

☰ 10

Que tal essa extração de métodos?

Abaixo estão 2 listagens, cada uma exibindo a mesma parte de um programa, porém a primeira listagem mostra o programa funcionando antes e a segunda listagem mostra o programa após a refatoração *Extrair Método*.

Antes da refatoração:

```
class Relatorio
{
    void Imprimir()
    {
        decimal total = 0.0m;
        var pedido = new Pedido("José da Silva");
        pedido.AddItem("Dentozap", 2, 10m, 0m, 3m);
        pedido.AddItem("Voldax", 3, 10m, 0m, 3m);
        pedido.AddItem("Tranlab", 7, 10m, 0m, 3m);

        // imprimir itens
        Console.WriteLine("*****");
        Console.WriteLine("***** Itens *****");
        Console.WriteLine("*****");
        foreach (var item in pedido.Itens)
        {
            decimal valorItem = item.Quantidade * item.PrecoBase;
            Console.WriteLine($"{item.Desconto}: {item.Quantidade} unidades, R$ {valorItem}");
            total = total + valorItem;
        }

        // imprimir detalhes
        Console.WriteLine("*****");
        Console.WriteLine("***** Resumo*****");
        Console.WriteLine("*****");
        Console.WriteLine("nome: " + pedido.Cliente);
        Console.WriteLine("valor: " + total);
    }
}
```

Depois da refatoração:

```
class Relatorio
{
    void Imprimir()
    {
        Pedido pedido = CriarPedido();
        decimal total = ImprimirParte1(pedido);
        ImprimirParte2(pedido, total);
    }

    private static Pedido CriarPedido()

```

```

    {
        var pedido = new Pedido("José da Silva");
        pedido.AddItem("Dentozap", 2, 10m, 0m, 3m);
        pedido.AddItem("Voldax", 3, 10m, 0m, 3m);
        pedido.AddItem("Tranlab", 7, 10m, 0m, 3m);
        return pedido;
    }

    private static decimal ImprimirParte1(Pedido pedido)
    {
        Console.WriteLine("*****");
        Console.WriteLine("***** Itens *****");
        Console.WriteLine("*****");
        decimal total = 0.0m;
        foreach (var item in pedido.Itens)
        {
            decimal valorItem = GetValorItem(item);
            Console.WriteLine($"{item.Desconto}: {item.Quantidade} unidades, R$ {valorItem}");
            total = total + valorItem;
        }

        return total;
    }

    private static void ImprimirParte2(Pedido pedido, decimal total)
    {
        Console.WriteLine("*****");
        Console.WriteLine("***** Resumo*****");
        Console.WriteLine("*****");
        Console.WriteLine("nome: " + pedido.Cliente);
        Console.WriteLine("valor: " + total);
    }

    private static decimal GetValorItem(Item item)
    {
        return item.Quantidade * item.PrecoBase;
    }
}

```

Qual o problema na refatoração feita acima?

Selezione uma alternativa

A A refatoração eliminou os comentários "imprimir itens" e "imprimir detalhes", e agora fica mais difícil entender o que o código faz. Esses comentários deveriam ter sido preservados no método original ou nos novos métodos criados.

B Não há problema algum. A refatoração está perfeitamente aceitável.

C Os métodos `ImprimirParte1` e `ImprimirParte2` têm nomes muito genéricos, e agora fica difícil entender o que o código desses métodos faz.

