

USJT – 2019/1 - Programação Orientada a Objetos - Laboratório de Exercícios

Professores: Machion e Mario

Aula: 05

Assunto: Introdução a OO

Exemplo Resolvido: Junto com seu professor, copie o código, cole no JGrasp; analise o código e entenda-o, depois execute-o para ver os resultados.

Ex1) Crie a classes conforme a descrição abaixo:

a) Crie a classe Aluno com os atributos privados: nome, do tipo String, idade, do tipo int, peso, do tipo double, formando, do tipo boolean e sexo, do tipo char.

b) Crie o construtor da classe Aluno que recebe parâmetros para inicializar os atributos nome, idade, peso e sexo. O atributo formando deve ser inicializado automaticamente com falso.

c) Crie os métodos de acesso para todos os atributos.

d) Crie os métodos modificadores para todos os atributos.

e) Crie a classe TesteAluno com um método main.

f) No método main, use o método JOptionPane.showInputDialog para ler os valores de nome, idade, peso e sexo.

g) Depois, ainda no main, instancie um objeto Aluno e passe como parâmetros do construtor os valores lidos.

h) Depois, ainda no main, imprima todos os atributos do objeto Aluno usando JOptionPane.showMessageDialog e os métodos de acesso de Aluno.

i) Depois, ainda no main, pergunte novamente ao usuário os valores da idade, do peso e se o aluno é formando. Altere estes atributos com os parâmetros informados por meio dos métodos modificadores.

j) Finalmente, ainda no main, imprima novamente os valores dos atributos com fez na letra h.

Solução: Classe Aluno

```

public class Aluno {
    // atributos
    private String nome;
    private int idade;
    private double peso;
    private boolean formando;
    private char sexo;

    // método construtor
    public Aluno(String n, int i, double p, char s) {
        nome = n;
        idade = i;
        peso = p;
        sexo = s;
        formando = false;
    }
    // métodos de acesso
    public String getNome() {
        return nome;
    }

    public int getIdade() {
        return idade;
    }

    public double getPeso() {
        return peso;
    }

    public boolean getFormando() {
        return formando;
    }

    public char getSexo() {
        return sexo;
    }
    // métodos modificadores
    public void setNome(String n) {
        nome = n;
    }

    public void setIdade(int i) {
        idade = i;
    }

    public void setPeso(double p) {
        peso = p;
    }

    public void setFormando(boolean f) {
        formando = f;
    }

    public void setSexo(char s) {
        sexo = s;
    }
}

```

Solução: Classe TesteAluno

```

import javax.swing.JOptionPane;
public class TesteAluno {

```

```

// cadastrar um novo aluno no metodo main
public static void main(String[] args) {

    // coletando os dados do aluno a ser cadastrado

    String nome = JOptionPane.showInputDialog("Nome");
    int idade = Integer.parseInt(JOptionPane.showInputDialog("Idade"));
    double peso = Double.parseDouble(JOptionPane.showInputDialog("Peso"));
    // pega o primeiro caracter da String e retorna como char
    char sexo = JOptionPane.showInputDialog("Sexo M/F").charAt(0);

    // cria um objeto aluno
    Aluno aluno = new Aluno(nome, idade, peso, sexo);

    // monta a String de saida chamando os metodos de acesso do aluno
    String msg = "Nome: " + aluno.getNome() + "\nIdade: "
        + aluno.getIdade() + " anos" + "\nPeso: " + aluno.getPeso()
        + " kg";
    if (aluno.getFormando()) {
        msg += "\nFormando: sim";
    } else {
        msg += "\nFormando: nao";
    }
    if (aluno.getSexo() == 'M') {
        msg += "\nsexo: masculino";
    } else {
        msg += "\nsexo: feminino";
    }
    // mostra o aluno
    JOptionPane.showMessageDialog(null, msg);

    // altera informacoes; nao precisa criar todas as variaveis novamente
    idade = Integer.parseInt(JOptionPane.showInputDialog("Idade"));
    peso = Double.parseDouble(JOptionPane.showInputDialog("Peso"));
    // tem que digitar true ou false
    boolean formando = Boolean.parseBoolean(JOptionPane
        .showInputDialog("E' formando?true/false"));

    // muda usando os metodo modificadores
    aluno.setIdade(idade);
    aluno.setPeso(peso);
    aluno.setFormando(formando);

    // mostra novamente o cadastro do aluno
    // monta a String de saida chamando os metodos de acesso do aluno
    msg = "Nome: " + aluno.getNome() + "\nIdade: " + aluno.getIdade()
        + " anos" + "\nPeso: " + aluno.getPeso() + " kg";
    if (aluno.getFormando()) {
        msg += "\nFormando: sim";
    } else {
        msg += "\nFormando: nao";
    }
    if (aluno.getSexo() == 'M') {
        msg += "\nsexo: masculino";
    } else {
        msg += "\nsexo: feminino";
    }
    // mostra o aluno
    JOptionPane.showMessageDialog(null, msg);
}
}

```

Note que há trechos onde o código se repete (montar a string msg para mostrar os dados do usuário, por exemplo. Na próxima aula iremos aprender a lidar melhor com isso.

Problemas Propostos:

Exercícios iniciais: valor 10,0 pontos

Resolva os exercícios desta seção para conquistar 10,0 pontos

1) Crie a classe Turma com seu construtor, métodos de acesso e modificadores e os atributos privados nome, do tipo String, curso, do tipo String, quantidadeDeAlunos, do tipo int, serie, do tipo int.

Exercícios intermediários: valor 10,0 pontos

Resolva os exercícios desta seção para conquistar mais 10,0 pontos

2) Crie a classe TesteTurma com o método main. De modo análogo ao exemplo, peça para o usuário entrar com os valores necessários para criar uma turma, instancie um objeto Turma e depois exiba os dados da turma criada. Depois, peça para o usuário uma nova quantidade de alunos, altere o valor do atributo e exiba os dados novamente.

Exercícios complementares (para praticar)

Resolva os exercícios desta seção para aprimorar seus conhecimentos

3) Crie a classe Produto com seu construtor, métodos de acesso e modificadores e os atributos privados nome, do tipo String, preço, do tipo double, quantidade, do tipo int. Crie uma classe TesteProduto com um método main, onde você vai ler valores de um produto, instanciar um produto e exibir o produto criado.

4) Crie a classe Disciplina com seu construtor, métodos de acesso e modificadores e os atributos privados nome, do tipo String, professor, do tipo String, semestre, do tipo int, ofertada, do tipo boolean. Crie a classe TesteDisciplina com o método main, leia dos dados necessários para criar uma disciplina, crie uma disciplina e exiba a disciplina criada.

Bibliografia

LOPES, ANITA. GARCIA, GUTO. Introdução à Programação: 500 algoritmos resolvidos. Rio de Janeiro: Elsevier, 2002.

DEITEL, P. DEITEL, H. Java: como programar. 8 Ed. São Paulo: Prentice – Hall (Pearson), 2010.