizar os bairros onde temos mais BOs registrados e/ou dispositivos roubados. O bairros onde se encontram com mais BOs registrados, são os bairros onde temos um alto número de concentração de pessoas e possivelmente turistas, pois possuímos pontos de passeio e pontos turísticos nessa região.

O último gráfico apresenta uma evolução dos registros de BOs e dispositivos furtados. No mês de março tivemos uma quebra brusca do padrão dos dados. Esse fato pode se dar realmente por pouco roubo de dispositivo, ou por perca dos dados na hora da extração da informação.

Podemos visualizar que de Janeiro até Abril tivemos uma cresce no número de BOs registrados, que se manteve um pouco no padrão até o mês de Junho, no qual começou a ter uma leve queda até o mês de Setembro.

Hoje existem fontes na internet que informam que o índice de criminalidade em São Paulo caiu bastante em 2024 ao decorrer do ano, e também se comparado a anos anteriores.



Nesse painel, podemos ter mais detalhes sobre os BOs.

Conseguimos ver um rankeamento das marcas que foram mais furtadas. Podemos ver que as 3 primeiras são as marcas que são mais vendidas hoje no mercado brasileiro, cerca de 82% dos furtos.

O segundo gráfico podemos ver o tipo de local onde ocorreram mais furtos. Há também uma comparação com a quantidade de dispositivos roubados relacionados a cada localização. na vias públicas ou lugares públicos, podemos ver que temos quase a média de 1 pra 1 em relação a dispositivos roubados com a quantidade de BOs. Exceto lugares com comércio, como lojas e shoppings, onde a o número de dispositivos sobe um pouco mais em relação aos outros, que se dá por conta de roubo em alta quantidade de dispositivos.

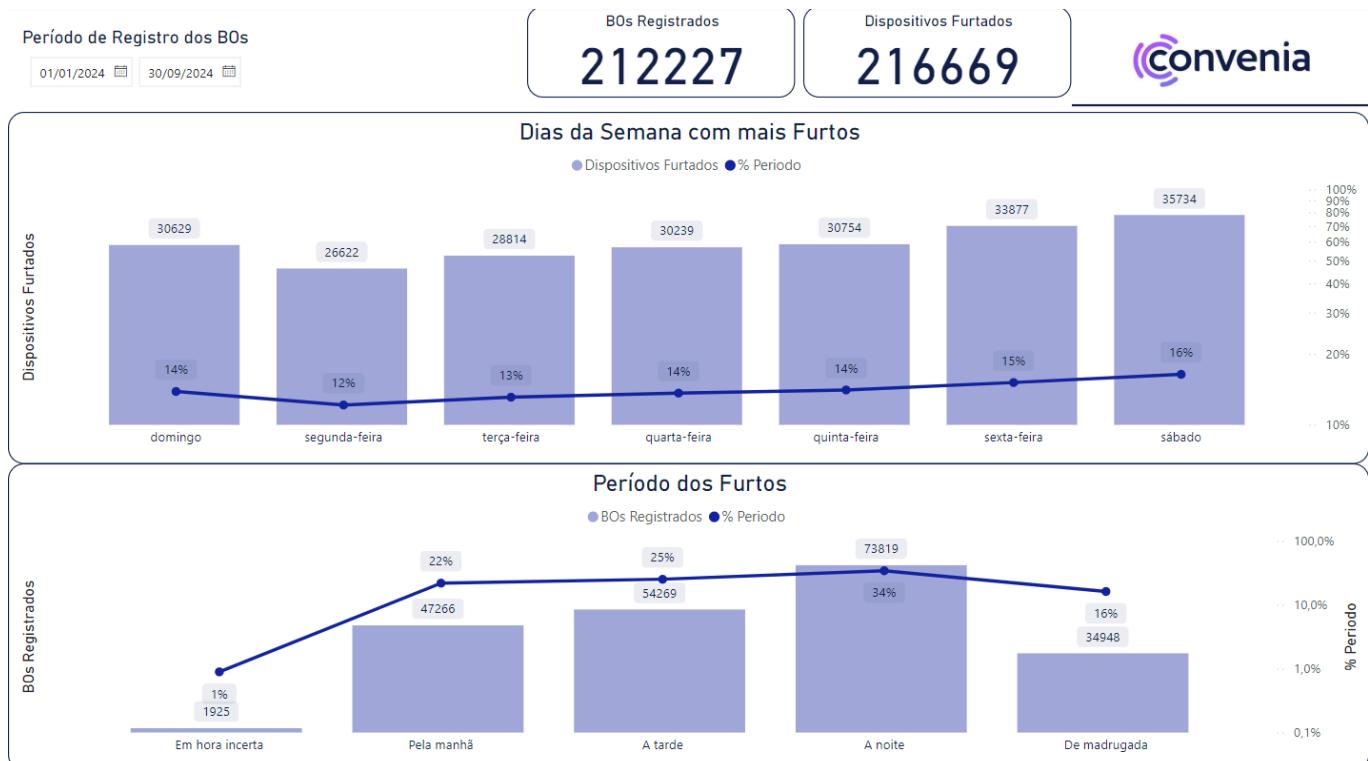
Podemos ver também os furtos por períodos do dia. Cerca de 60% dos furtos são realizados na parte da tarde ou noite, onde se tem uma grande concentração de pessoas nas ruas, seja para fins pessoais em visitar algum local, ou seja para retornar do trabalho.



Nesse painel, mostra os BOs registrados no qual a quantidade de celulares furtados é 30 ou mais.

Podemos constatar que os furtos em grandes quantidades foram realizados em estabelecimentos comerciais. E esses furtos foram realizados em grande parte na madrugada e na parte da manhã, muito devido ao horário em que o BO foi registrado.

Na listagem na parte inferior, podemos verificar melhor sobre esses BOs, como onde foi realizado, quantidade e qual a marca de telefone que foi mais roubada.



Nesse painel, conseguimos visualizar em qual dia da semana os roubos acontecem em grande quantidade.

Podemos constatar que os roubos são registrados em maior parte no final de semana, entre sexta-feira e domingo.

Esse número elevado se dá por conta do grande número de pessoas que saem aos finais de semana por motivos de lazer.



Neste último painel, podemos analisar mais alguns pequenos detalhes sobre os BOs.

Dos BOs registrados, 97% são registrados pela parte interessada, ou seja, a pessoa que foi roubada. Os demais foram divididas entre outras corporações.

Desses BOs, somente 10 tivemos casos de intolerância.

Os painéis de Flagrante e Menor Infrator, podemos concluir que a esmagadora minoria dos BOs tiveram o flagrante, e menos BOs ainda foram registrados por crimes cometido por menores de idade.

De todos os BOs registrados, quase 70% tiveram seu IMEI bloqueado, ou seja, grande parte desses celulares roubados serão inutilizados por quem realizou o roubo.

## Conclusão

Aqui cheamos ao final do desafio proposto pela Convenia.

### Sobre mim

Meu nome é Giovani Duarte, tenho 30 anos, sou Analista de Dados e sou de Belo Horizonte/MG.

## 🔗 Links

 [GITHUB](#)

 [LINKEDIN](#)