

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) Extraia o cenário de testes para um *data provider*:

```
public function leiloes()
{
    $naoFinalizado = new Leilao('Variante 0Km');
    $finalizado = new Leilao('Fiat 147 0Km');
    $finalizado->finaliza();

    return [
        [
            [$naoFinalizado, $finalizado]
        ]
    ];
}
```

2) No método de testes que temos até agora, passe a utilizar este *data provider*:

```
/**
 * @dataProvider leiloes
 */
public function testBuscaLeiloesNaoFinalizados(array $leiloes)
{
    // arrange
    $leilaoDao = new LeilaoDao(self::$pdo);
    foreach ($leiloes as $leilao) {
        $leilaoDao->salva($leilao);
    }

    // act
    $leiloes = $leilaoDao->recuperarNaoFinalizados();

    // assert
    self::assertCount(1, $leiloes);
    self::assertContainsOnlyInstancesOf(Leilao::class, $leiloes);
    self::assertSame(
        'Variante 0Km',
        $leiloes[0]->recuperarDescricao()
    );
    self::assertFalse($leiloes[0]->estaFinalizado());
}
```

3) Crie um novo teste para garantir que é possível recuperar corretamente os leilões finalizados:

```
/**
 * @dataProvider leiloes
 */
public function testBuscaLeiloesFinalizados(array $leiloes)
```

```
public function testBuscaLeiloesFinalizados(array $leiloes)
{
    // arrange
    $leilaoDao = new LeilaoDao(self::$pdo);
    foreach ($leiloes as $leilao) {
        $leilaoDao->salva($leilao);
    }

    // act

    $leiloes = $leilaoDao->recuperarFinalizados();

    // assert
    self::assertCount(1, $leiloes);
    self::assertContainsOnlyInstancesOf(Leilao::class, $leiloes);
    self::assertSame(
        'Fiat 147 0Km',
        $leiloes[0]->recuperarDescricao()
    );
    self::assertTrue($leiloes[0]->estaFinalizado());
}
```