



06

Mão na massa!

Corrija o problema de encapsulamento do código abaixo:

```
public class ProcessadorDeBoletos {

    public void processa(List<Boleto> boletos, Fatura fatura) {

        double total = 0;
        for(Boleto boleto : boletos) {
            Pagamento pagamento = new Pagamento(boleto.getValor(), MeioDePagamento.BOLETO);
            fatura.getPagamentos().add(pagamento);

            total += boleto.getValor();
        }

        if(total >= fatura.getValor()) {
            fatura.setPago(true);
        }
    }
}

public class Boleto {

    private double valor;

    public Boleto(double valor) {
        this.valor = valor;
    }

    public double getValor() {
        return valor;
    }

}

public class Fatura {

    private String cliente;
    private double valor;
    private List<Pagamento> pagamentos;
    private boolean pago;

    public Fatura(String cliente, double valor) {
        this.cliente = cliente;
        this.valor = valor;
        this.pagamentos = new ArrayList<Pagamento>();
        this.pago = false;
    }

    public String getCliente() {
        return cliente;
    }

}
```

```
public double getValor() {
    return valor;
}

public List<Pagamento> getPagamentos() {
    return pagamentos;
}

public boolean isPago() {
    return pago;
}

public void setPago(boolean pago) {
    this.pago = pago;
}

}

public enum MeioDePagamento {

    BOLETO,
    CARTAO
}

public class Pagamento {

    private double valor;
    private MeioDePagamento forma;

    public Pagamento(double valor, MeioDePagamento forma) {
        this.valor = valor;
        this.forma = forma;
    }

    public double getValor() {
        return valor;
    }

    public MeioDePagamento getForma() {
        return forma;
    }

    @Override
    public boolean equals(Object obj) {
        if (!(obj instanceof Pagamento)) {
            return false;
        }
        Pagamento outro = (Pagamento) obj;
        if (forma != outro.forma
            || Double.doubleToLongBits(valor) != Double
                .doubleToLongBits(outro.valor)) {
            return false;
        }

        return true;
    }
}
```

```
}
```

Responda

INSERIR CÓDIGO		FORMATAR
codigo aqui		