

Parte II - Lendo e entendendo código - objetivo é compreender a estrutura sintaxe da linguagem

Para as questões 1 a 6, considere que a classe `AlunoGraduacao` e classe `TesteAlunoGraduacao` que contem o método `main` que inicia aplicação

```
public class AlunoGraduacao {  
    private double nota;  
    private String nome;  
  
    public AlunoGraduacao() {  
        super();  
    }  
    public AlunoGraduacao( String nome, double nota) {  
        super();  
        this.nota = nota;  
        this.nome = nome;  
    }  
    public AlunoGraduacao(double nota) {  
        super();  
        this.nota = nota;  
    }  
    public String verificarAprovacao() {  
        if (nota < 7)  
            return "Reprovado";  
        else  
            return "Aprovado";  
    }  
    public double calcularNota(double pNP1, double pNP2) {  
        double media = (pNP1 + pNP2) / 2;  
        nota = media;  
        return media;  
    }  
}
```

```
3 public class TesteAlunoGraduacao {  
4  
5     public static void main(String[] args) {  
6  
7  
8  
9  
10  
11     }
```

1) Questão

- a) Instancia um objeto da classe aluno graduação utilizando o construtor default (padrão)
- b) Invoque o método calcularNota passando como parâmetros, respectivamente, os valores numéricos 8.0 e 9.0 e atribua a um variável vNota.

2) Questão

- a) Instancia um objeto da classe aluno graduação utilizando o construtor com parâmetro de nota com valor 9.5
- b) Invoque o método verificarAprovação que não tem parâmetros e exibe o valor de retorno do método.

3) Questão

- a. Instancia um objeto da classe aluno graduação o utilizando o construtor com parâmetro nome com valor nota 5.5 e “jose”

Parte III - Escreva pequenos programas – objetivo é estimular a construir pequenos programas utilizando os conceitos abordados em sala de aula

1) **Questão –**

(1) Crie uma classe conta no pacote (br.unp.lpoo.parte1) com os respectivos atributos numero(String) e saldo(double)?

- Faça o encapsulamento da classe?
- Cria os seguintes métodos indicado na tabela:

Tipo de retorno	Nome do Método	Parâmetro	Objetivo do método – corpo
double	adicionarSaldo	double pValor	Formula : $vResultado = saldo * 2 + pValor$
void	sacar	double pValor	Formula : $saldo = saldo - pValor$

- Cria os seguintes métodos construtores indicado na tabela:

Nome do Método Construtor	Parâmetro	Objetivo do método – corpo
Conta	sem parâmetro	Construtor padrão
Conta	double pSaldo	Formula : $saldo = pSaldo$
Conta	String pNumero, double pSaldo	Formula : $saldo = pSaldo$ $numero = pNumero$

(2) Crie a classe Teste Conta como o método main .:

Atividade 1 :

- a. Instancia um objeto da classe conta utilizando o construtor default (padrão)
- b. Atribua o valor de 15.000 ao saldo ;
- c. Invoque o método depositar passando como parâmetro o valor 500.00 reais e atribua o valor de retorno do método na variável vSaldo;

Atividade 2 :

- a. Instancia um objeto da classe conta o utilizando o construtor com parâmetro o saldo com valor 55.000 reais
- b. Invoque o método sacar passando como parâmetro o valor 600.00 reais.

Atividade 3 :

- c. Instancia um objeto da classe conta o utilizando o construtor com parâmetro numero com numero “12345-1” e saldo com valor 1500 reais
- d. Invoque o método sacar passando o valor 500 reais.
- e. Invoque o método adicionar saldo passando 300 reais e exhibe o valor de retorno do método;
