



Matéria: CBTLP1 (Java) – ADS 371 – Professor: Wellington Tuler Moraes  
Nome: \_\_\_\_\_ Data: \_\_/\_\_/\_\_\_\_

**TRABALHO PRÁTICO 04: Construir a classe Data, conforme especificação abaixo, este exercício comporá a avaliação final, portanto será obrigatória sua elaboração para a prova final.**

Crie a classe Data conforme especificado abaixo:

Data
- dia: int - mes: int - ano: int
+ Data() + Data(int d, int m, int a) + entraDia(int d) + entraMes(int m) + entraAno(int a) + entraDia() + entraMes() + entraAno() + retDia(): int + retMes(): int + retAno(): int + mostra1(): String + mostra2(): String + bissexto(): boolean + diasTranscorridos(): int + apresentaDataAtual(): void

- O construtor Data() deverá permitir ao usuário digitar os valores de dia, mês e ano e com eles inicializar os atributos da classe. Os valores digitados deverão ser consistentes e só aceitos se válidos, caso contrário, redigitar;
- O construtor Data(int d, int m, int a) deverá receber os valores de dia, mês e ano e com eles inicializar as propriedades da classe;
- Os métodos entraDia(int d), entraMes(int m) e entraAno(int a) devem receber um valor e atribuí-lo às respectivas propriedades;
- Os métodos entraDia(), entraMes() e entraAno() devem permitir ao usuário digitar um valor e atribuí-lo a respectiva propriedade. Os valores digitados devem sofrer consistência e só aceitos quando válidos, caso contrário, solicitar ao usuário redigitar;
- Os métodos retDia(), retMes() e retAno() devem nos devolver as respectivas propriedades;
- O método mostra1() deve nos devolver a data no formato: dd/mm/aaaa;
- O método mostra2() deve nos devolver a data no formato: dd/mesPorExtenso/ano;
- O método bissexto() deve nos devolver um boolean informando se o ano é ou não bissexto;
- O método diasTranscorridos, deve retornar a quantidade de dias transcorridos no ano até a data digitada.
- O método apresentaDataAtual() deve imprimir a data atual, utilizando as classes Date e DateFormat, o DateFormat empregando o seguinte método: getDateInstance(DateFormat.FULL);
- Conveniente colocar tratamento de exceção para as possíveis inconsistências na entrada de dados.

## Exercício 02

Agora, desenvolva um programa capaz de testar a classe e os métodos desenvolvidos no exercício anterior.