

1. Faça um código em Python que permita armazenar uma data composta por dia, mês e ano. A seguir o código deve implementar uma operação para verificar se a data é válida.
2. Utilizando o primeiro código faça uma função que recebe como entrada uma data e retorna como saída 1 caso o ano seja bissexto e 0 caso contrário;
3. Dando continuidade aos exercícios 1 e 2, acrescente uma função que recebe como entrada um ano e informa a data da páscoa.
4. Por fim, inclua as duas funções criadas no programa principal para testá-las.

Regras e Dicas

Passo a passo para realizar o trabalho orientado a objetos:

1. Implemente a Classe “**DateStruct**”. Esta Classe representa o **tipo de dado Data** com seu respectivo construtor e com os devidos atributos que compõe uma data, isto é, dia, mês e ano. Observação: aplique os conceitos de encapsulamento corretamente.
2. A partir da Classe *DateStruct*, crie outra Classe denominada **Controller** deverá implementar as operações referentes ao TAD criado nesta atividade. Desta forma esta classe deverá conter as seguintes operações:
 - a. Método “lerData”: recebe e armazena uma data composta por dia, mês e ano via teclado
 - b. Método “validarData”: recebe como entrada uma data, verifica e retorna a data em questão é válida ou inválida (Exemplos de regras de negócio a serem verificadas: não permitir dia maior do que 31, mês até 12 e assim sucessivamente....). O retorno desta função pode ser 1 caso a data seja válida ou 0 caso contrário.
 - c. Método “verificarBissexto”: recebe uma data e retorna como saída 1 caso o ano seja bissexto ou 0 caso contrário. (Regras de negócio para cálculo de ano bissexto são facilmente encontradas no Google, em caso de dificuldade, por favor entre em contato).
 - d. Método “verificarPascoa”: recebe uma data como entrada (ou apenas o ano) e informa a data da páscoa daquele respectivo ano.
 - e. Método “escreverExtenso”: recebe uma data e a escreve por extenso. Por exemplo: 10/03/2020 deve ser escrito como 10 de março de 2020. Retorna verdadeiro se a operação foi realizada com sucesso e falso, caso contrário.
3. Após baixar e utilizar a Classe **DateStruct** e implementar a classe **Controller** (descrita no item 2), crie uma classe “**DateTest**” e inclua todas os métodos implementados no item 2 em um programa principal a fim de testá-los e validá-los.

Informações de envio:

1. Envio:

a. Onde?

A entrega deve ser feita via *Blackboard* (no link desta especificação).

b. Quando?

Até o dia **07/04/2021** até às **19:00**. A atividade é individual e vale presença na aula do dia 19/03/2020.

c. O que eu devo enviar?

Compactar o trabalho incluindo os arquivos .py em uma única pasta zipada.

d. Onde eu realizo este envio?

via Atividades Avaliativas de modo a ficar oficializado o seu envio.