

PLP- LABORATÓRIO - LISTA 4



Faça os exercícios desta lista mas a resposta **não** deverá ser entregue no Canvas.

1) Considere que você é o gerente de uma empresa que trabalha em diversas áreas do mercado de forma simultânea. Você recebeu a tarefa de criar um conjunto de dados (data frame) para analisar o desempenho de três filiais dessa empresa em diferentes áreas de negócio. Cada coluna representa uma filial e cada linha representa o desempenho da empresa (avaliado a partir de uma nota de 0 até 10) em uma de 5 diferentes áreas de negócio. Para realizar essa tarefa, siga as instruções abaixo:

a) Apresente o código em Python utilizado para criar um data frame composto pelos dados da tabela abaixo:

Área de Negócio	Filial A	Filial B	Filial C
Segurança	7,8	6,2	9,8
Financeiro	6,7	4,1	7,3
Transporte	2,8	9,2	3,4
Tecnologia	5,6	4,5	7,2
Marketing	8,3	7,5	9,1

b) Escreva o código em Python para:

- Inserir uma nova coluna ao data frame, chamada “Media Desempenho”, com a média de desempenho de cada filial em todas as áreas de negócio. Utilize a função `mean()`;
- Inserir uma nova coluna ao data frame, chamada “Melhor Desempenho”, com a área de negócio em que cada filial teve o melhor desempenho. Utilize a função `idxmax()`;
- Imprimir o total de desempenho para cada área considerando todas as filiais. Utilize a função `sum()`.
-

c) Escreva o código em Python para:

- Exportar o data frame resultante para um arquivo CSV com um nome de sua escolha;
- Efetuar a leitura do arquivo CSV gerado e o atribuir a um novo data frame.

RESPOSTA

2) Para responder as questões abaixo, utilize o arquivo “pedidos_de_asilo.xlsx”, disponibilizado no material de aula. Este arquivo guarda informações sobre o número de pedidos de asilo entre países em um determinado mês e ano (1999 até 2017), sendo composto pelas colunas:

- País de Origem;
- País de Destino;
- Ano;
- Mês;
- Número de Pedidos.

a) Utilizando a biblioteca Pandas, faça a leitura do arquivo e atribua seus dados para um data frame. Em seguida, utilize a função head() e o atributo shape para mostrar as dimensões e as linhas iniciais do data frame gerado.

b) Utilizando a biblioteca Matplotlib, crie um histograma mostrando a distribuição do número de pedidos de asilo. Você pode escolher o número de bins que achar adequado para a visualização dos dados, porém, devido ao grande número de linhas existente no data frame, sugere-se uma quantidade de bins maior ou igual a 100. O histograma deve possuir as seguintes características:

- Título = 'Distribuição do Número de Pedidos de Asilo';
- Legenda no eixo x = 'Número de Pedidos';
- Legenda no eixo y = 'Frequência';
- Deve possuir Grid;
- Limite de valores para o eixo x = (0, 100).

c) Crie um gráfico de dispersão que mostra o número de pedidos de asilo em relação ao ano. Cada ponto no gráfico representa um par de coordenadas (x = ano, y = número de pedidos de asilo). O gráfico de dispersão deve possuir as seguintes características:

- Título = 'Número de Pedidos de Asilo ao Longo dos Anos';
- Legenda no eixo x = 'Ano';
- Legenda no eixo y = 'Número de Pedidos';
- Deve possuir Grid.

RESPOSTA

3) Analisador de Estoque de Produtos

Uma loja deseja um sistema que analise seu estoque e identifique quais produtos estão abaixo do nível mínimo necessário. O usuário deve fornecer uma lista de dicionários onde cada dicionário representa um produto com as seguintes informações:

- nome: nome do produto
- quantidade: quantidade atual em estoque
- minimo: quantidade mínima necessária

Tarefa:

Escreva uma função `produtos_abaixo_minimo(estoque)` que receba uma lista de produtos e retorne uma lista com os nomes dos produtos que estão abaixo do mínimo.

Exemplo de entrada:

```
estoque = [  
    {"nome": "Caneta", "quantidade": 10, "minimo": 15},  
    {"nome": "Lápis", "quantidade": 20, "minimo": 10},  
    {"nome": "Borracha", "quantidade": 5, "minimo": 8},  
]
```

Saída esperada:

```
["Caneta", "Borracha"]
```

RESPOSTA

4) Cálculo de Média Ponderada

Uma escola deseja calcular a média final dos alunos considerando pesos diferentes para cada nota. O usuário deve fornecer uma lista de tuplas onde cada tupla representa `(nota, peso)`.

Tarefa:

Crie uma função `media_ponderada(notas_pesos)` que recebe uma lista de `(nota, peso)` e retorna a média ponderada.

Exemplo de entrada:

`notas = [(7.5, 2), (8.0, 3), (6.5, 5)]`

Saída esperada:

7.15

RESPOSTA**5) Caminho Ótimo em uma Matriz**

Dada uma matriz $n \times m$ representando um tabuleiro onde cada célula tem um custo para ser atravessada, um robô começa no canto superior esquerdo (posição $[0,0]$) e deseja alcançar o canto inferior direito $([n-1, m-1])$. O robô pode mover-se apenas para a direita ou para baixo.

Tarefa:

Implemente a função `caminho_minimo(matriz)`, que retorna o custo mínimo para alcançar o destino.

Exemplo de entrada:

```
matriz = [
    [1, 3, 1],
    [1, 5, 1],
    [4, 2, 1]
]
```

Saída esperada:

Caminho: $1 \rightarrow 3 \rightarrow 1 \rightarrow 1 \rightarrow 1 \rightarrow 1$ (soma = 7)

(Caminho percorrido: $1 \rightarrow 3 \rightarrow 1 \rightarrow 1 \rightarrow 1$). Veja abaixo grifado de amarelo:

Digite a matriz:

	Col 0	Col 1	Col2	
	(0,0)	\rightarrow	\rightarrow	\rightarrow
Linha 0 \rightarrow	1	3	1	\downarrow
Linha 1 \rightarrow	1	5	1	\downarrow

Linha 2 → 4 2 1 (2,2)

RESPOSTA

6) Validação de Expressões Matemáticas

Escreva um validador de expressões matemáticas que verifica se os parênteses, colchetes e chaves estão corretamente balanceados.

Tarefa:

Implemente a função `expressao_valida(expressao)`, que recebe uma string contendo uma expressão matemática e retorna `True` se estiver bem formada, ou `False` caso contrário.

Exemplo de entrada:

```
expressao = "{[()]}"
```

Saída esperada:

True

Exemplo de entrada inválida:

```
expressao = "{[()]"
```

Saída esperada:

False

RESPOSTA