

Luis Gabriel Maximo

11721BCC027

Bad Smells Refactoring

- A seguir está um código feito no início do meu aprendizado com Java, o algoritmo é responsável por controlar os aluguéis de uma locadora, mais precisamente o código responsável por exibir o registro de aluguéis de um cliente.

aplicação possui algumas regras de negócio que são:

filmes são divididos em faixas de preços:

normal, infantil e lançamento

Cada aluguel rende um ponto para o cliente, sendo que se o filme for lançamento e o aluguel durar mais que dois dias ele ganha um ponto extra

valor do aluguel depende da faixa de preço a que o filme pertence e do tempo que ele ficou alugado

Com certa quantidade de pontos o cliente ganha o aluguel de um filme.

```
public class Filme {

    public static final int NORMAL = 0;
    public static final int INFANTIL = 1;
    public static final int LANCAMENTO = 2;

    private String título;
    private int preco;

    public Filme(String título, int preco) {
        this.título = título;
        this.preco = preco;
    }

    public int getPreco() {
        return preco;
    }

    public void setPreco(int preco) {
        this.preco = preco;
    }

    public String getTitulo() {
        return título;
    }
}
```

```
public class Aluguel {
    private Filme filme;
    private int diasAluguel;

    public Aluguel(Filme filme, int
diasAluguel) {
        this.filme = filme;
        this.diasAluguel =
diasAluguel;
    }

    public Filme getFilme() {
        return filme;
    }

    public int getDiasAluguel() {
        return diasAluguel;
    }
}
```

— — —

```

public class Cliente {

    private String nome;
    private List<Aluguel> alugueis;

    public Cliente(String nome) {
        this.nome = nome;
        this.alugueis = new ArrayList<Aluguel>();
    }

    public void addAluguel(Aluguel aluguel) {
        this.alugueis.add(aluguel);
    }

    public String getNome() {
        return nome;
    }

    public String exibirRegistroAlugueis() {
        double valorTotal = 0;
        int pontos = 0;
        StringBuilder dados = new StringBuilder();
        dados.append("Registro de Aluguéis do cliente: " + getNome() + "\n");
        for (Aluguel aluguel : this.alugueis) {
            double valor = 0;

            // Calcula o valor do aluguel
            switch (aluguel.getFilme().getPreco()) {
                case Filme.NORMAL:
                    valor += 1.5;
                    if (aluguel.getDiasAluguel() > 3)
                        valor += (aluguel.getDiasAluguel() - 3) * 1.5;
                    break;
                case Filme.INFANTIL:
                    valor += 2;
                    if (aluguel.getDiasAluguel() > 2)
                        valor += (aluguel.getDiasAluguel() - 2) * 1.5;
                    break;
                case Filme.LANCAMENTO:
                    valor += aluguel.getDiasAluguel() * 3;
                    break;
            }

            // Adiciona um ponto
            pontos++;

            // Bônus para mais de dois dias com um lançamento
            if (aluguel.getFilme().getPreco() == Filme.LANCAMENTO
                && aluguel.getDiasAluguel() > 1)
                pontos++;

            // Adiciona os dados desse aluguel
            dados.append("\t" + aluguel.getFilme().getTitulo() + "\t");
            dados.append(" R$ " + String.valueOf(valor) + "\n");

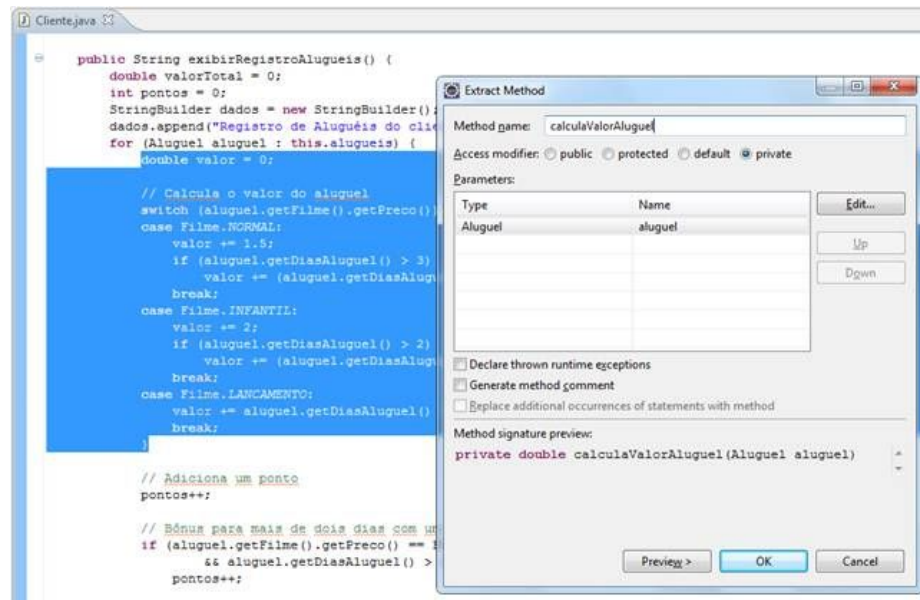
            valorTotal += valor;
        }

        // Rodapé
        dados.append("Total gasto com alugueis: R$ " + String.valueOf(valorTotal) + "\n");
        dados.append("Pontos ganhos: " + String.valueOf(pontos));

        return dados.toString();
    }
}

```

- O trecho de código grifado em cinza, pode ser considerado um Bad Smells, e para refatorar esse trecho usei o Eclipse e defini um método para calcular o valor do aluguel ao invés de implementar dentro da Class Cliente



```

public class Cliente {
    // ...
    public String exibirRegistroAlugueis() {
        double valorTotal = 0;
        int pontos = 0;
        StringBuilder dados = new StringBuilder();
        dados.append("Registro de Aluguéis do cliente: " + getNome() + "\n");
        for (Aluguel aluguel : this.alugueis) {

            // Calcula o valor do aluguel
            double valor = calculaValorAluguel(aluguel, valor);

            // Adiciona um ponto
            pontos++;

            // Bônus para mais de dois dias com um lançamento
            if (aluguel.getFilme().getPreco() == Filme.LANCAMENTO
                && aluguel.getDiasAluguel() > 1)
                pontos++;

            // Adiciona os dados desse aluguel
            dados.append("\t" + aluguel.getFilme().getTitulo() + "\t");
            dados.append(" = R$ " + String.valueOf(valor) + "\n");

            valorTotal += valor;
        }

        // Rodapé
        dados.append("Total gasto com alugueis: R$ " + String.valueOf(valorTotal) + "\n");
        dados.append("Pontos ganhos: " + String.valueOf(pontos));

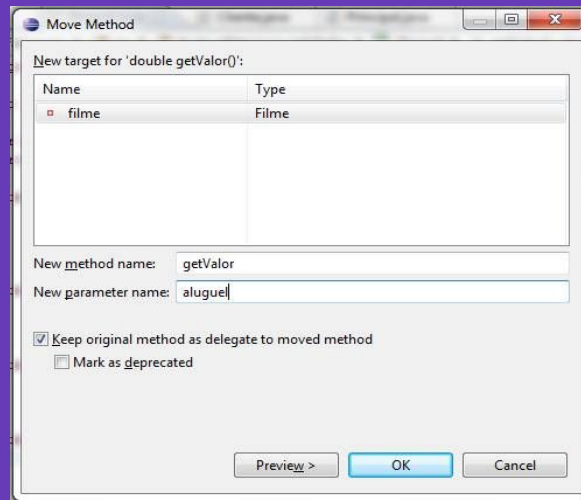
        return dados.toString();
    }

    private double calculaValorAluguel(Aluguel aluguel) {
        double valor = 0;
        switch (aluguel.getFilme().getPreco()) {
            case Filme.NORMAL:
                valor += 1.5;
                if (aluguel.getDiasAluguel() > 3)
                    valor += (aluguel.getDiasAluguel() - 3) * 1.5;
                break;
            case Filme.INFANTIL:
                valor += 2;
                if (aluguel.getDiasAluguel() > 2)
                    valor += (aluguel.getDiasAluguel() - 2) * 1.5;
                break;
            case Filme.LANCAMENTO:
                valor += aluguel.getDiasAluguel() * 3;
                break;
        }
        return valor;
    }
}

```

Após refatorar, a Class Cliente fica:

O método é posto abaixo, e para melhorar movi para a classe Aluguel, pois ele não utiliza nada da classe cliente. Para mover, Utilizei o Eclipse através do atalho de Move Method.



```

public class Aluguel {
    // ...
    public double getValor() {
        double valor = 0;
        switch (getFilme().getPreco()) {
            case Filme.NORMAL:
                valor += 1.5;
                if (getDiasAluguel() > 3)
                    valor +=
(getDiasAluguel() - 3) * 1.5;
                break;
            case Filme.INFANTIL:
                valor += 2;
                if (getDiasAluguel() > 2)
                    valor +=
(getDiasAluguel() - 2) * 1.5;
                break;
            case Filme.LANCAMENTO:
                valor += getDiasAluguel() * 3;
                break;
        }
        return valor;
    }
}

```

Nova Classe Aluguel



Nova Class Client



```

public class Cliente {
    // ...
    public String exibirRegistroAlugueis() {
        double valorTotal = 0;
        int pontos = 0;
        StringBuilder dados = new StringBuilder();
        dados.append("Registro de Aluguéis do cliente: " + getNome() + "\n");
        for (Aluguel aluguel : this.alugueis) {

            // Calcula o valor do aluguel
            double valor += aluguel.getValor();

            // Adiciona um ponto
            pontos++;

            // Bônus para mais de dois dias com um lançamento
            if (aluguel.getFilme().getPreco() == Filme.LANCAMENTO
                && aluguel.getDiasAluguel() > 1)
                pontos++;
            // Adiciona os dados desse aluguel
            dados.append("\t" + aluguel.getFilme().getTitulo() + "\t");
            dados.append(" = R$ " + String.valueOf(valor) + "\n");
            valorTotal += valor;

        }

        // Rodapé
        dados.append("Total gasto com aluguéis: R$ " + String.valueOf(valorTotal)
+ "\n");
        dados.append("Pontos ganhos: " + String.valueOf(pontos));

        return dados.toString();
    }
}

```

```

public class Aluguel {
    // ...
    public double getValor() {
        double valor = 0;
        switch (getFilme().getPreco()) {
            case Filme.NORMAL:
                valor += 1.5;
                if (getDiasAluguel() > 3)
                    valor += (getDiasAluguel() - 3) * 1.5;
                break;
            case Filme.INFANTIL:
                valor += 2;
                if (getDiasAluguel() > 2)
                    valor += (getDiasAluguel() - 2) * 1.5;
                break;
            case Filme.LANCAMENTO:
                valor += getDiasAluguel() * 3;
                break;
        }
        return valor;
    }

    public int getPontos() {
        int pontos = 1;

        // Bônus para mais de dois dias com um lançamento
        if (getFilme().getPreco() == Filme.LANCAMENTO &&
            getDiasAluguel() > 1)
            pontos++;
        return pontos;
    }
}

```

Adicionei em seguida o método `getPontos` que estava dentro da Class cliente, na Class `Aluguel`, Segue a resultado da Class `Cliente` atual abaixo:

```

public class Cliente {
    // ...
    public String exibirRegistroAlugueis() {
        double valorTotal = 0;
        int pontos = 0;
        StringBuilder dados = new StringBuilder();
        dados.append("Registro de Aluguéis do cliente: " + getNome() + "\n");
        for (Aluguel aluguel : this.alugueis) {
            double valor = 0;

            // Calcula o valor do aluguel
            valor += aluguel.getValor();

            // Calcula os pontos
            pontos += aluguel.getPontos();

            // Adiciona os dados desse aluguel
            dados.append("\t" + aluguel.getFilme().getTitulo() + "\t");
            dados.append(" = R$ " + String.valueOf(valor) + "\n");

            valorTotal += valor;
        }

        // Rodapé
        dados.append("Total gasto com alugueis: R$ " + String.valueOf(valorTotal) + "\n");
        dados.append("Pontos ganhos: " + String.valueOf(pontos));

        return dados.toString();
    }
}

```



```
public class Cliente {  
    // ...  
    public String exibirRegistroAlugueis() {  
        double valorTotal = 0;  
        int pontos = 0;  
        StringBuilder dados = new StringBuilder();  
        dados.append("Registro de Aluguéis do cliente: " + getNome() + "\n");  
        for (Aluguel aluguel : this.alugueis) {  
            double valor = 0;  
  
            // Calcula o valor do aluguel  
            valor += aluguel.getValor();  
  
            // Calcula os pontos  
            pontos += aluguel.getPontos();  
  
            // Adiciona os dados desse aluguel  
            dados.append("\t" + aluguel.getFilme().getTitulo() + "\t");  
            dados.append(" = R$ " + String.valueOf(valor) + "\n");  
  
            valorTotal += valor;  
        }  
  
        // Rodapé  
        dados.append("Total gasto com aluguéis: R$ " + String.valueOf(valorTotal) + "\n");  
        dados.append("Pontos ganhos: " + String.valueOf(pontos));  
        return dados.toString();  
    }  
}
```

**Após refatorar os 2
Bad Smells, a Class
Cliente no fim fica:**

