



Programação Orientada a Objetos I: Java

CONCEITOS BÁSICOS DE CLASSE:

- Comando: **class**: ESTADO (Conjunto dos Atributos) +
- Comando: **class**: COMPORTAMENTO (Conjunto dos Métodos)
- Comando: **class**, **this**, **return**.

1) Escrever um algoritmo em Java que implementa a classe: Estado + Comportamento do diagrama a seguir:

Funcionário
Nome: String Matrícula: int Salário Bruto: double Plano de Saúde: bool Dependentes: int
void setNome(String Nome) void setMatricula(int Matricula) void setSalarioBruto(double SalarioBruto) void setPlanoSaude(bool PlanoSaude) void setDependentes(int Dependentes) double calcularDescontos() double calcularSalarioLiquido() double gastosPlanoSaude() double valorValeAlimentacao()

Descrição do comportamento da classe:

Método `double calcularDescontos()`: Este método totaliza os descontos do funcionário = PLANO DE SAÚDE (2% do Salário Bruto) + 1% do Salário Bruto / Dependente + 11% INSS;

Método `double calcularSalarioLiquido()`: Este método faz o cálculo do Salário Bruto - descontos;

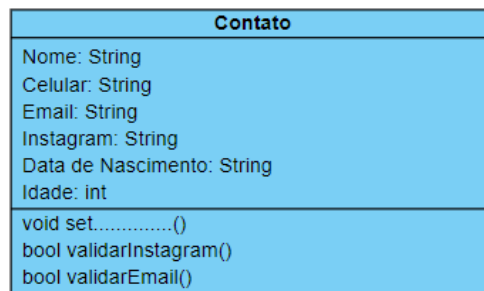
Método `double gastosPlanoSaude()`: Este método totaliza os gastos com Plano de Saúde;

Método `double valorValeAlimentacao()`: Este método faz o cálculo do Vale Alimentação = Salário Mínimo (R\$) + 1% do Salário Bruto / Dependente.

Na classe Drive (main), faça a instância de um **Objeto: Funcionário1** e exiba todo seu Estado + Comportamento com valores lidos do usuário.



- 2) Escrever um algoritmo que implementa a classe: Estado + Comportamento do diagrama a seguir:



Descrição do comportamento da classe:

Método `bool validarInstagram()`: Este método verifica se o primeiro caractere do instagram é um '@'; sendo que, caso afirmativo: return 1; caso contrário, return: 0;

Método `bool validarEmail()`: Este método verifica se há um caractere '@' no email e se depois dele há um caractere '.'; sendo que, caso afirmativo: return 1; caso contrário, return: 0;

Além disso, solicite ao usuário essas informações de um grupo de 5 pessoas. Pegue as informações via Scanner e após o término da entrada, apresente:

- ☐ a média das idades,
- ☐ a maior idade,
- ☐ a menor idade,
- ☐ a quantidade de pessoas maiores de idade.

- 3) Crie e implemente uma classe `PopulacaoBaratas` que simula o crescimento de uma população de baratas. A classe deve ter o seguinte Estado e Comportamento, a saber:

- ☐ A quantidade inicial da população de baratas é definida de forma Randômica. (pesquise a Classe Random...)
- ☐ O método `umentaBaratas`, simula a proporção que a população de baratas vai se multiplicar.
- ☐ O método `spray` pulveriza as baratas com um inseticida e reduz a população em 10%.
- ☐ O método `getQtdBaratas` devolve o número atual de baratas.

Na classe Drive (main), faça a instância de um Objeto: `PopulacaoBaratas` para VÁRIAS cozinhas que tenham uma população de baratas. Utilize a `umentaBaratas`, utilize o `spray` e imprima a contagem de baratas.