**01: Desenvolver uma Aplicação para o Cálculo do Índice de Massa Corporal (IMC)**

**Descrição:**

Implemente uma aplicação que realize o cálculo do Índice de Massa Corporal (IMC) de um indivíduo com base nas informações fornecidas pelo usuário. Siga os critérios descritos abaixo para garantir que a aplicação funcione corretamente e atenda às expectativas.

**Requisitos:**

**Solicitação de Dados:**

* + Ao iniciar a execução do script, a aplicação deve solicitar o nome do usuário.
  + Em seguida, a aplicação deve solicitar a altura do usuário em centímetros.
  + Por fim, solicite ao usuário que informe seu peso em quilogramas.

**Conversão de Dados:**

* + Certifique-se de converter os valores de altura e peso recebidos para o tipo de dado float.
  + Converta a altura de centímetros para metros (dividindo o valor por 100).

**Cálculo do IMC:**

* + A aplicação deve calcular o Índice de Massa Corporal (IMC) utilizando a fórmula:

Uma imagem contendo Texto

Descrição gerada automaticamente

* + **Classificação do IMC:**
  + Após calcular o IMC, a aplicação deve classificar o resultado de acordo com as seguintes faixas:
    - **IMC < 16:** Baixo peso muito grave
    - **IMC entre 16 e 16,99:** Baixo peso grave
    - **IMC entre 17 e 18,49:** Baixo peso
    - **IMC entre 18,50 e 24,99:** Peso normal
    - **IMC entre 25 e 29,99:** Sobrepeso
    - **IMC entre 30 e 34,99:** Obesidade grau I
    - **IMC entre 35 e 39,99:** Obesidade grau II
    - **IMC ≥ 40:** Obesidade grau III

**Exibição do Resultado:**

* + Ao finalizar o cálculo e a classificação, a aplicação deve exibir a seguinte mensagem para o usuário: "<Nome> possui um Indice de Massa Corporal igual a <IMC>, sendo classificado como: <Classificação>"\text{"<Nome> possui um Índice de Massa Corporal igual a <IMC>, sendo classificado como: <Classificação>"}"<Nome> possui um Iˊndice de Massa Corporal igual a <IMC>, sendo classificado como: <Classificac¸​a˜o>"
    - Onde <Nome>, <IMC> e <Classificação> devem ser substituídos pelos valores calculados pela aplicação.

**02: Desenvolver uma Aplicação para Classificação de Faixa Etária**

**Descrição:**

Implemente uma aplicação que seja capaz de identificar e classificar a faixa etária de uma pessoa com base na idade informada pelo usuário. A aplicação deve seguir as instruções abaixo e oferecer maior complexidade em termos de lógica de programação.

**Requisitos:**

**Solicitação de Dados:**

* + Ao iniciar a execução do script, a aplicação deve solicitar o nome do usuário.
  + Em seguida, solicite ao usuário que informe sua idade em anos.

**Validação de Dados:**

* + A aplicação deve verificar se a idade informada é um número inteiro positivo. Caso contrário, exiba uma mensagem de erro e solicite a entrada novamente até que um valor válido seja fornecido.

**Classificação da Faixa Etária:**

* + Com base na idade informada, a aplicação deve classificar o usuário em uma das seguintes faixas etárias:
    - **0 ≤ idade < 15:** Criança
    - **15 ≤ idade < 30:** Jovem
    - **30 ≤ idade < 60:** Adulto
    - **idade ≥ 60:** Idoso
  + **Desafio Adicional:** Utilize uma estrutura de decisão mais avançada, como o uso de funções, listas ou dicionários, para realizar a classificação da faixa etária em vez de simples estruturas condicionais (como if-else). Considere também a possibilidade de novas faixas etárias serem adicionadas futuramente.

**Exibição do Resultado:**

* + Após a classificação, a aplicação deve exibir a seguinte mensagem: "<Nome> tem <idade> anos e eˊ classificado como: <Classificac¸a˜o>"\text{"<Nome> tem <idade> anos e é classificado como: <Classificação>"}"<Nome> tem <idade> anos e eˊ classificado como: <Classificac¸​a˜o>"
  + Onde <Nome>, <idade>, e <Classificação> devem ser substituídos pelos valores informados e calculados pela aplicação.

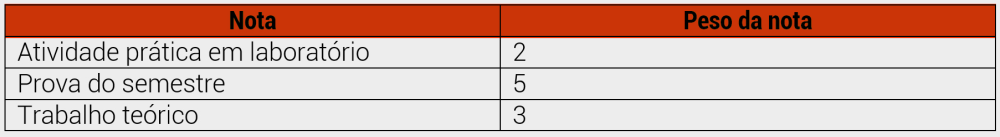
**Melhorias Extras (Opcional):**

* + Permita que o usuário insira várias idades consecutivamente, classificando cada uma e exibindo os resultados em sequência.
  + Adicione uma opção para o usuário encerrar a aplicação a qualquer momento.

**03: Desenvolvimento de uma Aplicação para Cálculo da Média Final de uma Disciplina**

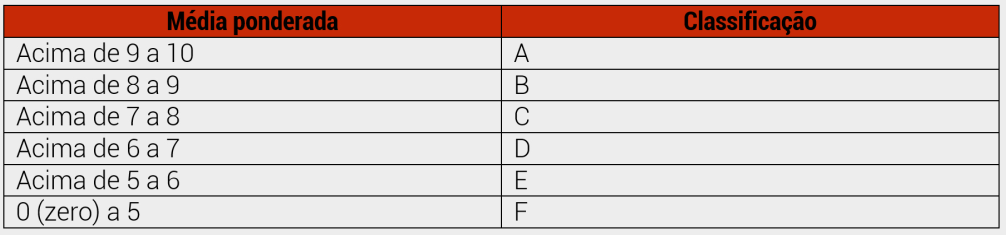
**Descrição:**

Um aluno está enfrentando dificuldades para compreender como é calculada a média final em uma determinada disciplina, composta por três notas com pesos específicos. Sua tarefa é desenvolver uma aplicação em JavaScript que permita ao usuário inserir as três notas, calcule a média ponderada e, em seguida, classifique o desempenho do aluno com base na média obtida.



**Requisitos:**

* **Entrada de Dados:**
  + A aplicação deve solicitar ao usuário a inserção das três notas do aluno, denominadas como nota1, nota2 e nota3.
  + Solicite também os pesos correspondentes a cada nota, denominados como peso1, peso2 e peso3.
* **Validação de Dados:**
  + A aplicação deve validar as entradas para garantir que as notas estejam entre 0 e 10 e que os pesos sejam números inteiros positivos. Caso contrário, o sistema deve solicitar a entrada novamente até que valores válidos sejam fornecidos.
* **Cálculo da Média Ponderada:**
  + A média ponderada deve ser calculada utilizando a fórmula:
* ((peso1 \* nota1) + (peso2\*nota2)+(peso3\*nota3))/ (soma dos pesos)
  + Certifique-se de que a aplicação calcule corretamente a soma dos pesos e o valor final da média.
* **Classificação da Média:**
  + Após calcular a média ponderada, a aplicação deve classificar o desempenho do aluno conforme a tabela de classificação abaixo:



* **Exibição do Resultado:**
  + Ao término, a aplicação deve exibir uma mensagem ao usuário no seguinte formato: "A média ponderada do aluno eˊ <Média>, sendo classificada como: <Classificac¸a˜o>"\text{"A média ponderada do aluno é <Média>, sendo classificada como: <Classificação>"}"A meˊdia ponderada do aluno eˊ <Meˊdia>, sendo classificada como: <Classificac¸​a˜o>"
  + Onde <Média> é o valor calculado e <Classificação> é a categoria correspondente conforme a tabela acima.
* **Melhorias Extras (Opcional):**
  + Permita ao usuário inserir novas notas e pesos para calcular diferentes médias sem precisar reiniciar a aplicação.

**04: Desenvolvimento de uma Aplicação para Cálculo de Frete e Rastreamento de Produtos**

**Descrição:**

Uma empresa do setor metalmecânico está conduzindo uma pesquisa para determinar o custo de transporte de seus produtos para clientes em diferentes estados. Após a pesquisa, uma transportadora foi selecionada, e o cliente tem a opção de escolher se deseja o rastreamento de sua entrega. Sua tarefa é desenvolver uma aplicação que calcule o valor total do frete, considerando diversos fatores, e a inclusão opcional do rastreamento.

**Requisitos:**

**Entrada de Dados:**

* + A aplicação deve solicitar ao cliente o número total de peças a serem transportadas.
  + Pergunte ao cliente se deseja rastreamento na entrega (S-Sim ou N-Não). Se a resposta for "Sim", adicione uma taxa fixa de R$ 200,00 ao custo total do frete.
  + Solicite a região de entrega, que pode ser:
    - **Região 1:** Sudeste
    - **Região 2:** Sul
    - **Região 3:** Centro-Oeste
  + Solicite a distância total em quilômetros até o destino, pois será necessário calcular o consumo de combustível.

**Cálculo do Frete:**

* + O valor do frete será influenciado por três fatores: número de peças, região de entrega e distância.
  + **Para até 1.000 peças:**
    - O valor do frete será calculado por peça, de acordo com a tabela de preços por região (ver tabela abaixo).
  + **Para mais de 1.000 peças:**
    - As primeiras 1.000 peças serão cobradas pelo valor normal da região.
    - As peças que excederem 1.000 terão um desconto no valor por peça, conforme a tabela abaixo.
    - Exemplo de cálculo:
      * Para 1.200 peças na Região 1 (Sudeste), o valor do frete será calculado como: (1.000×1,20)+(200×1,056)=R$1.411,20(1.000 \times 1,20) + (200 \times 1,056) = R\$ 1.411,20(1.000×1,20)+(200×1,056)=R$1.411,20
  + **Tabela de Valores de Frete por Peça:**
    - **Região 1 (Sudeste):** R$ 1,20 por peça
    - **Região 2 (Sul):** R$ 1,10 por peça
    - **Região 3 (Centro-Oeste):** R$ 1,15 por peça
  + **Desconto por Peça (para peças que excederem 1.000 unidades):**
    - **Região 1 (Sudeste):** 12% de desconto
    - **Região 2 (Sul):** 10% de desconto
    - **Região 3 (Centro-Oeste):** 8% de desconto

**Cálculo do Consumo de Combustível:**

* + A distância informada pelo cliente deve ser utilizada para calcular o consumo de combustível. A aplicação deve considerar que para cada quilômetro percorrido será consumido 1 litro de combustível.

**Exibição do Resultado:**

* + Ao final, a aplicação deve exibir o custo total do frete, incluindo ou não a taxa de rastreamento, no seguinte formato: "O valor total do frete eˊ de R$ <Valor Total>, incluindo <Valor do Combustıˊvel> para combustıˊvel e <Valor de Rastreamento> para rastreamento (se aplicaˊvel)."\text{"O valor total do frete é de R\$ <Valor Total>, incluindo <Valor do Combustível> para combustível e <Valor de Rastreamento> para rastreamento (se aplicável)."}"O valor total do frete eˊ de R$ <Valor Total>, incluindo <Valor do Combustıˊvel> para combustıˊvel e <Valor de Rastreamento> para rastreamento (se aplicaˊvel)."
  + Onde <Valor Total>, <Valor do Combustível>, e <Valor de Rastreamento> são valores calculados pela aplicação.

**Melhorias Extras (Opcional):**

* + Adicione uma funcionalidade para que o cliente possa calcular o frete para múltiplos destinos ou volumes em uma única sessão.

**05: Desenvolvimento de um aplicação para Informatização da Folha de Pagamento em uma Rede Hoteleira**

**Descrição:**

Uma rede hoteleira, com presença em diversas regiões do Brasil, deseja informatizar sua folha de pagamento para atender às especificações regionais e funcionais. Sua tarefa é desenvolver um pseudocódigo que automatize o cálculo do salário dos funcionários, levando em consideração o valor do salário-mínimo, o turno de trabalho, a categoria do funcionário, e outros fatores.

**Requisitos:**

1. **Entrada de Dados:**
   * O pseudocódigo deve solicitar ao usuário as seguintes informações para cada funcionário:
     + **Código do funcionário** (número inteiro)
     + **Número de horas trabalhadas no mês** (número inteiro)
     + **Turno de trabalho** (caractere: "M" para matutino, "V" para vespertino, "N" para noturno)
     + **Categoria do funcionário** (caractere: "F" para funcionário, "G" para gerente)
     + **Valor do salário-mínimo** (número real), uma vez que ele pode variar de estado para estado.
2. **Cálculo do Valor da Hora Trabalhada:**
   * Utilize o comando de seleção múltipla (escolha ou equivalente) para determinar o valor da hora trabalhada, conforme a combinação da categoria e do turno de trabalho, utilizando a seguinte lógica:
     + **Funcionário (F):**
       - **Matutino (M):** Valor da hora = 0,1 × salário-mínimo
       - **Vespertino (V):** Valor da hora = 0,12 × salário-mínimo
       - **Noturno (N):** Valor da hora = 0,15 × salário-mínimo
     + **Gerente (G):**
       - **Matutino (M):** Valor da hora = 0,2 × salário-mínimo
       - **Vespertino (V):** Valor da hora = 0,25 × salário-mínimo
       - **Noturno (N):** Valor da hora = 0,3 × salário-mínimo

**Cálculo do Salário Inicial:**

* + Calcule o salário inicial do funcionário multiplicando o valor da hora trabalhada pelo número de horas trabalhadas no mês.

**Cálculo do Auxílio-Alimentação:**

* + Com base no salário inicial calculado, determine o valor do auxílio-alimentação utilizando o comando de seleção composto encadeado, conforme as regras a seguir:
    - **Salário inicial ≤ 2 × salário-mínimo:** Auxílio-alimentação = 15% do salário inicial
    - **Salário inicial > 2 × salário-mínimo e ≤ 4 × salário-mínimo:** Auxílio-alimentação = 10% do salário inicial
    - **Salário inicial > 4 × salário-mínimo:** Auxílio-alimentação = 5% do salário inicial

**Cálculo do Salário Final:**

* + O salário final deve ser a soma do salário inicial com o valor do auxílio-alimentação.

**Exibição dos Resultados:**

* + O pseudocódigo deve exibir os seguintes dados como saída:
    - Código do funcionário
    - Número de horas trabalhadas
    - Valor da hora trabalhada
    - Salário inicial
    - Auxílio-alimentação
    - Salário final (salário inicial + auxílio-alimentação)

**Melhorias Extras (Opcional):**

* Adicione a capacidade de calcular e armazenar os resultados de vários funcionários em uma única execução.
* Implemente a funcionalidade para exibir o salário total pago pela empresa para todos os funcionários processados.