



TESTES AUTOMATIZADOS

FAPESC – DESENVOLVEDORES PARA TECNOLOGIA DA INFORMAÇÃO

HERCULANO DE BIASI

herculano.debiasi@unoesc.edu.br



BIBLIOGRAFIA

- O exemplo desta apresentação foi adaptado do livro abaixo
- ANICHE, Maurício. **Testes automatizados de software**: Um guia prático. São Paulo: Casa do Código, 2015.



JUNIT COM STS

- Crie um novo projeto do tipo Java com o nome de TesteJUnit

New Java Project

Create a Java Project

Create a Java project in the workspace or in an external location.

Project name:

☒ Use default location

Location: [Browse...](#)

JRE

☒ Use an execution environment JRE:

☐ Use a project specific JRE:

☐ Use default JRE 'jre' and workspace compiler preferences [Configure JREs...](#)

Project layout

☐ Use project folder as root for sources and class files

☒ Create separate folders for sources and class files [Configure default...](#)

Working sets

☐ Add project to working sets [New...](#)

Working sets: [Select...](#)

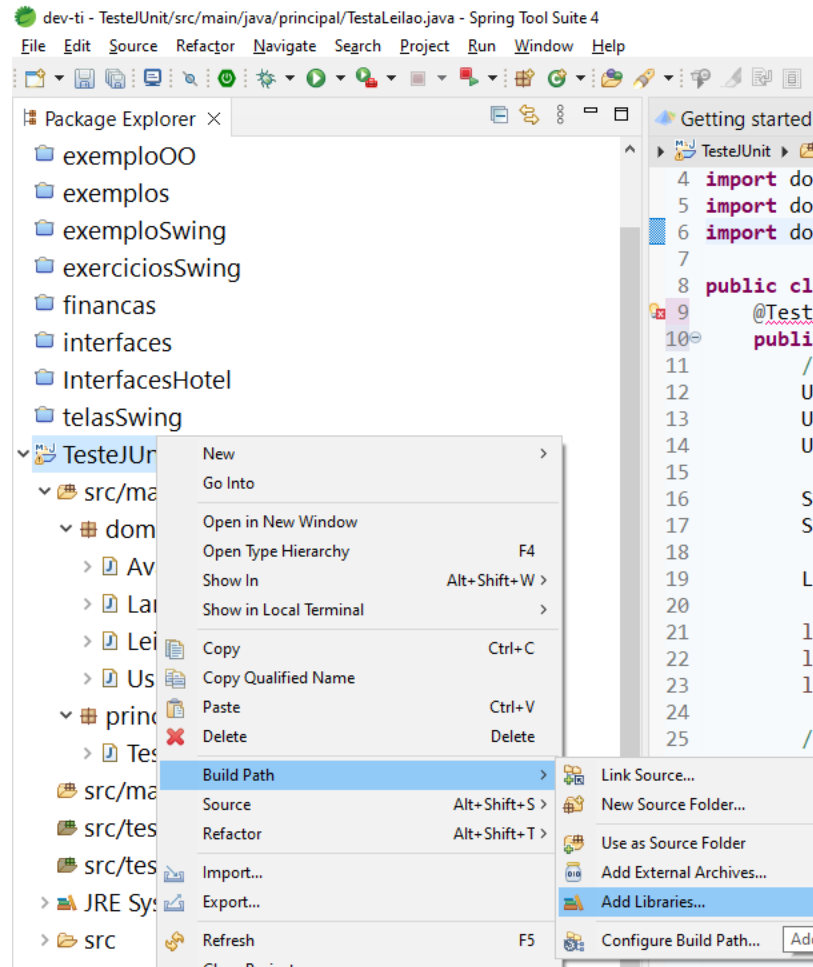
Module

☒ Create module-info.java file

[?](#) [< Back](#) [Next >](#) [Finish](#) [Cancel](#)

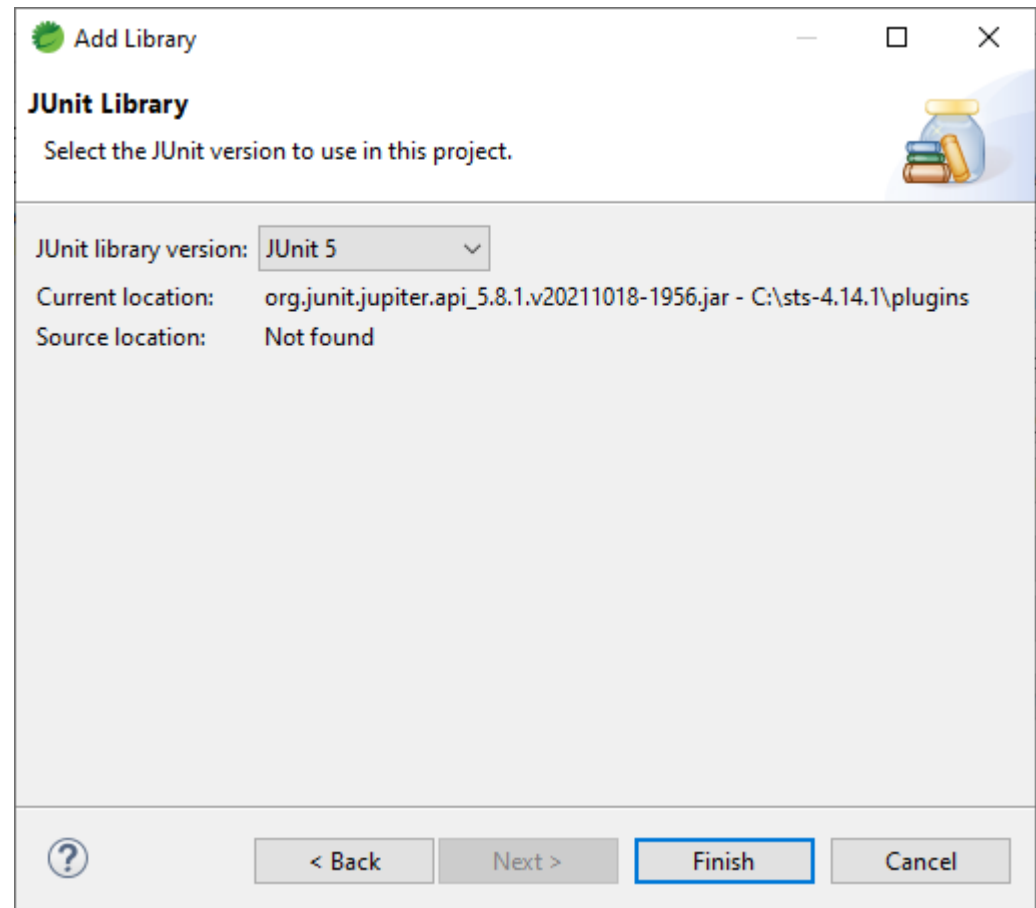
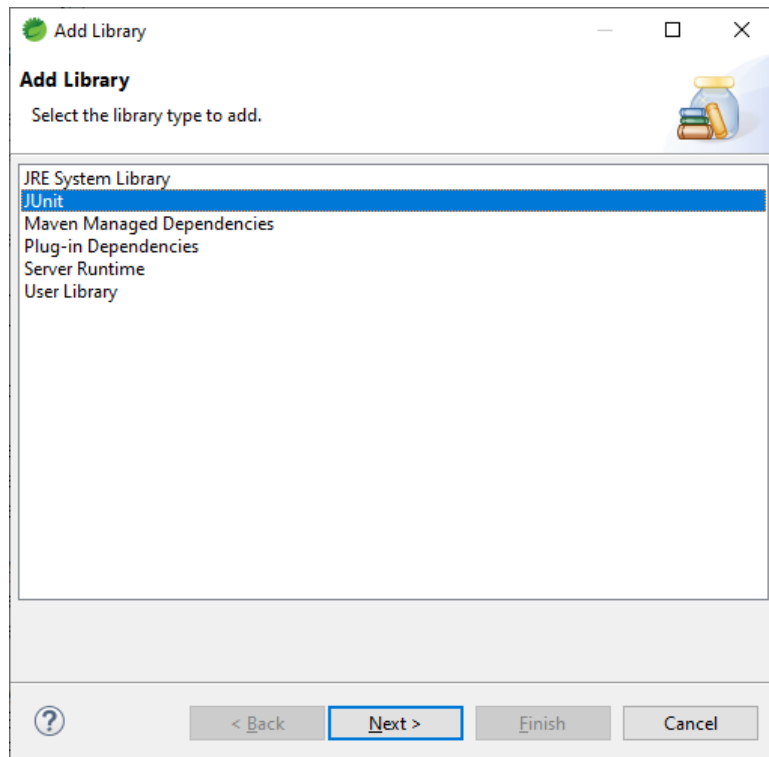
JUNIT COM STS

- Clique com o botão direito sobre o projeto e escolha *Build Path* e então a opção *Add Libraries ...*



JUNIT COM STS

- Selecione JUnit e clique em *Next* e, na próxima janela, clique em *Finish*



JUNIT COM STS

■ Crie os pacotes

- dominio
- principal

JUNIT COM STS

- Crie a classe `Usuario` dentro do pacote `dominio` com o código a seguir

```
1 package dominio;
2
3 public class Usuario {
4     private String nome;
5
6     public Usuario() { }
7
8     public Usuario(String nome) {
9         this.nome = nome;
10    }
11
12    public String getNome() {
13        return nome;
14    }
15
16    public void setNome(String nome) {
17        this.nome = nome;
18    }
19
20    @Override
21    public String toString() {
22        return "Usuario [nome=" + nome + "]";
23    }
24 }
```

JUNIT COM STS

- Crie a classe `Lance` dentro do pacote `dominio` com o código a seguir

```
1 package dominio;
2
3 public class Lance {
4     private Usuario usuario;
5     private double valor;
6
7     public Lance(Usuario usuario, double valor) {
8         this.usuario = usuario;
9         this.valor = valor;
10    }
11
12    public Usuario getUsuario() {
13        return usuario;
14    }
15
16    public double getValor() {
17        return valor;
18    }
19
20    @Override
21    public String toString() {
22        return "Lance [usuario=" + usuario + ", valor=" + valor + "];"
23    }
24 }
```


JUNIT COM STS

- Crie a classe `Leilao` dentro do pacote `dominio` com o código a seguir

```
1 package dominio;
2
3 import java.util.ArrayList;
4 import java.util.List;
5
6 public class Leilao {
7     private String descricao;
8     private List<Lance> lances;
9
10    public Leilao(String descricao) {
11        this.descricao = descricao;
12        this.lances = new ArrayList<Lance>();
13    }
14
15    public void propoe(Lance lance) {
16        lances.add(lance);
17    }
18
19    public String getDescricao() {
20        return descricao;
21    }
22
23    public List<Lance> getLances() {
24        return lances;
25    }
26 }
```

JUNIT COM STS

- Crie a classe `Avaliador` dentro do pacote `dominio` com o código a seguir
- Esse código irá informar qual o maior lance efetuado

```
1 package dominio;
2
3 public class Avaliador {
4     private double maiorLance = Double.NEGATIVE_INFINITY;
5
6     public void avalia(Leilao leilao) {
7         for (Lance lance : leilao.getLances()) {
8             if (lance.getValor() > maiorLance) {
9                 maiorLance = lance.getValor();
10            }
11        }
12    }
13
14    public double getMaiorLance() {
15        return maiorLance;
16    }
17 }
```

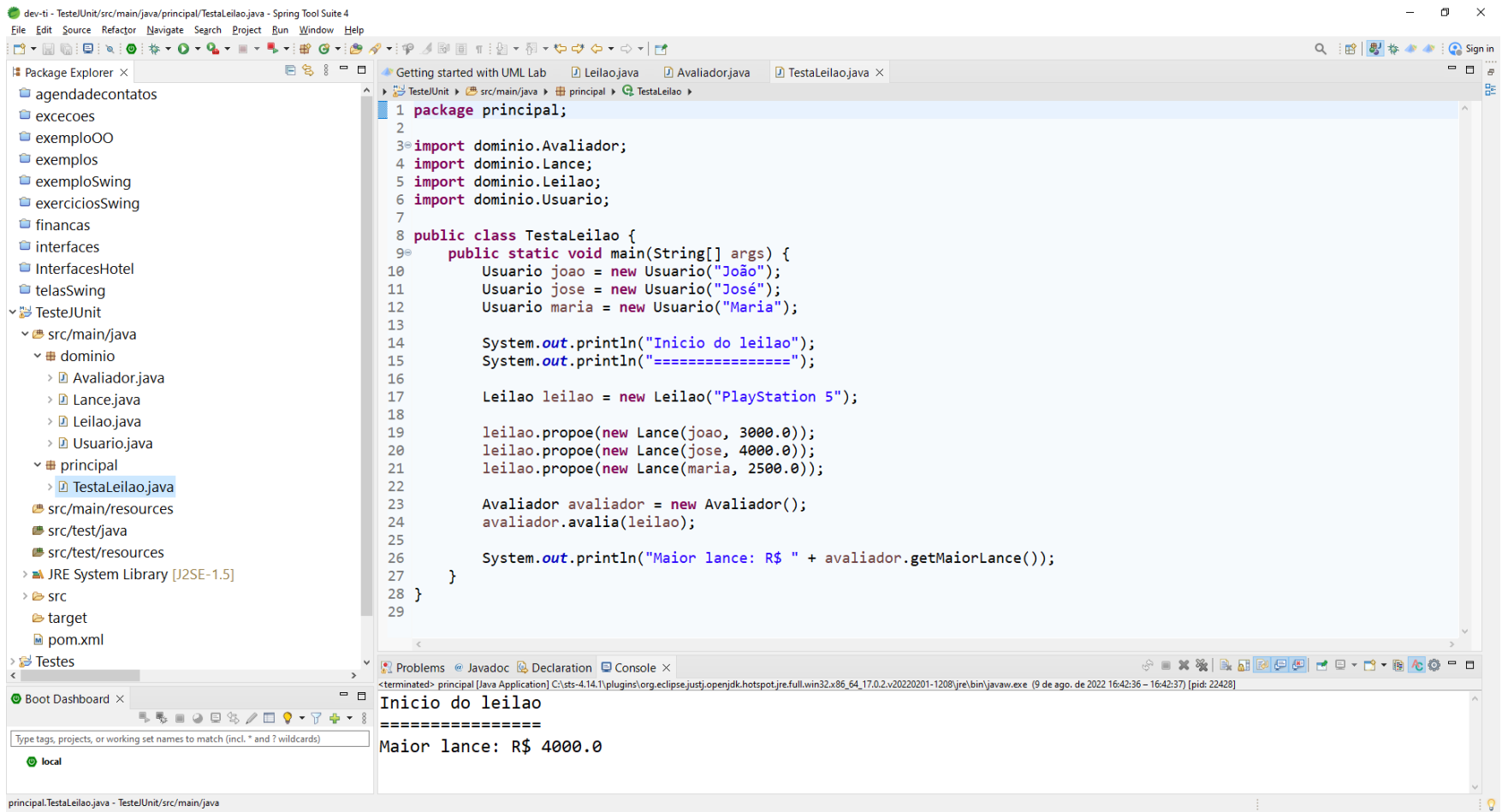
JUNIT COM STS

- Dentro do pacote `principal` crie a classe `TestaLeilao` com o código abaixo

```
1 package principal;
2
3 import dominio.Avaliador;
4 import dominio.Lance;
5 import dominio.Leilao;
6 import dominio.Usuario;
7
8 public class TestaLeilao {
9     public static void main(String[] args) {
10         Usuario joao = new Usuario("João");
11         Usuario jose = new Usuario("José");
12         Usuario maria = new Usuario("Maria");
13
14         System.out.println("Inicio do leilao");
15         System.out.println("=====");
16
17         Leilao leilao = new Leilao("PlayStation 5");
18
19         leilao.propoe(new Lance(joao, 3000.0));
20         leilao.propoe(new Lance(jose, 4000.0));
21         leilao.propoe(new Lance(maria, 2500.0));
22
23         Avaliador avaliador = new Avaliador();
24         avaliador.avalial(leilao);
25
26         System.out.println("Maior lance: R$ " + avaliador.getMaiorLance());
27     }
28 }
```

JUNIT COM STS

■ Ao executar o programa, o resultado correto deverá ser mostrado



```
1 package principal;
2
3 import dominio.Avaliador;
4 import dominio.Lance;
5 import dominio.Leilao;
6 import dominio.Usuario;
7
8 public class TestaLeilao {
9     public static void main(String[] args) {
10         Usuario joao = new Usuario("João");
11         Usuario jose = new Usuario("José");
12         Usuario maria = new Usuario("Maria");
13
14         System.out.println("Inicio do leilao");
15         System.out.println("=====");
16
17         Leilao leilao = new Leilao("PlayStation 5");
18
19         leilao.propoe(new Lance(joao, 3000.0));
20         leilao.propoe(new Lance(jose, 4000.0));
21         leilao.propoe(new Lance(maria, 2500.0));
22
23         Avaliador avaliador = new Avaliador();
24         avaliador.avaliao(leilao);
25
26         System.out.println("Maior lance: R$ " + avaliador.getMaiorLance());
27     }
28 }
29
```

Console Output:

```
<terminated> principal [Java Application] C:\sts-4.14.1\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64.17.0.2.v20220201-1208\jre\bin\javaw.exe (9 de ago. de 2022 16:42:36 - 16:42:37) [pid: 22428]
Inicio do leilao
=====
Maior lance: R$ 4000.0
```

JUNIT COM STS

- Na classe `Avaliador`, acrescente a funcionalidade de determinação do menor lance

```
1 package dominio;
2
3 public class Avaliador {
4     private double maiorLance = Double.NEGATIVE_INFINITY;
5     private double menorLance = Double.POSITIVE_INFINITY;
6
7     public void avalia(Leilao leilao) {
8         for (Lance lance : leilao.getLances()) {
9             if (lance.getValor() > maiorLance) {
10                 maiorLance = lance.getValor();
11             } else if (lance.getValor() < menorLance) {
12                 menorLance = lance.getValor();
13             }
14         }
15     }
16
17     public double getMaiorLance() {
18         return maiorLance;
19     }
20
21     public double getMenorLance() {
22         return menorLance;
23     }
24 }
```

adores:

IOVALE
E INOVAÇÃO VILLE DO RIO DO SUL

CENTRO DE
INOVAÇÃO
VIDEIRA



fapesc

GOV. DE
SANTA
CATARINA
SECRETARIA DE ESTADO
DO DESENVOLVIMENTO
ECONÔMICO SUSTENTÁVEL

DEV-TI

TEC

JUNIT COM STS

- Modifique agora o programa principal de modo que ele mostre também o menor lance

- Ao executá-lo novamente, o programa parece estar produzindo os resultados corretos

Início do leilao
=====
Maior lance: R\$ 4000.0
Menor lance: R\$ 2500.0

```
1 package principal;
2
3 import dominio.Avaliador;
4 import dominio.Lance;
5 import dominio.Leilao;
6 import dominio.Usuario;
7
8 public class TestaLeilao {
9     public static void main(String[] args) {
10         Usuario joao = new Usuario("João");
11         Usuario jose = new Usuario("José");
12         Usuario maria = new Usuario("Maria");
13
14         System.out.println("Início do leilao");
15         System.out.println("=====");
16
17         Leilao leilao = new Leilao("PlayStation 5");
18
19         leilao.propoe(new Lance(joao, 3000.0));
20         leilao.propoe(new Lance(jose, 4000.0));
21         leilao.propoe(new Lance(maria, 2500.0));
22
23         Avaliador avaliador = new Avaliador();
24         avaliador.avalia(leilao);
25
26         System.out.println("Maior lance: R$ " + avaliador.getMaiorLance());
27         System.out.println("Menor lance: R$ " + avaliador.getMenorLance());
28     }
29 }
```

UNICESP SISTEMA DE GESTÃO DE QUALIDADE

DEV-TI

SC
TEC

fapescc
FAPESP

GOV. DO ESTADO DE
SANTA CATARINA
SECRETARIA DE ESTADO
DO DESENVOLVIMENTO
ECONÔMICO E TECNOLÓGICO

JUNIT COM STS

- Antes de colocar o *software* em produção, altere o programa e faça mais um teste
- Mova a linha 21 para a linha 19, de forma a ficar como na imagem à direita

```
1 package principal;
2
3 import dominio.Avaliador;
4 import dominio.Lance;
5 import dominio.Leilao;
6 import dominio.Usuario;
7
8 public class Testaleilao {
9     public static void main(String[] args) {
10         Usuario joao = new Usuario("João");
11         Usuario jose = new Usuario("José");
12         Usuario maria = new Usuario("Maria");
13
14         System.out.println("Inicio do leilao");
15         System.out.println("=====");
16
17         Leilao leilao = new Leilao("PlayStation 5");
18
19         leilao.propoe(new Lance(joao, 3000.0));
20         leilao.propoe(new Lance(jose, 4000.0));
21         leilao.propoe(new Lance(maria, 2500.0));
22
23         Avaliador avaliador = new Avaliador();
24         avaliador.avalia(leilao);
25
26         System.out.println("Maior lance: R$ " + avaliador.getMaiorLance());
27         System.out.println("Menor lance: R$ " + avaliador.getMenorLance());
28     }
29 }
```

```
1 package principal;
2
3 import dominio.Avaliador;
4 import dominio.Lance;
5 import dominio.Leilao;
6 import dominio.Usuario;
7
8 public class Testaleilao {
9     public static void main(String[] args) {
10         Usuario joao = new Usuario("João");
11         Usuario jose = new Usuario("José");
12         Usuario maria = new Usuario("Maria");
13
14         System.out.println("Inicio do leilao");
15         System.out.println("=====");
16
17         Leilao leilao = new Leilao("PlayStation 5");
18
19         leilao.propoe(new Lance(maria, 2500.0));
20         leilao.propoe(new Lance(joao, 3000.0));
21         leilao.propoe(new Lance(jose, 4000.0));
22
23         Avaliador avaliador = new Avaliador();
24         avaliador.avalia(leilao);
25
26         System.out.println("Maior lance: R$ " + avaliador.getMaiorLance());
27         System.out.println("Menor lance: R$ " + avaliador.getMenorLance());
28     }
29 }
```

JUNIT COM STS

- Desta vez ele não produziu os resultados corretos 🤖

```
1 package principal;
2
3 import dominio.Avaliador;
4 import dominio.Lance;
5 import dominio.Leilao;
6 import dominio.Usuario;
7
8 public class TestaLeilao {
9     public static void main(String[] args) {
10         Usuario joao = new Usuario("João");
11         Usuario jose = new Usuario("José");
12         Usuario maria = new Usuario("Maria");
13
14         System.out.println("Inicio do leilao");
15         System.out.println("=====");
16
17         Leilao leilao = new Leilao("PlayStation 5");
18
19         leilao.propoe(new Lance(maria, 2500.0));
20         leilao.propoe(new Lance(joao, 3000.0));
21         leilao.propoe(new Lance(jose, 4000.0));
22
23         Avaliador avaliador = new Avaliador();
24         avaliador.avalial(leilao);
25
26         System.out.println("Maior lance: R$ " + avaliador.getMaiorLance());
27         System.out.println("Menor lance: R$ " + avaliador.getMenorLance());
28     }
29 }
30
```

Problems @ Javadoc Declaration Console X

<terminated> TestaLeilao [Java Application] C:\sts-4.14.1\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_17.0.2.v20220201-1208\jre\bin\javaw.exe (9 de

Inicio do leilao

=====

Maior lance: R\$ 4000.0

Menor lance: R\$ Infinity

JUNIT COM STS

- Um código com *bug* quase foi colocado em produção
 - Isso poderia ter sido evitado se fossem utilizadas técnicas de testes automatizados de *software*
 - Testes automatizados de *software* consistem em programas que testam outros programas
 - Se o resultado não for esperado, um alerta é emitido
 - Estes testes são realizados toda vez que o código é modificado
- Ao realizar um teste, normalmente os seguintes passos são realizados
 - Criação de um cenário
 - Execução da ação a ser testada
 - Verificação se o sistema se comportou da maneira esperada

JUNIT COM STS

- Modificando o código do programa principal e executando-o
- Como esperado, o primeiro teste retornou `true` e o segundo `false`

```
8 public class TestaLeilao {
9     public static void main(String[] args) {
10         // Parte 1: Cenário
11         Usuario joao = new Usuario("João");
12         Usuario jose = new Usuario("José");
13         Usuario maria = new Usuario("Maria");
14
15         System.out.println("Inicio do leilao");
16         System.out.println("=====");
17
18         Leilao leilao = new Leilao("PlayStation 5");
19
20         leilao.propoe(new Lance(maria, 2500.0));
21         leilao.propoe(new Lance(joao, 3000.0));
22         leilao.propoe(new Lance(jose, 4000.0));
23
24         // Parte 2: Ação
25         Avaliador avaliador = new Avaliador();
26         avaliador.avaliao(leilao);
27
28         // Parte 3: Verificação
29         double maiorLanceEsperado = 4000;
30         double menorLanceEsperado = 2500;
31
32         boolean resultadoMaiorLanceEsperado = (maiorLanceEsperado == avaliador.getMaiorLance());
33         boolean resultadoMenorLanceEsperado = (menorLanceEsperado == avaliador.getMenorLance());
34
35         System.out.println("Maior lance: " + resultadoMaiorLanceEsperado);
36         System.out.println("Menor lance: " + resultadoMenorLanceEsperado);
37     }
38 }
```

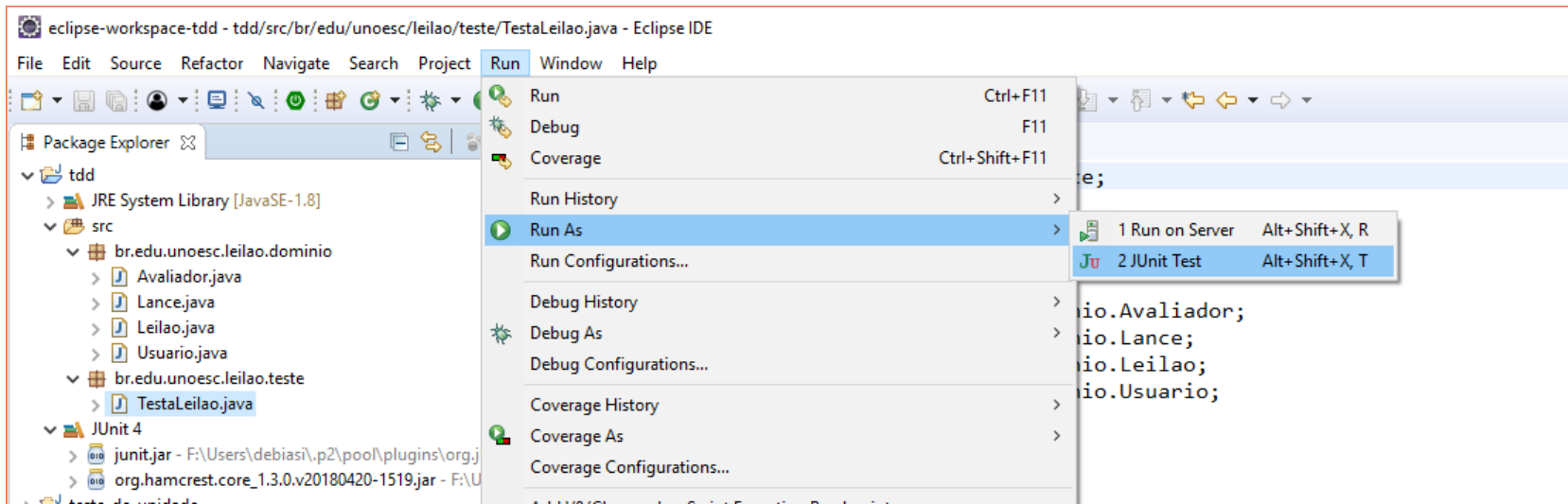
JUNIT COM STS

- O próximo passo é adaptar o código ao *framework* JUnit
- As modificações são mostradas ao lado
 - Deve-se usar a anotação `@Test` da classe `org.junit.jupiter.api.Test`
 - O método deve ser alterado para `public void`
 - O nome do método deve refletir o que se está testando
 - O método a ser usado deve ser o `assertEquals()` da classe `org.junit.jupiter.api.Assertions.assertEquals`

```
1 package principal;
2
3 import static org.junit.jupiter.api.Assertions.assertEquals;
4
5 import org.junit.jupiter.api.Test;
6
7 import dominio.Avaliador;
8 import dominio.Lance;
9 import dominio.Leilao;
10 import dominio.Usuario;
11
12 public class TestaLeilao {
13     @Test
14     public void testarLancesOrdemCrescente() {
15         // Parte 1: Cenário
16         Usuario joao = new Usuario("João");
17         Usuario jose = new Usuario("José");
18         Usuario maria = new Usuario("Maria");
19
20         Leilao leilao = new Leilao("PlayStation 5");
21
22         leilao.propoe(new Lance(maria, 2500.0));
23         leilao.propoe(new Lance(joao, 3000.0));
24         leilao.propoe(new Lance(jose, 4000.0));
25
26         // Parte 2: Ação
27         Avaliador avaliador = new Avaliador();
28         avaliador.avalia(leilao);
29
30         // Parte 3: Verificação
31         double maiorLanceEsperado = 4000;
32         double menorLanceEsperado = 2500;
33
34         assertEquals(maiorLanceEsperado, avaliador.getMaiorLance());
35         assertEquals(menorLanceEsperado, avaliador.getMenorLance());
36     }
37 }
```

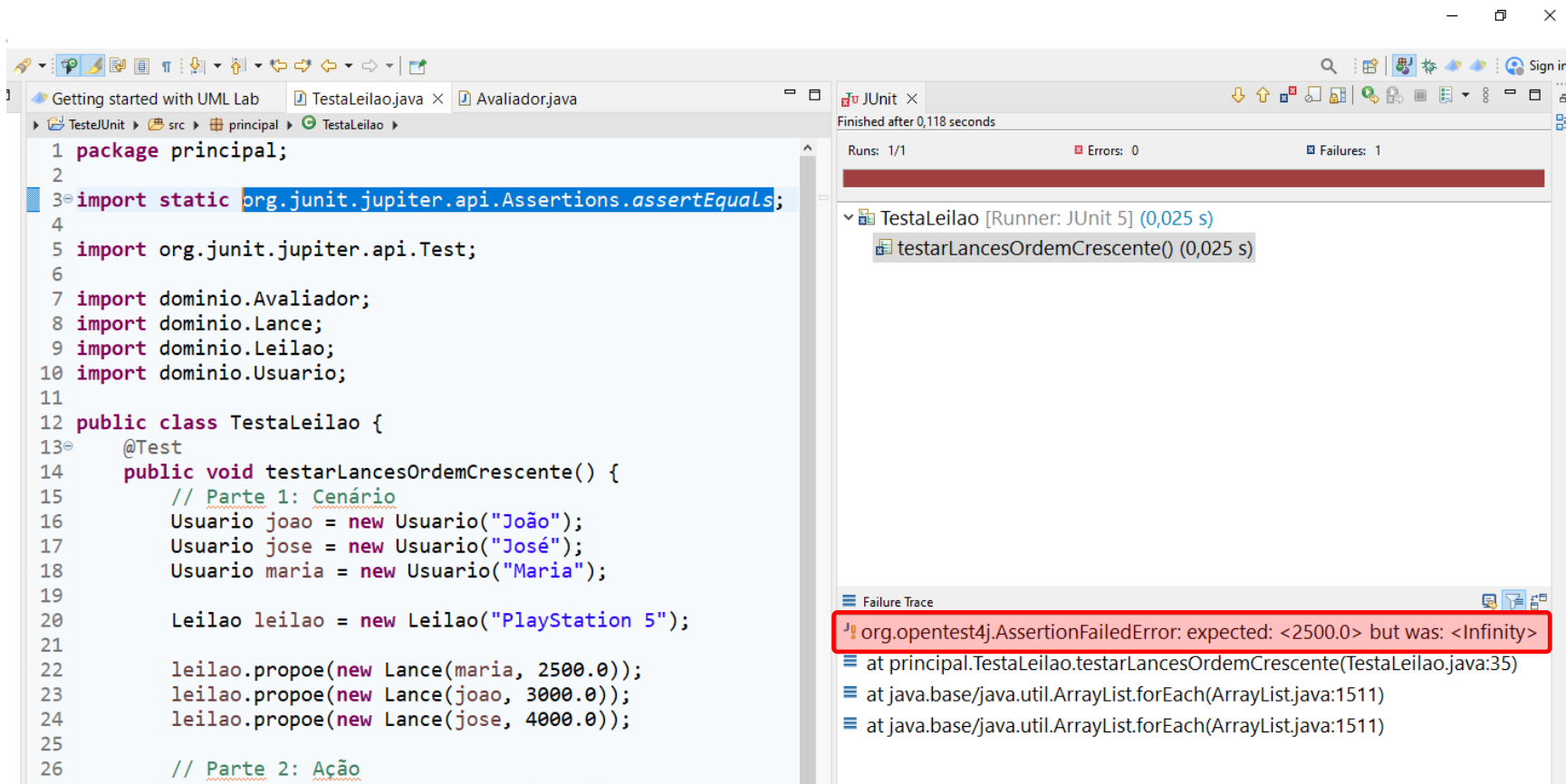
JUNIT COM STS

- O arquivo deve ser executado não como um aplicativo, mas como uma unidade de teste JUnit



JUNIT COM STS

■ Abaixo o resultado da execução



The screenshot displays the STS IDE interface. On the left, the 'TesteLeilao.java' file is open, showing the following code:

```
1 package principal;
2
3 import static org.junit.jupiter.api.Assertions.assertEquals;
4
5 import org.junit.jupiter.api.Test;
6
7 import dominio.Avaliador;
8 import dominio.Lance;
9 import dominio.Leilao;
10 import dominio.Usuario;
11
12 public class TesteLeilao {
13     @Test
14     public void testarLancesOrdemCrescente() {
15         // Parte 1: Cenário
16         Usuario joao = new Usuario("João");
17         Usuario jose = new Usuario("José");
18         Usuario maria = new Usuario("Maria");
19
20         Leilao leilao = new Leilao("PlayStation 5");
21
22         leilao.propoe(new Lance(maria, 2500.0));
23         leilao.propoe(new Lance(joao, 3000.0));
24         leilao.propoe(new Lance(jose, 4000.0));
25
26         // Parte 2: Ação
```

On the right, the JUnit test runner shows the execution results. The test 'testarLancesOrdemCrescente()' passed successfully. The failure trace is empty, indicating no failures.

JUNIT COM STS

■ Corrigindo o *bug* na classe Avaliador

```
1 package dominio;
2
3 public class Avaliador {
4     private double maiorLance = Double.NEGATIVE_INFINITY;
5     private double menorLance = Double.POSITIVE_INFINITY;
6
7     public void avalia(Leilao leilao) {
8         for (Lance lance : leilao.getLances()) {
9             if (lance.getValor() > maiorLance) {
10                 maiorLance = lance.getValor();
11             }
12
13             if (lance.getValor() < menorLance) {
14                 menorLance = lance.getValor();
15             }
16         }
17     }
18
19     public double getMaiorLance() {
20         return maiorLance;
21     }
22
23     public double getMenorLance() {
24         return menorLance;
25     }
26 }
```

JUNIT COM STS

■ Executando novamente os testes unitários

The screenshot displays the Spring Tool Suite 4 (STS) IDE interface. On the left, the Package Explorer shows the project structure, including the 'TesteJunit' package and the 'TestaLeilao.java' file. The main editor window shows the source code of 'TestaLeilao.java', which includes imports for 'dominio.Avaliador', 'dominio.Lance', 'dominio.Leilao', and 'dominio.Usuario'. The code defines a 'TestaLeilao' class with a '@Test' method 'testarLancesOrdemCrescente()'. This method sets up a scenario with three users (João, José, Maria) and a leilao (PlayStation 5), then performs assertions on the leilao's properties. The right-hand side of the IDE shows the JUnit runner output, indicating that the tests passed successfully after 0.123 seconds, with 0 errors and 0 failures. The console at the bottom shows the execution details, including the path to the Java runtime and the PID of the JVM.

```
6
7 import dominio.Avaliador;
8 import dominio.Lance;
9 import dominio.Leilao;
10 import dominio.Usuario;
11
12 public class TestaLeilao {
13     @Test
14     public void testarLancesOrdemCrescente() {
15         // Parte 1: Cenário
16         Usuario joao = new Usuario("João");
17         Usuario jose = new Usuario("José");
18         Usuario maria = new Usuario("Maria");
19
20         Leilao leilao = new Leilao("PlayStation 5");
21
22         leilao.propoe(new Lance(maria, 2500.0));
23         leilao.propoe(new Lance(joao, 3000.0));
24         leilao.propoe(new Lance(jose, 4000.0));
25
26         // Parte 2: Ação
27         Avaliador avaliador = new Avaliador();
28         avaliador.avalua(leilao);
29
30         // Parte 3: Verificação
31         double maiorLanceEsperado = 4000;
32         double menorLanceEsperado = 2500;
33
34         assertEquals(maiorLanceEsperado, avaliador.getMaiorLance());
35         assertEquals(menorLanceEsperado, avaliador.getMenorLance());
36     }
37 }
38
```

JUnit 5 [Runner: JUnit 5] (0,018 s)

Finished after 0,123 seconds

Runs: 1/1 Errors: 0 Failures: 0

Failure Trace

Problems Javadoc Declaration Console

<terminated> TestaLeilao (2) [JUnit] C:\sts-4.14.1\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64.17.0.2.v20220201-1208\jre\bin\javaw.exe (9 de ago. de 2022 17:44:53 - 17:44:54) [pid: 7724]