|  |  |
| --- | --- |
| **PARTE 1: SQL** | |
| **QUESTÕES** | **RESPOSTAS** |
| 1. **Consultas Básicas**   Dada a tabela `Vendas`:    | ID\_Venda | Data       | Cliente\_ID | Produto\_ID | Quantidade | Preco\_Total |  |----------|------------|------------|------------|------------|-------------|  | 1        | 2024-01-10 | 101        | 501        | 2          | 200.00      |  | 2        | 2024-01-15 | 102        | 502        | 1          | 100.00      |  | 3        | 2024-02-01 | 103        | 501        | 3          | 300.00      |  | 4        | 2024-02-05 | 101        | 503        | 4          | 400.00      |  | 5        | 2024-02-10 | 104        | 504        | 1          | 150.00      |    Escreva uma query SQL que retorna o total de vendas (em valor) por cliente. |  |
| 1. **Agrupamento e Ordenação**   Dada a tabela `Produtos`:    | Produto\_ID | Categoria  | Preco\_Unitario |  |------------|------------|----------------|  | 501        | Eletrônico | 100.00         |  | 502        | Móveis     | 100.00         |  | 503        | Eletrônico | 100.00         |  | 504        | Móveis     | 150.00         |  | 505        | Eletrodoméstico | 200.00     |    Escreva uma query SQL que retorna a soma total de vendas por categoria de produto, ordenada do maior para o menor valor. |  |
| 1. **Joins e Subqueries**   Usando as tabelas `Vendas` e `Produtos`, escreva uma query SQL que retorna o nome da categoria e a quantidade total vendida de produtos dessa categoria. |  |

|  |  |
| --- | --- |
| **PARTE 2: PYTHON** | |
| **QUESTÕES** | **RESPOSTAS** |
| 1. **Manipulação de Dados**   Você tem o seguinte DataFrame em Python:    ```python  import pandas as pd    data = {      'Cliente\_ID': [101, 102, 103, 101, 104],      'Produto\_ID': [501, 502, 501, 503, 504],      'Quantidade': [2, 1, 3, 4, 1],      'Preco\_Total': [200.00, 100.00, 300.00, 400.00, 150.00]  }    df = pd.DataFrame(data)  ```    Escreva um código Python que calcula o valor total das vendas por cliente. |  |
| 1. **Funções e Aplicações**   Escreva uma função em Python que recebe um DataFrame e retorna um novo DataFrame onde todos os valores da coluna `Preco\_Total` sejam multiplicados por 1.1 (um aumento de 10%). |  |
| 1. **Análise de Dados**   Dado o seguinte DataFrame `df`:    ```python  df = pd.DataFrame({      'A': [1, 2, 3, 4],      'B': [10, 20, 30, 40],      'C': [100, 200, 300, 400]  })  ```    Escreva um código Python para normalizar os valores de cada coluna no intervalo [0, 1].  --- |  |

|  |  |
| --- | --- |
| **PARTE 3: NOÇÕES DE ENGENHARIA DE DADOS** | |
| **QUESTÕES** | **RESPOSTAS** |
| 1. **ETL (Extract, Transform, Load)**   Explique como você realizaria um processo de ETL para integrar dados de várias fontes (por exemplo, bancos de dados SQL, arquivos CSV e uma API REST) em um Data Warehouse. |  |
| 1. **Modelagem de Dados**   Descreva o processo de normalização e quando não é recomendável utilizá-lo em um Data Warehouse. |  |
| 1. **Data Lakes vs Data Warehouses**   Explique a diferença entre um Data Lake e um Data Warehouse, destacando os principais casos de uso de cada um. |  |