

# Desafio Técnico – Analista de Dados

## Secretaria Municipal de Transportes

### Prefeitura da Cidade do Rio de Janeiro

#### Objetivo

Este desafio avalia a capacidade do candidato em associar viagens planejadas (**GTFS**) com viagens realizadas no transporte público do Rio de Janeiro. A análise busca medir a aderência da operação ao planejamento, identificando padrões e propondo métricas para avaliação da qualidade do serviço.

O candidato deverá desenvolver um código em **Python e SQL** que realize essa associação para todas as viagens realizadas no mês de **dezembro de 2024**, seguindo critérios estabelecidos.

---

## Descrição do Desafio

### 1. Combinar as tabelas relevantes

O candidato deverá realizar o cruzamento de diferentes tabelas do **GTFS** e dos dados operacionais para estruturar a base de análise. As tabelas envolvidas incluem:

- [rj-smtr.gtfs.trips](#) – Contém informações sobre as viagens planejadas;
- [rj-smtr.gtfs.frequencies](#) – Define os horários de partida das viagens ao longo do dia;
- [rj-smtr.gtfs.routes](#) – Fornece informações sobre os serviços (linhas) de ônibus;
- [rj-smtr.planejamento.calendario](#) – Define os **service\_ids** válidos para cada dia do mês;
- [rj-smtr.projeto\\_subsidio\\_sppo.viagem\\_completa](#) – Contém registros das viagens realizadas.

### 2. Filtrar os dados para incluir apenas ônibus e dias válidos

- Manter apenas as viagens (**trips**) cujo **service\_id** esteja presente na tabela [rj-smtr.planejamento.calendario](#), garantindo que a viagem está prevista para aquele dia;

- Filtrar apenas as linhas de ônibus, identificadas pelos **agency\_ids** que representam os **consórcios de ônibus** (Internorte, Intersul, Santa Cruz e Transcarioca).

### 3. Tratar os horários de início e fim das viagens (**start\_time** e **end\_time**)

- Os horários na tabela **frequencies** estão no formato **HH:MM:SS**, mas podem ultrapassar **23:59:59**, indicando viagens que iniciam ou terminam no dia seguinte;
  - Exemplo: para o dia **2025-02-12**, um horário **25:00:00** corresponde a **2025-02-13 01:00:00**;
- Ajustar esses horários para garantir que sejam corretamente interpretados no contexto do dia da viagem.

### 4. Gerar todas as partidas das viagens

- Para cada **trip** da tabela **frequencies**, criar os horários de partida com base nas colunas **start\_time**, **end\_time** e **headway\_secs** (intervalo entre partidas).

### 5. Associar viagens planejadas e realizadas

- A partir das **viagens partidas geradas anteriormente**, desdobrar as partidas e associá-las com as viagens registradas em [rj-smtr.projeto\\_subsidio\\_sppo.viagem\\_completa](#), respeitando uma tolerância máxima de 50% do intervalo entre partidas do mesmo serviço (linha) em viagens consecutivas.

### 6. Utilizar o feed correto do GTFS

- Identificar o **feed correto** para cada viagem utilizando os campos **feed\_start\_date** e **feed\_end\_date** das tabelas [rj-smtr.gtfs.feed\\_info](#) e [rj-smtr.projeto\\_subsidio\\_sppo.viagem\\_completa](#).

### 7. Tratar erros e dados incompletos

- O candidato deve sugerir abordagens para lidar com viagens sem correspondência exata.

### 8. Remover viagens duplicadas

- Garantir que cada viagem tenha um identificador único e eliminar registros redundantes.

## 9. Indicar uma métrica de avaliação

- O candidato deve propor um indicador para avaliar **quais serviços operam melhor (maior regularidade) e quais operam pior (menor regularidade)**;
- Deve-se levar em consideração a regularidade em diferentes faixas horárias (**a cada hora**) e **subfaixas horárias (a cada 15 minutos)**;
- Um serviço pode operar bem no **pico da manhã**, mas muito mal no **pico da tarde**, e isso deve ser levado em consideração na análise.

## 10. Gerar um ranking final de serviços

- O resultado final deve obrigatoriamente conter um **ranking por serviço para o mês**, que será um **output no formato CSV (UTF-8)**;
- O arquivo deve conter as seguintes colunas:
  - **posicao** – Posição no ranking (1º lugar = melhor serviço);
  - **servico** – Identificador do serviço avaliado (ex: '006');
  - **indicador** – Indicador de desempenho calculado pelo candidato;
- O ranking deve apresentar os serviços ordenados do melhor para o pior com base no **indicador desenvolvido pelo candidato**.

---


## Requisitos Adicionais

1. **Todas as consultas SQL devem ser executadas diretamente no notebook**
  - **Não é permitido importar arquivos CSV** gerados externamente;
  - As consultas SQL devem ser realizadas dentro do próprio **notebook Python**, utilizando bibliotecas como **pandas-gbq**, **google.cloud.bigquery**, **sqlalchemy**, entre outras;
  - Para candidatos com maior familiaridade com SQL, **recomenda-se o uso da biblioteca pandassql** para manipulação dos dados de maneira mais intuitiva.
2. **Otimização do uso do BigQuery**
  - O BigQuery cobra pelo volume de dados processados;
  - As tabelas usualmente são **particionadas por data** (o candidato deve verificar a coluna de particionamento), então o candidato deve **filtrar apenas os dados necessários** para o mês de **dezembro de 2024**, evitando consultas desnecessárias;
  - **Recomenda-se que, após realizar a consulta SQL, o candidato salve os dados em um arquivo CSV**, evitando consultas repetitivas e reduzindo custos no BigQuery. Isso será levado em consideração na avaliação.
3. **Referência para entendimento do GTFS**
  - Para compreender o relacionamento entre os elementos do GTFS, recomenda-se a leitura da documentação oficial:  
[!\[\]\(67ff022fd78f943b679992c2874bbfd1\_img.jpg\) \*\*GTFS Reference\*\*](#)

## Entrega

O candidato deve entregar um **arquivo .zip** em resposta ao e-mail contendo:

1. **Notebook Python (.ipynb)** com o código-fonte, explicações e documentação;
2. **Arquivo CSV (UTF-8) contendo o ranking final dos serviços;**
3. **Demais arquivos necessários para a execução do notebook** (caso existam).

 **Prazo de entrega: até 23/02/2025 (domingo) às 23h59.**

---

## Processo Seletivo

- Os candidatos que apresentarem as melhores análises serão selecionados para a etapa de entrevista, que ocorrerá entre 24/02/2025 e 27/02/2025;
  - Durante a entrevista, os candidatos selecionados deverão apresentar o notebook entregue, explicando a lógica utilizada e suas escolhas na análise.
- 

## Critérios de Avaliação

A avaliação será baseada nos seguintes pontos:

1. **Precisão da Associação** – A lógica da correspondência entre viagens planejadas e realizadas será analisada;
  - **Não haverá um algoritmo computacional para verificar a lógica;** todos os passos e decisões serão considerados;
2. **Clareza e Documentação** – O código deve ser bem estruturado e comentado;
3. **Otimização do Algoritmo** – Estratégias para lidar com um grande volume de dados serão avaliadas;
4. **Uso de Python e SQL** – O candidato deve demonstrar habilidades no uso dessas tecnologias;
5. **Eficiência no Uso do BigQuery** – Deve-se evitar consultas desnecessárias e filtrar apenas os dados essenciais.