1) Crie uma nova classe de teste, chamada LeilaoTest.php, na pasta tests\Models:

class LeilaoTest extends TestCase

{

}

2) Dentro da classe LeilaoTest, adicione o provedor de dados e o primeiro teste:

/\*\*

\* @dataProvider geraLances

\*/

public function testLeilaoDeveReceberLances(

int $qtdLances,

Leilao $leilao,

array $valores

) {

static::assertCount($qtdLances, $leilao->getLances());

foreach ($valores as $i => $valorEsperado) {

static::assertEquals($valorEsperado, $leilao->getLances()[$i]->getValor());

}

}

public function geraLances()

{

$joao = new Usuario('João');

$maria = new Usuario('Maria');

$leilaoCom2Lances = new Leilao('Fiat 147 0KM');

$leilaoCom2Lances->recebeLance(new Lance($joao, 1000));

$leilaoCom2Lances->recebeLance(new Lance($maria, 2000));

$leilaoCom1Lance = new Leilao('Fusca 1972 0KM');

$leilaoCom1Lance->recebeLance(new Lance($maria, 5000));

return [

'2-lances' => [2, $leilaoCom2Lances, [1000, 2000]],

'1-lance' => [1, $leilaoCom1Lance, [5000]]

];

}

3) Execute os testes na linha de comando:

vendor\bin\phpunit

4) De volta ao projeto, adicione um novo teste na classe LeilaoTest, que garante que não existe um lance repetido:

public function testLeilaoNaoDeveReceberLancesRepetidos()

{

$leilao = new Leilao('Variante');

$ana = new Usuario('Ana');

$leilao->recebeLance(new Lance($ana, 1000));

$leilao->recebeLance(new Lance($ana, 1500));

static::assertCount(1, $leilao->getLances());

static::assertEquals(1000, $leilao->getLances()[0]->getValor());

}

5) Execute os testes na linha de comando e veja o teste falhando:

vendor\bin\phpunit

O teste falha pois você ainda não implementou nada!

7) Faça a implementação da funcionalidade, seguindo o ciclo de TDD. Tente usar uma implementação simples e refatore o seu código.

8) Agora, implemente o teste "o leilão não deve aceitar mais de 5 lances por usuário". Adicione o seguinte método na classe LeilaoTest:

public function testLeilaoNaoDeveAceitarMaisDe5LancesPorUsuario()

{

$leilao = new Leilao('Brasília Amarela');

$joao = new Usuario('João');

$maria = new Usuario('Maria');

$leilao->recebeLance(new Lance($joao, 1000));

$leilao->recebeLance(new Lance($maria, 1500));

$leilao->recebeLance(new Lance($joao, 2000));

$leilao->recebeLance(new Lance($maria, 2500));

$leilao->recebeLance(new Lance($joao, 3000));

$leilao->recebeLance(new Lance($maria, 3500));

$leilao->recebeLance(new Lance($joao, 4000));

$leilao->recebeLance(new Lance($maria, 4500));

$leilao->recebeLance(new Lance($joao, 5000));

$leilao->recebeLance(new Lance($maria, 5500));

$leilao->recebeLance(new Lance($joao, 6000));

static::assertCount(10, $leilao->getLances());

static::assertEquals(5500, $leilao->getLances()[array\_key\_last($leilao->getLances())]->getValor());

}

9) Execute os testes na linha de comando e veja o teste falhando novamente:

vendor\bin\phpunit

10) Implemente a funcionalidade na classe Leilao e faça com que o teste passe!