

Exercicios - Aula 1

Exercicio 1

Classe Pessoa

```
package exercices.aula1.pessoa;

public class People {

    private String cpf;
    private String name;
    private String sex;
    private int age;

    public People() {}

    public People(String cpf, String name, String sex, int age) {
        this.cpf = cpf;
        this.name = name;
        this.sex = sex;
        this.age = age;
    }

    public String getCpf() {
        return cpf;
    }

    public void setCpf(String cpf) {
        this.cpf = cpf;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public String getSex() {
        return sex;
    }

    public void setSex(String sex) {
        this.sex = sex;
    }

    public int getAge() {
        return age;
    }
}
```

```
    }

    public void setAge(int age) {
        this.age = age;
    }

    public String printPeople() {
        return "Cpf: " + this.cpf + "\nNome: " + this.name + "\nSexo: " +
this.sex + "\nIdade: " + this.age;
    }
}
```

Classe Principal

```
package exercices.aula1.pessoa;

import javax.swing.*;
import java.util.ArrayList;
import java.util.List;
import java.util.stream.Collectors;

public class App {

    public static void main(String[] args) {
        List<People> list = new ArrayList<People>();
        int op;

        op = Integer.parseInt(JOptionPane.showInputDialog("Digite uma
opcao: \n 1 - Criar pessoa \n 2 - Imprimir pessoa \n 0 - Sair"));

        do {
            switch (op) {
                case 1:
                    list.add(createPeople());
                    break;
                case 2:
                    printPeople(list);
                    break;
                default:
                    JOptionPane.showMessageDialog(new JFrame(), "Opcao
invalida!!!", "Error",
                        JOptionPane.ERROR_MESSAGE);
            }

            op = Integer.parseInt(JOptionPane.showInputDialog("Deseja
continuar? uma opcao: \n 1 - Criar pessoa \n 2 - Imprimir pessoa \n 0 -
Sair"));
        } while (op != 0);
    }
}
```

```
JOptionPane.showMessageDialog(new JFrame(), "Bye!!!", "Info",
    JOptionPane.INFORMATION_MESSAGE);
}

static People createPeople() {
    People p = new People();
    int opSex;

    p.setCpf(JOptionPane.showInputDialog("Digite o cpf da pessoa"));

    p.setName(JOptionPane.showInputDialog("Digite o nome da pessoa"));

    opSex = Integer.parseInt(JOptionPane.showInputDialog("Digite o sexo
da pessoa -> \n 1(M) \n 2(F)"));
    if(opSex == 1) {
        p.setSex("M");
    } else {
        p.setSex("F");
    }

    p.setAge(Integer.parseInt(JOptionPane.showInputDialog("Digite a
idade da pessoa"))));

    return p;
}

static void printPeople(List<People> peoples) {
    String name;

    name = JOptionPane.showInputDialog("Digite o nome da pessoa: ");

    List<People> people = peoples.stream()
        .filter(p ->
p.getName().equals(name)).collect(Collectors.toList());

    JOptionPane.showMessageDialog(new JFrame(),
people.get(0).printPeople(), "Info",
    JOptionPane.INFORMATION_MESSAGE);
}
}
```

Exercicio 2

Classe Aluno

```
package exercices.aula1.aluno;
```

```
public class Student {

    private String ra;
    private String name;
    private float ac1;
    private float ac2;
    private float ag;
    private float af;

    public Student() {}

    public Student(String ra, String name, float ac1, float ac2, float ag,
float af) {
        this.ra = ra;
        this.name = name;
        this.ac1 = ac1;
        this.ac2 = ac2;
        this.ag = ag;
        this.af = af;
    }

    public String getRa() {
        return ra;
    }

    public void setRa(String ra) {
        this.ra = ra;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public float getAc1() {
        return ac1;
    }

    public void setAc1(float ac1) {
        this.ac1 = ac1;
    }

    public float getAc2() {
        return ac2;
    }

    public void setAc2(float ac2) {
        this.ac2 = ac2;
    }

    public float getAg() {
```

```
        return ag;
    }

    public void setAg(float ag) {
        this.ag = ag;
    }

    public float getAf() {
        return af;
    }

    public void setAf(float af) {
        this.af = af;
    }

    public double calcularMedia() {
        return (this.ac1 * 0.15) + (this.ac2 * 0.30) + (this.ag * 0.10) +
        (this.af * 0.45);
    }

    public String verificarAprovacao() {
        return calcularMedia() > 5 ? "Aprovado" : "Reprovado";
    }

    public String imprimir() {
        return
            "ra: " + this.ra + "\n" +
            "name: " + this.name + "\n" +
            "ac1: " + this.ac1 + "\n" +
            "ac2:" + this.ac2 + "\n" +
            "ag: " + this.ag + "\n" +
            "af: " + af + "\n" +
            "Media: " + this.calcularMedia() + "\n" +
            "Situacao: " + this.verificarAprovacao();
    }
}
```

Classe Principal

```
package exercices.aula1.aluno;

import javax.swing.*;
import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;
import java.util.stream.Collectors;

public class App {

    public static void main(String[] args) {
```

```
List<Student> list = new ArrayList<Student>();
int op;

op = Integer.parseInt(JOptionPane.showInputDialog("Digite uma
opcao: \n 1 - Criar aluno \n 2 - Imprimir aluno \n 0 - Sair"));

do {
    switch (op) {
        case 1:
            list.add(createStudent());
            break;
        case 2:
            printStudent(list);
            break;
        default:
            JOptionPane.showMessageDialog(new JFrame(), "Opcao
invalida!!!", "Error",
                                      JOptionPane.ERROR_MESSAGE);
    }

    op = Integer.parseInt(JOptionPane.showInputDialog("Deseja
continuar? uma opcao: \n 1 - Criar aluno \n 2 - Imprimir aluno \n 0 -
Sair"));
} while(op != 0);

JOptionPane.showMessageDialog(new JFrame(), "Bye!!!", "Info",
                              JOptionPane.INFORMATION_MESSAGE);

}

static Student createStudent() {
    Scanner sc = new Scanner(System.in);
    Student p = new Student();

    p.setRa(JOptionPane.showInputDialog("Digite o Ra do Aluno:"));

    p.setName(JOptionPane.showInputDialog("Digite o nome do aluno"));

    p.setAc1(Float.parseFloat(JOptionPane.showInputDialog("Digite a
nota da AC1: ")));

    p.setAc2(Float.parseFloat(JOptionPane.showInputDialog("Digite a
nota da AC2: ")));

    p.setAg(Float.parseFloat(JOptionPane.showInputDialog("Digite a nota
da AG: ")));

    p.setAf(Float.parseFloat(JOptionPane.showInputDialog("Digite a nota
da AF: ")));

    return p;
}
```

```
}

static void printStudent(List<Student> students) {
    Scanner sc = new Scanner(System.in);
    String ra;

    ra = JOptionPane.showInputDialog("Digite o ra do aluno: ");

    List<Student> student = students.stream()
        .filter(p ->
            p.getRa().equals(ra)).collect(Collectors.toList());

    JOptionPane.showMessageDialog(new JFrame(),
        student.get(0).imprimir(), "Info",
        JOptionPane.INFORMATION_MESSAGE);
}

}
```

Exercicio 3

Classe Funcionario

```
package exercices.aula1.funcionario;

public class Funcionario {

    private int cracha;
    private String nome;
    private char tipoVinculo;
    private double valorHora;
    private double qtdeHora;
    private double salario;
    private double valorDesconto;

    public int getCracha() {
        return cracha;
    }

    public void setCracha(int cracha) {
        this.cracha = cracha;
    }

    public String getNome() {
        return nome;
    }

    public void setNome(String nome) {
        this.nome = nome;
    }
}
```

```
public char getTipoVinculo() {
    return tipoVinculo;
}

public void setTipoVinculo(char tipoVinculo) {
    this.tipoVinculo = tipoVinculo;
}

public double getValorHora() {
    return valorHora;
}

public void setValorHora(double valorHora) {
    this.valorHora = valorHora;
}

public double getQtdeHora() {
    return qtdeHora;
}

public void setQtdeHora(double qtdeHora) {
    this.qtdeHora = qtdeHora;
}

public double getSalario() {
    return salario;
}

public void setSalario(double salario) {
    this.salario = salario;
}

public double getValorDesconto() {
    return valorDesconto;
}

public void setValorDesconto(double valorDesconto) {
    this.valorDesconto = valorDesconto;
}

public double calcularSalario() {
    return this.tipoVinculo == 'H' ? (this.valorHora * this.qtdeHora) -
this.valorDesconto : this.salario - this.valorDesconto;
}

public String imprimir() {

    double salary = this.tipoVinculo == 'H' ? this.valorHora *
this.qtdeHora : this.salario;

    return
        "Cracha: " + this.cracha + "\n" +
        "Nome: " + this.nome + "\n" +
```



```

        "Tipo: " + this.tipoVinculo + "\n" +
        "Salario: " + salary + "\n" +
        "Desconto: " + this.valorDesconto + "\n" +
        "Valor a receber: " + this.calcularSalario() +
        "\n";
    }
}

```

Classe Principal

```

package exercices.aula1.funcionario;

import exercices.aula1.aluno.Student;

import javax.swing.*;
import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;
import java.util.stream.Collectors;

public class App {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);
        List<Funcionario> list = new ArrayList<Funcionario>();
        int op;

        op = Integer.parseInt(JOptionPane.showInputDialog("Digite uma
opcao: \n 1 - Criar funcionario \n 2 - Imprimir funcionario \n 3 - Alterar
tipo salario \n 0 - Sair"));

        do {
            switch (op) {
                case 1:
                    list.add(criarFuncionario());
                    break;
                case 2:
                    imprimirFuncionario(list);
                    break;
                case 3:
                    list = alterarTipoSalario(list);
                    break;
                default:
                    JOptionPane.showMessageDialog(new JFrame(), "Opcao
invalida!!!", "Error",
                                                JOptionPane.ERROR_MESSAGE);
            }

            op = Integer.parseInt(JOptionPane.showInputDialog("Deseja

```

```
realizar nova operacao?: \n 1 - Criar funcionario \n 2 - Imprimir  
funcionario \n 3 - Alterar tipo salario \n 0 - Sair"));  
    } while(op != 0);  
  
    System.out.println("Bye!!!");  
  
}  
  
static Funcionario criarFuncionario() {  
    Funcionario p = new Funcionario();  
    int tipo;  
  
    p.setCracha(Integer.parseInt(JOptionPane.showInputDialog("Digite o  
cracha do usuario: ")));  
  
    p.setNome(JOptionPane.showInputDialog("Digite o nome do  
funcionario: "));  
  
    return salarioFuncionario(p);  
}  
  
static Funcionario salarioFuncionario(Funcionario func) {  
    int tipo;  
  
    tipo = Integer.parseInt(JOptionPane.showInputDialog("Digite o tipo  
de vinculo do funcionario: \n 1(H) \n 2(M) "));  
  
    if(tipo == 1) {  
        func.setTipoVinculo('H');  
  
        func.setQtdeHora(Double.parseDouble(JOptionPane.showInputDialog("Digite o  
quantidade de horas trabalhadas: ")));  
  
        func.setValorHora(Double.parseDouble(JOptionPane.showInputDialog("Digite o  
valor da hora trabalhada: ")));  
    } else {  
        func.setTipoVinculo('M');  
  
        func.setSalario(Double.parseDouble(JOptionPane.showInputDialog("Digite o  
valor do salario: ")));  
    }  
  
    func.setValorDesconto(Double.parseDouble(JOptionPane.showInputDialog("Digite  
o valor do desconto: ")));  
  
    return func;  
}  
  
static List<Funcionario> alterarTipoSalario(List<Funcionario>
```

```
funcionarios) {
    int cracha;
    Funcionario func;

    cracha = Integer.parseInt(JOptionPane.showInputDialog("Digite o
cracha do usuario: "));

    List<Funcionario> funcionario = funcionarios.stream()
        .filter(p -> p.getCracha() ==
cracha).collect(Collectors.toList());

    funcionarios.removeIf(p -> p.getCracha() == cracha);

    func = salarioFuncionario(funcionario.get(0));

    funcionarios.add(func);

    return funcionarios;
}

static void imprimirFuncionario(List<Funcionario> funcionarios) {
    Scanner sc = new Scanner(System.in);
    int cracha;

    cracha = Integer.parseInt(JOptionPane.showInputDialog("Digite o
cracha do usuario: "));

    List<Funcionario> funcionario = funcionarios.stream()
        .filter(p -> p.getCracha() ==
cracha).collect(Collectors.toList());

    JOptionPane.showMessageDialog(new JFrame(),
funcionario.get(0).imprimir(), "Info",
JOptionPane.INFORMATION_MESSAGE);
}
}
```

Gabriel D. Padua