RM82245 – Rafael Eidi Makiyama

**Código JUnit**

package br.fiap.exercicios;

import static org.junit.jupiter.api.Assertions.\*;

import org.junit.jupiter.api.Test;

import junit.framework.Assert;

class JulgamentoPrisioneiroTest {

@Test

void testCalculaPenaInocencia() {

JulgamentoPrisioneiro testeJulgamento = new JulgamentoPrisioneiro();

int respostaEsperada = 0;

int resultadoFinal = testeJulgamento.calculaPena("Culpado", "Inocente");

assertEquals(respostaEsperada, resultadoFinal);

}

@Test

void testCalculaPenaCondenacaoMutua() {

JulgamentoPrisioneiro testeJulgamento = new JulgamentoPrisioneiro();

int respostaEsperada = 5;

int resultadoFinal = testeJulgamento.calculaPena("Culpado", "Culpado");

assertEquals(respostaEsperada, resultadoFinal);

}

@Test

void testCalculaPenaCondenacaoIndividual() {

JulgamentoPrisioneiro testeJulgamento = new JulgamentoPrisioneiro();

int respostaEsperada = 10;

int resultadoFinal = testeJulgamento.calculaPena("Inocente", "Culpado");

assertEquals(respostaEsperada, resultadoFinal);

}

@Test

void testCalculaPenaCondenacaoCumplices() {

JulgamentoPrisioneiro testeJulgamento = new JulgamentoPrisioneiro();

int respostaEsperada = 1;

int resultadoFinal = testeJulgamento.calculaPena("Inocente", "Inocente");

assertEquals(respostaEsperada, resultadoFinal);

}

}

**PRIMEIRO TESTE**

A JUnit não reconheceu nenhum método que recebe 2 strings.

**SOLUÇÃO**

**public** **int** calculaPena(String respostaPrisioneiroA, String respostaPrisioneiroB)

**SEGUNDO TESTE**

As lógicas foram feitas baseado num enum que não existe e nem recebe via parâmetro.

**SOLUÇÃO**

**if** (respostaPrisioneiroA == "Culpado") {

**if** (respostaPrisioneiroB == "Culpado") {

**return** PENA\_CONDENACAO\_MUTUA;

} **else** {

**return** PENA\_INOCENCIA;

}

}

**else** {

**if** (respostaPrisioneiroB == "Culpado") {

**TERCEIRO TESTE**

Era esperado a resposta 0 para Inocente e foi retornado 13.

**SOLUÇÃO**

**private** **int** PENA\_INOCENCIA = 0;

**QUARTO TESTE**

Era esperado a resposta 5 para Condenação Mútua e foi retornado 15.

**SOLUÇÃO**

**private** **int** PENA\_CONDENACAO\_MUTUA = 5;

**QUINTO TESTE**

Era esperado a resposta 1 para Condenação de Cúmplices e foi retornado 11.

**SOLUÇÃO**

**private** **int** PENA\_CONDENACAO\_CUMPLICES = 1;

**Versão Documento 1.11**

**Classe JAVA Corrigida**

**package** br.fiap.exercicios;

**public** **class** JulgamentoPrisioneiro {

**private** **int** PENA\_INOCENCIA = 0;

**private** **int** PENA\_CONDENACAO\_MUTUA = 5;

**private** **int** PENA\_CONDENACAO\_INDIVIDUAL = 10;

**private** **int** PENA\_CONDENACAO\_CUMPLICES = 1;

**public** **int** calculaPena(String respostaPrisioneiroA, String respostaPrisioneiroB) {

**if** (respostaPrisioneiroA == "Culpado") {

**if** (respostaPrisioneiroB == "Culpado") {

**return** PENA\_CONDENACAO\_MUTUA;

} **else** {

**return** PENA\_INOCENCIA;

}

} **else** {

**if** (respostaPrisioneiroB == "Culpado") {

**return** PENA\_CONDENACAO\_INDIVIDUAL;

} **else** {

**return** PENA\_CONDENACAO\_CUMPLICES;

}

}

}

}

**Classe JUnit Final**

package br.fiap.exercicios;

import static org.junit.jupiter.api.Assertions.\*;

import org.junit.jupiter.api.Test;

import junit.framework.Assert;

class JulgamentoPrisioneiroTest {

@Test

void testCalculaPenaInocencia() {

JulgamentoPrisioneiro testeJulgamento = new JulgamentoPrisioneiro();

int respostaEsperada = 0;

int resultadoFinal = testeJulgamento.calculaPena("Culpado", "Inocente");

assertEquals(respostaEsperada, resultadoFinal);

}

@Test

void testCalculaPenaCondenacaoMutua() {

JulgamentoPrisioneiro testeJulgamento = new JulgamentoPrisioneiro();

int respostaEsperada = 5;

int resultadoFinal = testeJulgamento.calculaPena("Culpado", "Culpado");

assertEquals(respostaEsperada, resultadoFinal);

}

@Test

void testCalculaPenaCondenacaoIndividual() {

JulgamentoPrisioneiro testeJulgamento = new JulgamentoPrisioneiro();

int respostaEsperada = 10;

int resultadoFinal = testeJulgamento.calculaPena("Inocente", "Culpado");

assertEquals(respostaEsperada, resultadoFinal);

}

@Test

void testCalculaPenaCondenacaoCumplices() {

JulgamentoPrisioneiro testeJulgamento = new JulgamentoPrisioneiro();

int respostaEsperada = 1;

int resultadoFinal = testeJulgamento.calculaPena("Inocente", "Inocente");

assertEquals(respostaEsperada, resultadoFinal);

}

}