

Mãos a obra: Médias Móveis mais flexíveis

A `MediaMovelSimples` é já um `Indicador` e agora também vai depender de algum *outro* `Indicador`. Esse outro indicador vamos usar para calcular (por exemplo o `IndicadorFechamento`). Vamos aplicar o padrão de projeto **Decorator**!

1) Abra a classe `MediaMovelSimples` e crie um novo atributo `outroIndicador` do tipo `Indicador`:

```
public class MediaMovelSimples implements Indicador {  
  
    private final Indicador outroIndicador;  
  
    //métodos omitidos  
}
```

2) Ainda na classe `MediaMovelSimples` gera o construtor que recebe um `Indicador`:

```
public MediaMovelSimples(Indicador outroIndicador) {  
    this.outroIndicador = outroIndicador;  
}
```

3) Dentro do método `calcula`, em vez de chamarmos o `getFechamento`, delegamos a chamada para o `outroIndicador`:

```
@Override  
public double calcula(int posicao, SerieTemporal serie) {  
    double soma = 0.0;  
    for (int i = posicao; i > posicao - 3; i--) {  
        soma += outroIndicador.calcula(i, serie); //delegando  
    }  
    return soma / 3;  
}
```

4) Modifique também o método `toString()`:

```
public String toString() {  
    return "MMS" + this.outroIndicador.toString(); //delegando tbm  
}
```

5) Na classe `ArgumentBean`, no construtor dela teste a `MediaMovelSimples`:

```
GeradorModeloGrafico gerador = new GeradorModeloGrafico(serie, 2, 40);  
MediaMovelSimples mms = new MediaMovelSimples(new IndicadorFechamento());  
gerador.plotaIndicador(mms);
```

6) Verifique todo o seu código e reinicie o Tomcat. Teste a aplicação pela interface web!

7) Uma vez funcionando, teste a `MediaMovelSimples` com `IndicadorAbertura`:

```
MediaMoveISimples mms = new MediaMoveISimples(new IndicadorAbertura());
```