

# **Tarefa 06**

# **Banco de Dados Avançado**

Aluna: Letícia Trein Medeiros

# Escolha da Base

- O sistema **156** unificado da prefeitura de Curitiba recebe **informações e demandas** provindas dos cidadãos, ela é interessante e fornece informações de demandas da população de forma eficaz e possibilita o **direcionamento adequado** ao órgão responsável para melhor resposta.
- A presença de dados de **geolocalização** de bairros e ruas fornece informações específicas e exclusivas por **bairros e regionais**, que podem ser **diferentes** por questões socioeconômicas e **mapeamento social** da cidade.
- Por ser um sistema de atualização **diária**, é possível fazer um acompanhamento temporal das demandas
- Ter uma boa noção das demandas da população pode ser muito útil para tomada de **decisões políticas e sociais**, redirecionamento de recursos e tomada de decisões de prevenções.

# Detalhamento Base de Dados

- Sistema Integrado de Atendimento ao Cidadão - **SIAC 156**
- Descrição: Base de Dados contendo as solicitações geradas na Central 156, principal **canal de comunicação** entre o **cidadão** e a **Prefeitura Municipal de Curitiba**. Inclui todas as demandas direcionadas às Secretarias e Órgãos da Administração Municipal. Estes dados são oriundos do Sistema Integrado de Atendimento ao Cidadão - SIAC
- Órgão Responsável: Governo Municipal de Curitiba
- Frequência de Atualização: Diária
- Banco de Dados: Dados abertos da Cidade de Curitiba
- Formato: CSV
- Disponível em: <https://dadosabertos.c3sl.ufpr.br/curitiba/156/>
- Data de download: 01/10/2024 -> 2024-09-30\_156\_-\_Base\_de\_Dados.csv

# Descrição Atributos - Dicionário

- Tipo: Tipo do pedido (Solicitação, Denúncia, Reclamação,etc)
- Orgao: Orgão responsável pelo atendimento (Secretarias)
- DataCriacao: Data de criação do pedido (dd/mm/aaaa)
- Assunto: Nome do assunto do serviço
- Subdivisao: Nome da subdivisão do serviço (detalhamento do assunto)
- Situacao: Situação do pedido (Concluído, aberto, etc)
- Logradouro: Endereço de atendimento do pedido
- Bairro: Nome do bairro de atendimento do pedido
- Regional: Nome da regional de atendimento do pedido
- DataResposta: Data da resposta do atendimento (dd/mm/aaaa)
- Origem: Origem de entrada do pedido (Mobile, Telefone, Portal ou Chat)
- ID: Coluna criada para Primary Key (como valor único por tupla)

## Desafios para inserção - Pré-processamento: Tabela original

1 - Avaliação existência de valores nulos e colunas inválidas no dataset.

2 - Retirada as colunas com maior número de valores nulos

```
1 df_new = df.drop(columns=['Unnamed: 11'])  
2
```

	1	df.isnull().sum()
	✓	0.2s
Tipo		0
Orgao		0
DataCriacao		0
Assunto		0
Subdivisao		0
Situacao		0
Logradouro		0
Bairro		644
Regional		643
DataResposta		50588
Origem		0
Unnamed: 11		537526
dtype:	int64	

## 3 - Retirada de valores nulos

```
[43] ✓ 0.4s
```

4 - Criação de coluna ID para chave primária de valor único

# Desafios para inserção - Criação de tabela e importação

- 1 - Pelo formato original das colunas com DATAS, foi necessário classificar a variável como VARCHAR, uma vez que o formato DATE estava apresentando erro para inserção
- 2 - Foi necessário especificar o separador para ‘,’ para inserção corretas de dados.

```
CREATE TABLE ESPECIALIZACAO_LETICIA TREIN_OCORRENCIAS_156 (
    Tipo          VARCHAR(20),
    Orgao         VARCHAR(200),
    DataCriacao  VARCHAR(15),
    Assunto       VARCHAR(100),
    Subdivisao   VARCHAR(100),
    Situacao      VARCHAR(50),
    Logradouro   VARCHAR(100),
    Bairro        VARCHAR(30),
    Regional      VARCHAR(50),
    DataResposta  VARCHAR(15),
    Origem        VARCHAR(10),
    ID             INT PRIMARY KEY
);
```

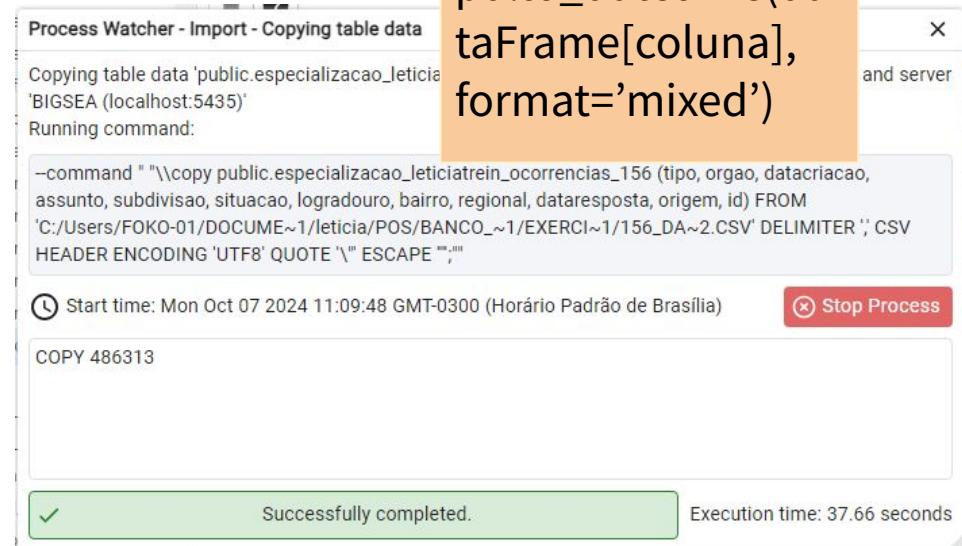


# Desafios para inserção - Criação de tabela e importação

1 - Pelo formato original das colunas com DATAS, foi necessário classificar a variável como VARCHAR, uma vez que o formato DATE estava apresentando erro para inserção

2 - Foi necessário especificar o separador para ‘,’ para inserção corretas de dados.

```
CREATE TABLE ESPECIALIZACAO_LETICIA TREIN_OCORRENCIAS_156 (
    Tipo      VARCHAR(20),
    Orgao     VARCHAR(200),
    DataCriacao VARCHAR(15),
    Assunto   VARCHAR(100),
    Subdivisao VARCHAR(100),
    Situacao  VARCHAR(50),
    Logradouro VARCHAR(100),
    Bairro    VARCHAR(30),
    Regional  VARCHAR(50),
    DataResposta VARCHAR(15),
    Origem    VARCHAR(10),
    ID        INT PRIMARY KEY
);
```



Sugestão recebida:  
comando  
pd.to\_datetime(da  
taFrame[coluna],  
format='mixed')

# Localização e número de Tuplas

- Caminho: postgiscwb.public.especializacao\_leticiatrein\_ocorrencias\_156
  - Número total de Tuplas:
    - 486.313

19    **SELECT COUNT(\*) FROM ESPECIALIZACAO LETICIA TREIN OCORRENCIAS\_156**  
20

Data Output    Messages    Notifications

---

count  
bigint

	count bigint
1	486313

# Perguntas Tarefa 3



Sugestão recebida:  
adicionar um  
comentário sucinto  
em cada uma das  
consultas

# Pergunta 1: Quantos pedidos foram criados por tipo?

SQL Input:

```
SELECT Tipo, COUNT(*) AS QuantidadePedidos  
FROM  
ESPECIALIZACAO_LETICIA TREIN_OCORRENCIAS_156  
GROUP BY Tipo;
```

	tipo character varying (20)	quantidadepedidos bigint
1	Denúncia	6074
2	Elogio	6621
3	Informação	741
4	Recadastro	2188
5	Reclamação	4519
6	Solicitação	424253
7	Sugestão	1317

Temos que o tipo com mais número de ocorrências seria “Solicitações”, seguido por “Reclamações”, “Elogios” e “Denúncias”

# Pergunta 2: Quais são os 10 principais assuntos das ocorrências no bairro Centro?

SQL Input:

```
SELECT Assunto, COUNT(*) AS  
QuantidadeOcorrencias  
  
FROM  
ESPECIALIZACAO_LETICIATREIN_OCORRE  
NCIAS_156  
  
WHERE Bairro = 'CENTRO'  
  
GROUP BY Assunto  
  
ORDER BY QuantidadeOcorrencias DESC  
  
LIMIT 10;
```

	assunto character varying (100)	quantidadeocorrencias bigint
1	Abordagem social de rua - Adulto e Idoso	5978
2	Trânsito	3770
3	Coleta	1566
4	Iluminação pública - Via pública	1553
5	Central Saúde Já	1401
6	Hospitais ensino/contratualizados	639
7	Limpeza	502
8	Passeio	460
9	Motoristas, cobradores e demais funcionários do transpor...	449

Foi observado um maior número de ocorrências relacionadas a abordagem social de rua, seguido por transito e coleta no bairro Centro.

# Pergunta 3: Quais os 15 bairros com mais Solicitações?

SQL Input:

```
SELECT Bairro, COUNT(*) AS QuantidadePedidos  
FROM  
ESPECIALIZACAO_LETICIA TREIN_OCORRENCIAS_1  
56  
GROUP BY Bairro  
  
ORDER BY QuantidadePedidos DESC  
LIMIT 15;
```

	bairro character varying (30)	quantidadepedidos bigint
1	CIDADE INDUSTRIAL	32328
2	CENTRO	22745
3	CAJURU	22152
4	SITIO CERCADO	20490
5	BOQUEIRAO	20431
6	UBERABA	16075
7	XAXIM	14930
8	NOVO MUNDO	14297
9	BAIRRO ALTO	12863
10	CAPAO RASO	12787

Vemos que os bairros com maior número de ocorrência foram: CIC, Centro, Cajuru, Sítio Cercado, Boqueirão, Uberaba.

# Pergunta 4: Quais são as Subdivisões do Assunto “Animais”?

SQL Input:

```
SELECT DISTINCT Subdivisao  
FROM  
ESPECIALIZACAO_LETICIATREIN_OCORR  
ENCIAS_156  
WHERE Assunto = 'Animais'
```

	subdivisao character varying (100)	locked
1	Cão sem guia e sem focinheira	
2	Cães bravos ou avançando em via pública - Fiscalizaç...	
3	Maus tratos - Animais silvestres	
4	Abelhas	
5	Dejetos gerados pelos animais de estimação	
6	Maus tratos - Animais domésticos	

As subdivisões de animais são: “Cão sem guia e sem focinheira”, “Cães bravos”, “Maus tratos - animais silvestres”, “Abelhas”, etc.

# Pergunta 5: Quais são os bairros que apresentam mais demandas relacionadas a violência?

SQL Input:

```
SELECT Bairro, COUNT(*) AS TotalDenuncias  
FROM  
ESPECIALIZACAO_LETICIA TREIN_OCORRENCIAS_156  
WHERE Assunto like '%Violência%'  
GROUP BY Bairro  
ORDER BY TotalDenuncias DESC  
LIMIT 10;
```

	bairro character varying (30)	totaldenuncias bigint
1	CENTRO	148
2	CIDADE INDUSTRIAL	128
3	BATEL	99
4	SITIO CERCADO	82
5	AGUA VERDE	58
6	TATUQUARA	54

Os bairros que apresentam maior número de ocorrências relacionadas a violência são: Centro, CIC, Batel, Sítio Cercado, Água Verte e Tatuquara

# Geoprocessamento

# Preparo dos dados

Para avaliação geoespacial da base de dados, foi adicionada um campo GEOM como multi polígono.

- `SELECT AddGeometryColumn('public', 'especializacao_letricatrein_ocorrencias_156', 'geom', 4326, 'MULTIPOLYGON', 2);`

Para georeferenciamento, foram utilizadas as geometrias dos bairros disponibilizadas na base de limites\_legais. Foi necessária adequação de dois bairros para total compatibilidade entre os dados

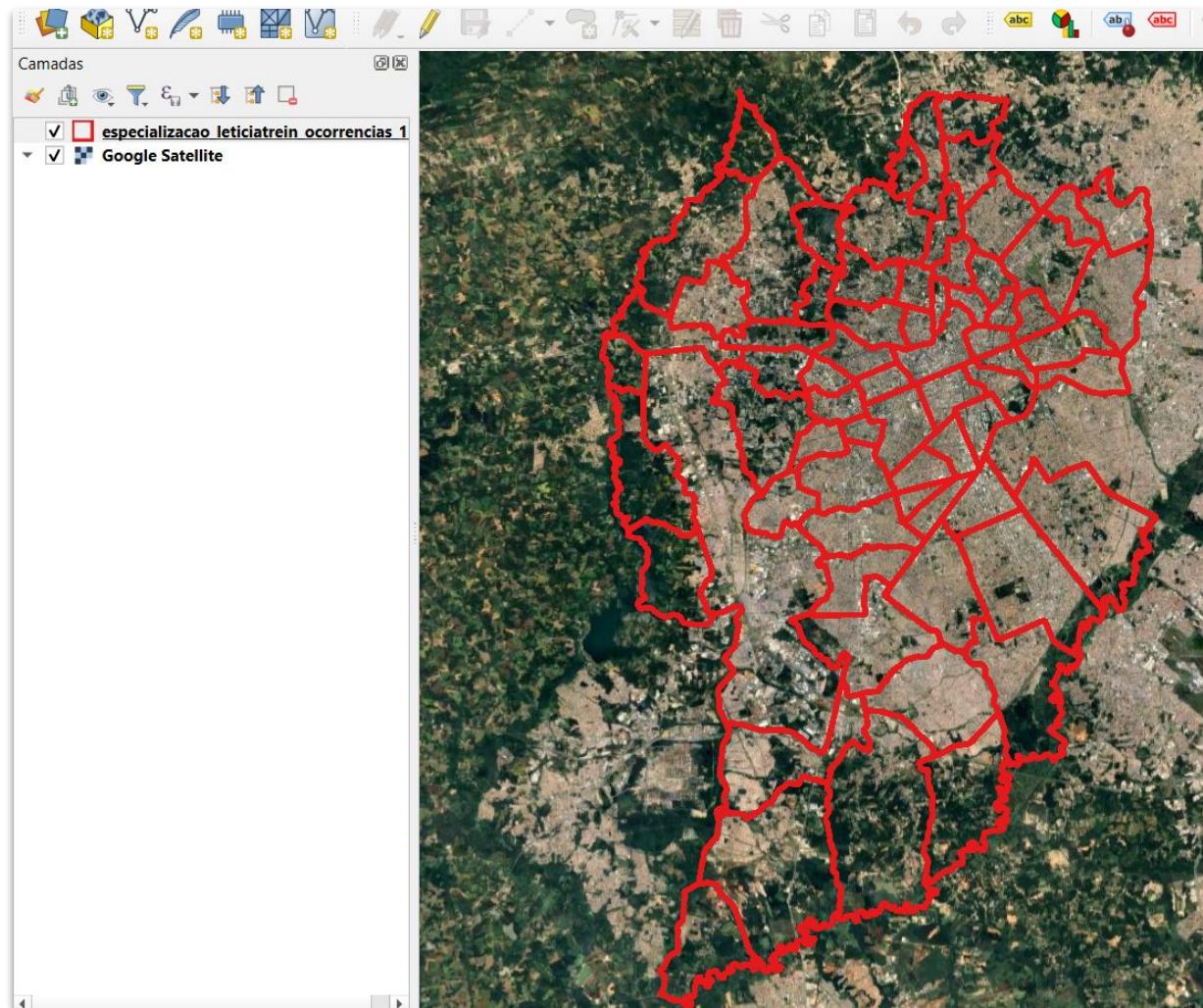
- `UPDATE public.especializacao_letricatrein_ocorrencias_156 SET bairro = CASE WHEN bairro = 'AGUA VERDE' THEN 'ÁGUA VERDE' WHEN bairro = 'REBOUCAS' THEN 'REBOUÇAS' ELSE bairro ;`
- `UPDATE public.especializacao_letricatrein_ocorrencias_156 b SET geom = a.geom FROM limites_legais.divisa_de_bairros a WHERE soundex(a.nome) = soundex(b.bairro);`

# QGIS

Com a utilização do complemento QuickMapService, foi plotado como fundo o Google Satellite

Os Dados do 156 foram plotados com a adição de camada ligada ao banco de dados

Foi necessária adequação do SRC para SIRGAS 2000 / UTM Zone 22S, tanto no projeto quanto nas camadas para visualização no mapa corretamente



# **Pergunta Tarefa 6:**

## Pergunta:

**Quais bairros registraram ocorrências de violência dentro de um raio de 500 metros de um módulo policial?**

### Dados Adicionais:

Para ter acesso a localização das guardas municipais, foram utilizados os dados disponibilizados no caminho: postgiscwb.seguranca.modulo\_guarda\_municipal

# Query Input

```
WITH violencia_ocorrencias AS ( SELECT DISTINCT bairro, geom FROM
especializacao_leticiatrein_ocorrencias_156 WHERE assunto LIKE '%Violência%'),
bairros_com_policia AS (SELECT DISTINCT v.bairro, v.geom FROM violencia_ocorrencias
AS v JOIN postgiscwb.seguranca.modulo_guarda_municipal AS d ON
ST_DWithin(v.geom, d.geom, 500)),
SELECT * FROM bairros_com_policia;
```

Gerenciador BD

Banco de Dados Esquema Tabela Schema

Importar camada/arquivo... Exportar para o arquivo...

Provedores Info Tabela Pré-visualizar aa (BIGSEA) X

Consulta salva aa Nome aa Salvar Excluir Carregar arquivo Salvar como arquivo

```
10 JOIN postgiscwb.seguranca.modulo_guarda_municipal AS d
11 ON ST_DWithin(v.geom, d.geom, 500) -- Verifica se o ponto de violência está dentro do bairro
12 )
13
14 SELECT *
15 FROM bairros_com_policia;
16
```

Executar 32 linhas, 6.526 segundos Criar uma vista Limpar Histórico de consultas

	bairro	geom
29	SITIO CERCADO	0106000020E61...
30	TABOAO	0106000020E61...
31	TINGUI	0106000020E61...
32	UMBARA	0106000020E61...

Carregar como uma nova camada

Coluna(s) com valores únicos bairro  Coluna de geometria geom Retornar colunas

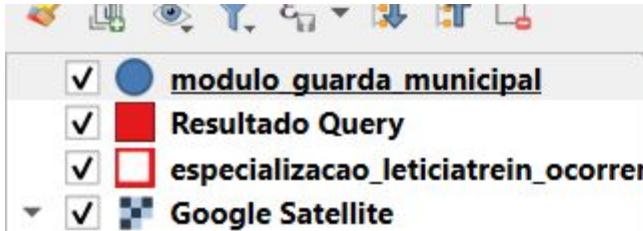
Nome da camada (prefixo) Resultado Query Definir filtro

Evite selecionar pela id da feição Carregar

Cancelar

Spa Can

This screenshot shows the QGIS Database Manager dialog for a PostgreSQL database named 'aa'. The main area displays a SQL query that joins a local table 'v' with a table 'modulo\_guarda\_municipal' from the schema 'seguranca' in the database 'postgiscwb'. The query uses the spatial function 'ST\_DWithin' to check if points in 'v' fall within the geometry of records in the joined table. The results show 32 rows, each containing a 'bairro' name and its corresponding geometry. Below the results, there are options to load the data as a new layer, define unique columns, and set a prefix for the new layer name. There are also checkboxes for filtering by feature ID and buttons for executing the query, clearing it, and saving it as a file.



Apenas um dos módulos  
policiais não apresentou  
registros relacionados à  
Violência ao redor do módulo  
(raio de 500m)

