

# Praticando Test-Driven Development (TDD)

## 1. Testes, do jeito que você já sabe

Apesar de ser simples, no contexto de um sistema de leilão, essa classe é de extrema importância, portanto, merece ser testada.

```
public class Leilao {  
  
    private String descricao;  
    private List<Lance> lances;  
  
    public Leilao(String descricao) {  
        this.descricao = descricao;  
        this.lances = new ArrayList<Lance>();  
    }  
  
    public void propoe(Lance lance) {  
        lances.add(lance);  
    }  
  
    public String getDescricao() {  
        return descricao;  
    }  
  
    public List<Lance> getLances() {  
        return Collections.unmodifiableList(lances);  
    }  
}
```

O método `propoe()`, em especial, é essencial para o leilão e provavelmente sofrerá mudanças no decorrer do projeto. Por isso, ela precisa ser testada para termos certeza de que ela está funcionando conforme se espera. Inicialmente, vamos analisar se um determinado lance que foi proposto ficará armazenado no leilão. Para isso, temos dois casos a serem averiguados: a realização de apenas um lance e a de mais de um lance.

O código para realizar tal teste não tem muito segredo. Começaremos instanciando um leilão e serão propostos alguns lances nele:

```
import static org.junit.Assert.assertEquals;  
import org.junit.Test;  
  
public class LeilaoTest {  
  
    @Test  
    public void deveReceberUmLance() {  
        Leilao leilao = new Leilao("Macbook Pro 15");  
        assertEquals(0, leilao.getLances().size());  
  
        leilao.propoe(new Lance(new Usuario("Steve Jobs"), 2000));  
        assertEquals(1, leilao.getLances().size());  
        assertEquals(2000, leilao.getLances().get(0).getValor(), 0.00001);  
    }  
  
    @Test  
    public void deveReceberVariosLances() {  
        Leilao leilao = new Leilao("Macbook Pro 15");  
        leilao.propoe(new Lance(new Usuario("Steve Jobs"), 2000));  
        leilao.propoe(new Lance(new Usuario("Steve Wozniak"), 3000));  
        assertEquals(2, leilao.getLances().size());  
        assertEquals(2000, leilao.getLances().get(0).getValor(), 0.00001);  
        assertEquals(3000, leilao.getLances().get(1).getValor(), 0.00001);  
    }  
}
```

Agora implementaremos duas novas regras de negócio no processo de lances em um leilão:

- Uma pessoa não pode propor dois lances em sequência;
- Uma pessoa não pode dar mais do que cinco lances no mesmo leilão.