

Consolidando o seu conhecimento

Chegou a hora de você pôr em prática o que foi visto na aula. Para isso, execute os passos listados abaixo.

1) Abra a classe `Avaliador` e adicione o seguinte código no início do método `avalia`:

```
if (empty($leilao->getLances())) {
    throw new \DomainException('Não é possível avaliar leilão vazio');
}
```

2) Agora, implemente o teste na classe `AvaliadorTest`. Adicione um novo método de teste:

```
public function testLeilaoVazioNaoPodeSerAvaliado()
{
    $this->expectException(\DomainException::class);
    $this->expectExceptionMessage('Não é possível avaliar leilão vazio');

    $leilao = new Leilao('Fusca Azul');
    $this->leiloeiro->avalia($leilao);
}
```

3) Execute os testes na linha de comando:

```
vendor\bin\phpunit
```

4) De volta ao seu projeto, abra a classe `Leilao` e procure o método `recebeLance`. Altere o método para jogar um `DomainException` em cada `if` que já existe.

Segue o código completo do método:

```
public function recebeLance(Lance $lance)
{
    if (!empty($this->lances) && $this->ehDoUltimoUsuario($lance)) {
        throw new \DomainException('Usuário não pode propor 2 lances consecutivos');
    }

    $totalLancesUsuario = $this
        ->quantidadeLancesPorUsuario($lance->getUsuario());
    if ($totalLancesUsuario >= 5) {
        throw new \DomainException('Usuário não pode propor mais de 5 lances por leilão');
    }

    $this->lances[] = $lance;
}
```

5) Agora, abra a classe `LeilaoTest` e procure o método `testLeilaoNaoDeveReceberLancesRepetidos`. Nele, apague todos os `asserts` e adicione o uso da exceção no início:

```
$this->expectException(\DomainException::class);  
$this->expectExceptionMessage('Usuário não pode propor 2 lances consecutivos');
```

6) Faça o mesmo no método `testLeilaoNaoDeveAceitarMaisDe5LancesPorUsuario` (ainda na classe `LeilaoTest`). Isto é, apagar o código com os `asserts` e adicionar o uso da exceção:

```
$this->expectException(\DomainException::class);  
$this->expectExceptionMessage('Usuário não pode propor mais de 5 lances por leilão');
```

7) Execute os testes na linha de comando:

```
vendor\bin\phpunit
```

8) (Opcional) Implemente as funcionalidades e testes *Leilao Vazio Nao Pode Ser Avaliado* e *Leilao Finalizado Nao Pode Ser Avaliado*.

Tente escrever os testes primeiro, para depois implementar as regras. Mão na massa!