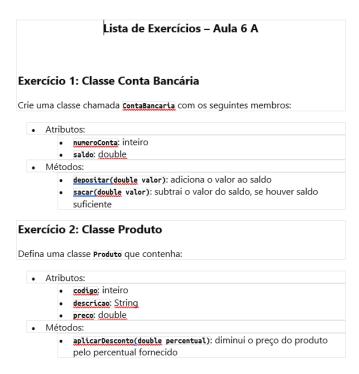
### Aula 6A



### Aula 6B

### 3) Classe Endereco

Objetivo: Criar uma classe chamada Endereco que contenha os atributos comuns a um endereço, como rua, cidade, estado, e cep. Inclua métodos de acesso (getters) e modificação (setters) para cada atributo.

# 4) Classe Pessoa Utilizando a Classe Endereco

Objetivo: Criar uma classe chamada Pessoa que contenha os atributos nome e idade, além de um atributo endereco que utilize a classe Endereco criada anteriormente. Inclua métodos de acesso (getters) e modificação (setters) para os atributos nome e idade, e um método específico para definir e obter o endereco.

## Aula 7

- 2) Crie uma classe em Java, que com as seguintes instruções:
  - Nome da classe: Universidade
  - Método: calculaMediaSemestral(): Esse método deverá receber como parâmetro 4 notas do aluno para efetuar a media final do aluno. O retorno do método será uma String que com a descrição da situação do aluno, se acordo com a regra abaixo:

MÉDIA FINAL	SITUAÇÃO
média ≥ 7.0	Aprovado
média ≥ 5.0 e < 7	Exame
média < 5	Reprovado

Para validar o método, crie uma classe chamada **TesteCalculoMediaSemestral**, que contém o método main ( )

- a) Instanciar a classe Universidade
- b) Chamar o método calcula Media Semestral, com diferentes alunos.
- c) Para obter os valores das notas de cada paciente, usar a classe Scanner ou JOptionPane.

Dica: Validar se as notas informadas são validas, antes de realizar o calculo da média =)

Ex: notas invalidas: -1 ou maior que 10

## Aula 8

- 2) Crei uma classe em Java chamada Looping com o seguinte método:
- exibeNumerosSequencial que recebe 2 parâmetros inteiro que indica o valor inicial e valor final da sequencia a ser exibida.

### Exemplo

- exibeNumerosSequencial (0,100) -> Exibe no console os números de 0 -100
- exibeNumerosSequencial (10,40) -> Exibe no console os números de 10 40

Para testar o método, crie uma nova classe chamada LoopingTest que contém o método main.

### Aula 9

 Testar a classe da página 17 da aula 9 com <u>String</u>, <u>StringBuffer</u> e <u>StringBuilder</u>.

Anotar os tempos de cada um e o processador e a memória RAM do seu computador.

```
public class Exemplo {
    public static void main(String[] args) {
        String msg = "a";
        msg = msg + "bc";
        System.out.println(msg);
    }
}
```

2) Fazer uma classe para testar todas as funções da página 13 da aula 9.

length, trim, replace, split, charAt e concat

- lenght() -> Retorna a quantidade de caracteres da String
- trim() -> Remove os espaços da esquerda e a direita
- replace(String regex, String new) -> Troca o contéudo da String pelo conteúdo do barâmetro.
- split(String regex) -> Divide a String de acordo com parâmetro regex
- charAt(int index) -> Devolve o caracter que está na posição index.
- concat(String str) -> Faz a junção (no final) da String original com a string passada como parametro.