**Transformação de Dados I – Básico**

**1. Agenda:**

1. Novas perguntas de negócio.

2. Planejamento da Solução.

3. Estrutura de Dados.

4. Transformação de Dados.

5. Exercícios Práticos.

**1. Novas perguntas do CEO**

1. Qual o número de imóveis por ano de construção?

2. Qual o menor número de quartos por ano de construção dos imóveis?

3. Qual o preço de compra mais alto por cada número de quartos?

4. Qual a soma de todos os preços de compra por cada número de quartos?

5. Qual a soma de todos os preços de compra pelo número de quartos e banheiros?

6. Qual o tamanho médio das salas dos imóveis por ano de construção?

7. Qual o tamanho mediano das salas dos imóveis por ano de construção?

8. Qual o desvio-padrão do tamanho das salas dos imóveis por ano de construção?

9. Como é o crescimento médio dos preços de compra dos imóveis, por dia e semana do ano?

10. Eu gostaria de olhar no mapa e conseguir identificar as casas com o maior preço.

2. Planejamento da solução:

2.1 Produto Final: O que eu vou entregar? Planilha, gráfico, modelo de ML, …

- Email + 2 anexos:

- Email: As respostas das perguntas.

- Pergunta | Resposta

- Anexo 01: Um dashboard com 3 gráficos.

- Anexo 02: A foto de um mapa 2.0 em .html

2.2. Ferramenta: Qual ferramenta usar? As ferramentas que eu tenho são suficientes para essa entrega?

- Python 3.8.0

- Jupyter Notebook

2.3 Processo: Como fazer?

1. Qual o número de imóveis por ano de construção?

- Contar o número de ids por ano de construção

2. Qual o menor número de quartos por ano de construção dos imóveis?

- Filtrar todos os imóveis por ano de construção e selecionar o menor número de quartos.

2.4 Processo: Como fazer?

1. Qual o número de imóveis por ano de construção?

- Contar o número de ids por ano de construção

2. Qual o menor número de quartos por ano de construção dos imóveis?

- Filtrar todos os imóveis por ano de construção e selecionar o menor número de quartos.

3. Qual o preço de compra mais alto por cada número de quartos?

- Filtrar todos os imóveis por número de quarto e selecionar o maior preço.

4. Qual a soma de todos os preços de compra por cada número de quartos?

- Filtrar todos os imóveis por número de quarto e somar todos os preços.

5. Qual a soma de todos os preços de compra pelo número de quartos e banheiros?

- Filtrar todos os imóveis por número de quarto e banheiro e somar

todos os preços.

6. Qual o tamanho médio das salas dos imóveis por ano de construção?

- Filtrar todos os imóveis por ano de construção e fazer a média

do tamanho das salas.

7. Qual o tamanho mediano das salas dos imóveis por ano de construção?

- Filtrar todos os imóveis por ano de construção e calcular a mediana do tamanho das salas.

8. Qual o desvio-padrão do tamanho das salas dos imóveis por ano de construção?

- Filtrar todos os imóveis por ano de construção e calcular o desvio padrão do tamanho das salas.

9. Como é o crescimento médio dos preços de compra dos imóveis, por dia e semana do ano?

- Filtrar todos os imóveis por data e calcular o preço médio.

- Procurar uma Biblioteca em Python que tenha uma Função que

desenhe um gráfico de linha.

- Aprender a usar a função e desenhar um a variação do preço médio por dia e semana do ano.

10. Eu gostaria de olhar no mapa e conseguir identificar as casas com o maior preço.

- Modificar o mapa da entrega anterior fazendo com que os pontos tenham o tamanho dependente do preço.

**3. As ferramentas para criar códigos em Python:**

- IDEs (Interface Development Environment)

- PyCharm

- VSCode

- Spyder

- JupyterLab

- Notebooks

- Jupyter Notebook

A principal Vantagem e Desvantagem:

- IDE’s é sempre necessário transformar TODOS os comandos em linguagem de máquina, todas as vezes que você executa os arquivos que contém o seu código.

- Notebook vantagens:

- É necessário transformar apenas os comandos escolhidos em linguagem de máquina.

- Podemos “rodar” comandos independentemente um dos outros.

- Organizar melhor nossos códigos.

- A análise exploratória dos dados fica mais fácil e organizada.

- Storytelling.

Desvantagens:

- Os notebooks precisam de um “ambiente” para funcionar. Esse “ambiente” possui:

- Editor de Texto (Notebook)

- O interpretador do Python.

- As bibliotecas que você está usando.

- O ambiente mais fácil para estudantes é o Anaconda

4. As estruturas de Dados em Python.

As 4 estruturas de dados mais usadas em Python são:

- Listas

- Dicionários

- Tuples

- Dataframes

4.1. Dicionários:

- É uma estrutura de dados que armazena informação em uma estrutura de chave-valor.

- Não aceitam valores duplicados ( mesmo par chave-valor ).

- Precisam de um nome.

- O dicionário tem a seguinte forma:

dict = {‘chave01’: valor01, ‘chave02’: valor02, ‘chave03’: valor03, ‘chave04’: valor04}

skirt = {'size': 'M', 'price': 139.90, 'color': 'black'}

skirt = {'size': 'M', 'price': 139.90, 'color': 'black', date:

’2020-01-01’}

skirt = {'size': 'M', 'price': 139.90, 'color': ['black', 'red', 'white']}

- Acessar os valores do dicionário via chave:

skirt['size'] -> ‘M’

skirt['color'] = ['black', 'red', 'white']

skirt['color'][0] -> ‘black’

- Como criar um dicionário vazio:

# skirt = {}

- Adicionar novos dados dentro de um dicionário:

skirt['category'] = 'bottom'

4.2. Dataframes:

- Armazenam dados na forma tabular com nomes nas linhas e colunas

- Precisam de um nome.

- Como criar um dataframe vazio:

data = pd.DataFrame()

- Como popular um dataframe a partir de um dicionário:

data = {'size': ['P', 'M', 'G'], 'price': [139.90, 89.90, 29.90], 'color': ['black', 'red', 'white']}

data = pd.DataFrame( data )

5. Transformação de Dados:

- Agrupamento

- Operações matemáticas

5.1. Agrupamento:

- Sequência de 3 tarefas: Split, Apply e Combine (Separa, Aplica e Combina)

- Agrupamento:

print( data[ data[ ‘bedrooms’ ] == 0 ].shape )

print( data[ data[ ‘bedrooms’ ] == 1 ].shape )

print( data[ data[ ‘bedrooms’ ] == 2 ].shape )

5.2. Operações:

- Com os dados agrupados, podemos realizar operações

matemáticas:

- Exemplos de operações matemática:

- Contagem.

- Mínimo.

- Máximo.

- Soma.

- Média.

- Mediana.

- Desvio Padrão.

**6. Executando o processo planejado**

1. Qual o número de imóveis por ano de construção?

- Contar o número de ids por ano de construção

2. Qual o menor número de quartos por ano de construção dos imóveis?

- Filtrar todos os imóveis por ano de construção e selecionar o menor número de quartos.

3. Qual o preço de compra mais alto por cada número de quartos?

- Filtrar todos os imóveis por número de quarto e selecionar o maior preço.

4. Qual a soma de todos os preços de compra por cada número de quartos?

- Filtrar todos os imóveis por número de quarto e somar todos os preços.

5. Qual a soma de todos os preços de compra pelo número de quartos e banheiros?

- Filtrar todos os imóveis por número de quarto e banheiro e somar todos os preços.

6. Qual o tamanho médio das salas dos imóveis por ano de construção?

- Filtrar todos os imóveis por ano de construção e fazer a média do tamanho das salas.

7. Qual o tamanho mediano das salas dos imóveis por ano de construção?

- Filtrar todos os imóveis por ano de construção e calcular a mediana do tamanho das salas.

8. Qual o desvio-padrão do tamanho das salas dos imóveis por ano de construção?

- Filtrar todos os imóveis por ano de construção e calcular o desvio-padrão do tamanho das salas.

9. Como é o crescimento médio preços de compra dos imóveis, por dia e semana do ano?

- Filtrar todos os imóveis por data e calcular o preço médio.

- Procurar uma Biblioteca em Python que tenha uma Função que

desenhe um gráfico de linha.

- Aprender a usar a função e desenhar um a variação do preço médio por dia e semana do ano.

10. Eu gostaria de olhar no mapa e conseguir identificar as casas com o maior preço.

- Modificar o mapa da entrega anterior fazendo com que os pontos tenham o tamanho dependente do preço.