

## Exercício 1

- 1) Faça um projeto chamado Aula02-Exercicio01-ExemploRobo
- 2) Crie a classe Data com um método de inclusão `incluiData(byte dia, byte mes, short ano)`
- 3) Crie uma classe chamado Robo e um método para exibir as características do Robo.

```
public short codigoidentificacao;  
public String versao;  
public Data datacriacao;  
Crie o método incluiRobo(short codigo, String versao, byte dia, byte mes, short  
ano)
```

- 4) Crie uma Classe chamado Program com método main
- 5) No método Main crie uma referência chamado r2d2 e outro chamado c3po, ambos objetos (instâncias) da classe Robo
- 6) Coloque no código de identificação do Robo r2d2 o valor 1234
- 7) Coloque no código de identificação do Robo c3po o valor 999
- 8) Coloque na versão do Robo r2d2 o valor "V1.0"
- 9) Coloque na data de criação do Robo r2d2 e do c3po 15/12/2051
- 10) Coloque na versão do Robo c3po a mesma do Robo r2d2.
- 11) Crie o Robo bb8 inicializando com `bb8.incluiRobo((short) 777, "v2", (byte) 29, (byte) 2, (short) 2080);`
- 12) Altere o mes da data de criação do Robo bb8 para abril.
- 13) Mostra na tela as características dos robôs.

```
public class Data {  
    private byte dia;  
    public byte getDia() {  
        return dia;  
    }  
    public void setDia(byte dia) {  
        this.dia = dia;  
    }  
    public byte getMes() {  
        return mes;  
    }  
    public void setMes(byte mes) {  
        this.mes = mes;  
    }  
    public short getAno() {  
        return ano;  
    }  
    public void setAno(short ano) {  
        this.ano = ano;  
    }  
    private byte mes;  
    private short ano;  
  
    public String mostraData() {  
        return dia+"/"+mes+"/"+ano;  
    }  
    public void incluiData(byte d, byte m, short a) {  
        if (validaData(d,m,a)) {  
            dia = d; mes = m; ano = a;  
        }else {  
            System.out.println("Data inválida");  
            dia = 0; mes = 0; ano = 0;  
        }  
    }  
    boolean validaData(byte d,byte m,short a)
```

```

    {
        if ((d >=1) && (d <= 31) &&
            (m >= 1) && (m <= 12)) {
            return true;
        }
        else {
            return false;
        }
    }
}

```

```

public class Robo {
    public short getCodigodeidentificacao() {
        return codigodeidentificacao;
    }
    public void setCodigodeidentificacao(short codigodeidentificacao) {
        this.codigodeidentificacao = codigodeidentificacao;
    }
    public String getVersao() {
        return versao;
    }
    public void setVersao(String versao) {
        this.versao = versao;
    }
    public Data getDatacriacao() {
        return datacriacao;
    }
    public void setDatacriacao(Data datacriacao) {
        this.datacriacao = datacriacao;
    }
    private short codigodeidentificacao;
    private String versao;
    private Data datacriacao;
    public void mostraRobo() {
        System.out.println("\nCodigo="+codigodeidentificacao
            +"\nCriação:"+datacriacao.mostraData()
            +"\nCriação:"+versao);
    }
    public void incluiRobo(short cod, String vers, byte d, byte m, short a) {
        codigodeidentificacao=cod;
        versao=vers;
        datacriacao=new Data();
        datacriacao.incluiData(d, m, a);
    }
}

```

```

public class Program {
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Robo r2d2 =new Robo();
        Robo c3po =new Robo();
        r2d2.setCodigodeidentificacao((short)1234);
        c3po.setCodigodeidentificacao((short)999);
        r2d2.setVersao("V1.0");
        r2d2.setDatacriacao(new Data());
        r2d2.getDatacriacao().incluiData((byte) 15,(byte) 12, (short)2051);
        c3po.setDatacriacao(new Data());
        c3po.getDatacriacao().incluiData((byte) 15,(byte) 12, (short)2051);
        c3po.setVersao(r2d2.getVersao());
        Robo bb8=new Robo();
        bb8.incluiRobo((short) 777, "v2", (byte) 29, (byte) 2, (short) 2080);
    }
}

```

```
        bb8.getDatacriacao().setMes((byte)4);  
        r2d2.mostraRobo();  
        c3po.mostraRobo();  
        bb8.mostraRobo();  
    }  
}
```

## Exercício 2

- 1) Faça um projeto chamado Aula02-Exercício02
- 2) Crie uma classe chamado Conta

```
public double saldo;
public double limite;
public int conta;
```
- 3) Crie um método para incluir uma conta `incluiConta(int conta, double limite)` e inicialize o saldo com zero
- 4) Crie uma Classe chamado Program
- 5) No método Main crie uma referências chamados cc1,cc2 e cc3,objetos (instâncias) da classe Conta

```
cc1.incluiConta(111, 100);
cc2.incluiConta(222, 500);
cc3.incluiConta(333, 1000);
```
- 6) Implemente os Métodos Depositar, Sacar mostrar saldo na classe Conta
- 7) implementar o método transferirPara informando a conta destino e o valor.
- 8) Faça as seguintes operações:
  - a. Atribua os valores 111,222 e 333 nas contas criadas
  - b. Deposite 340 na conta 333
  - c. Na conta 222 saque 100
  - d. Transfira 150 da conta 111 para a conta 222
- 9) Encapsule de tal forma que apenas o limite seja possível alterar e o valor so saldo visível apenas pelo método MostraSaldo
- 10) Altere o limite da conta cc2 para 800

Sem encapsular:

```
public class Conta {
    public double saldo;
    public double limite;
    public int conta;
    public void depositar(double valor) {
        saldo+=valor;
    }
    public void incluiConta(int conta, double limite) {
        this.conta=conta;
        this.limite=limite;
        this.saldo=0;
    }
    public void sacar(double valor) {
        saldo-=valor;
    }
    public void mostraSaldo() {
        System.out.println("\n\nConta:"+conta+
            "\nSaldo:"+saldo+
            "\nSaldo disponível: "+(saldo+limite));
    }
    public void transferirPara(Conta cc,double valor) {
        sacar(valor);
        cc.depositar(valor);
    }
}

public class Principal {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Conta cc1 = new Conta();
```

```

        Conta cc2 = new Conta();
        Conta cc3 = new Conta();
        cc1.incluiConta(111, 100);
        cc2.incluiConta(222, 500);
        cc3.incluiConta(333, 1000);

        cc3.depositar(340);
        cc3.mostraSaldo();

        cc2.sacar(100);
        cc2.mostraSaldo();

        cc1.transferirPara(cc2, 150);
        cc1.mostraSaldo();
        cc2.mostraSaldo();
    }
}

```

Encapsulando

```

public class Conta {
    private double saldo;

    private double limite;
    public double getLimite() {
        return limite;
    }
    public void setLimite(double limite) {
        this.limite = limite;
    }
    private int conta;
    public int getConta() {
        return conta;
    }
    public void depositar(double valor) {
        saldo+=valor;
    }
    public void incluiConta(int conta, double limite) {
        this.conta=conta;
        this.limite=limite;
        this.saldo=0;
    }
    public void sacar(double valor) {
        saldo-=valor;
    }
    public void mostraSaldo() {
        System.out.println("\n\nConta:"+conta+
            "\nSaldo:"+saldo+
            "\nSaldo disponível: "+(saldo+limite));
    }
    public void transferirPara(Conta cc, double valor) {
        sacar(valor);
        cc.depositar(valor);
    }
}

public class Principal {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Conta cc1 = new Conta();
        Conta cc2 = new Conta();
    }
}

```

```
Conta cc3 = new Conta();
cc1.incluiConta(111, 100);
cc2.incluiConta(222, 500);
cc3.incluiConta(333, 1000);

cc3.depositar(340);
cc3.mostraSaldo();

cc2.sacar(100);
cc2.mostraSaldo();

cc1.transferirPara(cc2, 150);
cc1.mostraSaldo();
cc2.mostraSaldo();

cc2.setLimite(800);
cc2.mostraSaldo();
}
```

### Exercício 3

- 1) Em novo projeto Monte uma classe chamado Carro, e que guarde as seguintes características Marca, Modelo, Cor e ano Motorização.
- 2) Monte uma classe Motor com as suas características: Número de cilindros, Volume em litros, potencia, tipo de refrigeração, combustível (E - Elétrico, H híbrido, G-Gasolina, Alcool,F-Flex).
- 3) Crie os seguintes carros:
  - o Meu carro, um Volkswagen, fusca, Beje Nilo, 1968, motor bx1300 (4 cilindros, 1.3L , 40cv, refrigerado a ar, Gasolina), e o carro do Ze, um Gurgel BR500 ano 1985, cinza, motor g800 (2 cilindros, 800cc , 15cv, refrigerado a ar, Alcool).
- 4) Mostre os atributos dos objetos na tela.
- 5) Faça o encapsulamento validando o tipo de combustível dos motores

Sem encapsular

```
public class Carro {
    public String marca;
    public String modelo;
    public String cor;
    public int ano;
    public Motor motor;
    public void mostraCarro() {
        System.out.println(marca+"\n"+modelo+"
"+ano+"\n"+cor+"\n"+motor.mostraMotor());
    }
}
public class Motor {
    public int cilindros;
    public float volume;
    public int potencia;
    public String refrigeracao;
    public char combustivel;
    public String mostraMotor() {
        String m="\ncil:"+cilindros+"\nCC:"+volume;
        m=m+"\npotencia:"+potencia+"\nRefrigeração:"+refrigeracao;
        switch (combustivel) {
            case 'e': m=m+"\nEletrico";
                        break;
            case 'h': m=m+"\nHibrido";
                        break;
            case 'g': m=m+"\nGasolina";
                        break;
            case 'a': m=m+"\nAlcool";
                        break;
            case 'f': m=m+"\nFlex";
                        break;
        }
        return m;
    }
}

public class Principal {

    public static void main(String[] args) {
        Motor bx1300=new Motor();
        bx1300.cilindros=4;
        bx1300.volume=1.3f;
        bx1300.potencia=40;
```

```

        bx1300.refrigeracao="ar";
        bx1300.combustivel='g';
        Motor g800=new Motor();
        g800.cilindros=2;
        g800.volume=0.8f;
        g800.potencia=15;
        g800.refrigeracao="ar";
        g800.combustivel='a';
        Carro meucarro=new Carro();
        meucarro.marca="Volkswagen";
        meucarro.modelo="Fusca";
        meucarro.cor="Beje Nilo";
        meucarro.ano=1968;
        meucarro.motor=bx1300;

        Carro carroDoZe=new Carro();
        carroDoZe.marca="Gurgel";
        carroDoZe.modelo="BR500";
        carroDoZe.cor="Cinza";
        carroDoZe.ano=1985;
        carroDoZe.motor=g800;

        System.out.println("-----\nMeu Carro");
        meucarro.mostraCarro();
        System.out.println("-----\nCarro do Zé");
        carroDoZe.mostraCarro();
    }
}

```

Encapsulado:

```

public class Motor {
    public int cilindros;
    public int getCilindros() {
        return cilindros;
    }
    public void setCilindros(int cilindros) {
        this.cilindros = cilindros;
    }
    public float getVolume() {
        return volume;
    }
    public void setVolume(float volume) {
        this.volume = volume;
    }
    public int getPotencia() {
        return potencia;
    }
    public void setPotencia(int potencia) {
        this.potencia = potencia;
    }
    public String getRefrigeracao() {
        return refrigeracao;
    }
    public void setRefrigeracao(String refrigeracao) {
        this.refrigeracao = refrigeracao;
    }
    public char getCombustivel() {
        return combustivel;
    }
    public void setCombustivel(char combustivel) {
        if (combustivel == 'a'

```



```

        || combustivel == 'e'
        || combustivel == 'h'
        || combustivel == 'g'
        || combustivel == 'f')
    {
        this.combustivel = combustivel;
    }else {
        System.out.println("outro");
        this.combustivel = 'o';
    }
}
private float volume;
private int potencia;
private String refrigeracao;
private char combustivel;
public String mostraMotor() {
    String m="\ncil:"+cilindros+"\nCC:"+volume;
    m=m+"\npotencia:"+potencia+"\nRefrigeração:"+refrigeracao;
    switch (combustivel) {
        case 'e': m=m+"\nEletrico";
                    break;
        case 'h': m=m+"\nHibrido";
                    break;
        case 'g': m=m+"\nGasolina";
                    break;
        case 'a': m=m+"\nAlcool";
                    break;
        case 'f': m=m+"\nFlex";
                    break;
    }
    return m;
}
}
public class Motor {
    public int cilindros;
    public int getCilindros() {
        return cilindros;
    }
    public void setCilindros(int cilindros) {
        this.cilindros = cilindros;
    }
    public float getVolume() {
        return volume;
    }
    public void setVolume(float volume) {
        this.volume = volume;
    }
    public int getPotencia() {
        return potencia;
    }
    public void setPotencia(int potencia) {
        this.potencia = potencia;
    }
    public String getRefrigeracao() {
        return refrigeracao;
    }
    public void setRefrigeracao(String refrigeracao) {
        this.refrigeracao = refrigeracao;
    }
    public char getCombustivel() {
        return combustivel;
    }
}

```

```

    }
    public void setCombustivel(char combustivel) {
        if (combustivel == 'a'
            || combustivel == 'e'
            || combustivel == 'h'
            || combustivel == 'g'
            || combustivel == 'f')
        {
            this.combustivel = combustivel;
        } else {
            System.out.println("outro");
            this.combustivel = 'o';
        }
    }
    private float volume;
    private int potencia;
    private String refrigeracao;
    private char combustivel;
    public String mostraMotor() {
        String m="\ncil:"+cilindros+"\nCC:"+volume;
        m=m+"\npotencia:"+potencia+"\nRefrigeração:"+refrigeracao;
        switch (combustivel) {
            case 'e': m=m+"\nEletrico";
                       break;
            case 'h': m=m+"\nHibrido";
                       break;
            case 'g': m=m+"\nGasolina";
                       break;
            case 'a': m=m+"\nAlcool";
                       break;
            case 'f': m=m+"\nFlex";
                       break;
        }
        return m;
    }
}

```

```

public class Principal {

    public static void main(String[] args) {
        Motor bx1300=new Motor();
        bx1300.setCilindros(4);
        bx1300.setVolume(1.3f);
        bx1300.setPotencia(40);
        bx1300.setRefrigeracao("ar");
        bx1300.setCombustivel('g');
        Motor g800=new Motor();
        g800.setCilindros(2);
        g800.setVolume(0.8f);
        g800.setPotencia(15);
        g800.setRefrigeracao("ar");
        g800.setCombustivel('a');
        Carro meucarro=new Carro();
        meucarro.setMarca("Volkswagen");
        meucarro.setModelo("Fusca");
        meucarro.setCor("Beje Nilo");
        meucarro.setAno(1968);
        meucarro.setMotor(bx1300);

        Carro carroDoZe=new Carro();
        carroDoZe.setMarca("Gurgel");
    }
}

```

```
        carroDoZe.setModelo("BR500");
        carroDoZe.setCor("Cinza");
        carroDoZe.setAno(1985);
        carroDoZe.setMotor(g800);

        System.out.println("-----\nMeu Carro");
        meuCarro.mostraCarro();
        System.out.println("-----\nCarro do Zé");
        carroDoZe.mostraCarro();
    }
}
```