

## Lista de Exercícios

### Tema 07. Atributos e métodos estáticos.

#### Recomendações:

- Utilizando a linguagem de programação da sua preferência (Java, Python ou ambas) implemente as classes especificadas nos exercícios a seguir.
  - Utilize apropriadamente os conceitos de atributos e métodos estáticos discutidos na sala de aulas
  - Crie em cada caso uma classe cliente, ou aplicação simples onde mostre o uso da classe implementada.
1. Usando como base a classe **Contato** desenvolvida nas listas anteriores, introduza um atributo para armazenar o total de contatos que existem na agenda. Crie um método que mostre o número de contatos na tela.
  2. Usando como base a classe **Time** implementada nas listas anteriores introduza os seguintes métodos estáticos na classe:
    - **Time diffTime(Time t1, Time t2)** calcula e retorna a diferença de tempo entre as horas t1 e t2.
    - **Time diffTime(Time t)** calcula e retorna a diferença de tempo entre a hora t1 e a hora corrente do computador, pesquise como recuperar a hora do computador na linguagem escolhida.
    - **boolean isAm(Time t)**, retorna verdadeiro se a hora t for anterior ao meio-dia
    - **boolean isPm(Time t)**, retorna verdadeiro se a hora t for anterior ao meio-dia
  3. Usando como base classe **Data** implementada nas listas anteriores adicione os seguintes métodos estáticos:
    - **int howManyDaysUntilEndYear(Data d)** calcula e retorna a quantidade de dias desde a data d até o final do ano da data d.
    - **int howManyDaysUntilNextMonth(Data d)** calcula e retorna a quantidade de dias desde a data d o dia primeiro do mês seguinte.
    - **boolean isBisexto(Data d)** retorna verdadeiro se o ano da Data d for bisesto.

- **String dayOfWeek(Data d)**, retorna uma String com o dia da semana da data d
  - **String dayToPrintShort(Data d)**, retorna uma String representando a data d no formato dd/mm/yyyy
  - **String dayToPrintLong(Data d)**, retorna uma String representando a data d no formato dd month de yyyy (1 de janeiro de 2023).
4. Crie uma biblioteca de métodos estáticos para manipular instâncias da classe Ponto2D construída anteriormente, considere os métodos abaixo:
- **float distance(Ponto2D P1, Ponto2D P2)**, retorna a distância entre os pontos P1 e P2
  - **float distance(Ponto2D P)**, calcula a distância entre o ponto P origem de coordenadas
  - **int quadrant(Ponto2D P)**, retorna o quadrante do ponto P (1, 2, 3, ou 4), ou zero se estiver sobre os eixos.
5. Crie uma biblioteca de funções para conversão de unidades de peso. A biblioteca deve implementar conversões entre as unidades: gramas, quilogramas, libras (*pound*) e onças (*ounce*). Pesquise na internet quais as fórmulas para implementar as mudanças de unidades em cada caso. Sua biblioteca deve conter 12 métodos um para cada conversão.