FACULDADE IMPACTA DE SÃO PAULO

MBA ENGENHARIA DE DADOS

GILVANEIDE BATISTA DOS SANTOS RA: 2401013

ALY CAMILO SOARES RABELO RA: 2401082

RICARDO SOARES RA: 2400451

PROJETO 2: ANÁLISE DE DADOS DE VOO

SÃO PAULO

2024

GILVANEIDE BATISTA DOS SANTOS RA: 2401013

ALY CAMILO SOARES RABELO RA: 2401082

RICARDO SOARES RA: 2400451

PROJETO 2: ANÁLISE DE DADOS DE VOO

Projeto 2 apresentado a disciplina de Python for Data Engineer,

ministrada pela professora Carolina Zambelli Kamada

SÃO PAULO

2024

**Projeto 2 – Análise de dados de voo**

Este projeto tem como objetivo processar, limpar e realizar a engenharia de dados em uma base de dados de voos, utilizando Python. O pipeline desenvolvido realiza várias etapas, incluindo a padronização de dados, verificação de chaves, tratamento de nulos, e criação de novas colunas para análise. O resultado final é armazenado em um banco de dados SQLite.

**ATIVIDADES:**

1.Complete os projeto criando as funções:  
Função de validação de chaves - keys\_check (utils.py)

* Input da função: colunas chave
* Lógica: df[[cols\_chave]] == len(df)
* Output da função: Log/warning dizendo que os campos declarados como chave, não são chave
* Hint: Não é necessário fazer bater, ou seja, não é obrigatório retornar a condicional retornar True, concentrem-se na lógica.

Função para a criação de novos campos / Engenharia de Features - feat\_eng (app.py)

* tempo\_voo\_esperado -> datetime\_chegada - datetime\_partida
* dia\_semana -> [data do voo]dt.day\_of\_week hint: tem que ser do tipo datetime para funcionar
* horario -> classificar em manhã, tarde, noite e madrugada
* tempo\_voo\_hr -> tempo de voo que está em minutos, mas em horas
* atraso -> tempo\_voo\_hr  - tempo\_voo\_esperado
* flg\_status -> classificar quanto tempo de atraso é "ATRASO" e os demais casos "ON-TIME"

A lógica da implementação foi realizada em aula e está disponível no **notebook em anexo.** Link: **https://drive.google.com/file/d/1qxVjXqlewnAOdDzcHF\_JnC5UnystATNS/view?usp=sharing**   
2.No main inclua a execução de todas as etapas do processo (funções desenvolvidas)  
3.Inclua um logger para cada step do processo.  
4.Documente as funções desenvolvidas utilizando docstrings.  
5.Inclua um arquivo readme no projeto descrevendo o objetivo dele e a organização dos arquivos nas pastas.  
6.Inclua um arquivo de requirements.txt  
7.Siga a mesma estrutura de projeto que está no git da disciplina (link acima)  
  
**FORMATO DE ENTREGA:**  
1.Todos os projetos podem ser feitos em grupos de até 5 pessoas;  
2.Todos os integrantes do grupo devem ter um github com os códigos do projeto  
3.Entrega em documento word/pdf com:

* Nome dos integrantes do grupo
* Link do git de cada integrante, contendo o projeto
  + <https://github.com/camilo-8/Disciplina-Python-for-Data-Engineer/tree/main/Projeto2/db-pipeline>
  + <https://github.com/neide53/Disciplina-Python-for-Data-Engineer/tree/main/Projeto_2/db-pipeline>
* Print da execução: select dos primeiro registros da tabela do sqlite

Tela de computador com texto preto sobre fundo branco

Descrição gerada automaticamente