Nome: Victor Laruccia Ramos

RM: 79842

1° Erro na classe JulgamentoPrisioneiro que impede rodar o código: Parâmetros do método calculaPena() estavam com o tipo Resposta, que não existe. Foram alterados para o tipo String.

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

**private** **int** PENA\_INOCENCIA = 10;

**private** **int** PENA\_CONDENACAO\_MUTUA = 15;

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

**private** **int** PENA\_CONDENACAO\_CUMPLICES = 11;

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

**if** (respostaPrisioneiroA == Resposta.DELACAO) {

**if** (respostaPrisioneiroB == Resposta.DELACAO) {

**return** PENA\_CONDENACAO\_MUTUA;

} **else** {

**return** PENA\_INOCENCIA;

}

} **else** {

**if** (respostaPrisioneiroB == Resposta.DELACAO) {

**return** PENA\_CONDENACAO\_INDIVIDUAL;

} **else** {

**return** PENA\_CONDENACAO\_CUMPLICES;

}

}

}

}

2° Erro na classe JulgamentoPrisioneiro que impede rodar o código: nos ifs as respostas do prisioneiro estavam com erro, alterei as respostas para “Culpado” conforme a lógica do código portugol indica e o erro foi removido.

Também se trata do código corrigido da classe JulgamentoPrisioneiro.

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

**private** **int** PENA\_INOCENCIA = 10;

**private** **int** PENA\_CONDENACAO\_MUTUA = 15;

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

**private** **int** PENA\_CONDENACAO\_CUMPLICES = 11;

**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 de testes JUNIT funcionando perfeitamente, após a correção dos erros citados anteriormente:

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

**import** org.junit.jupiter.api.Test;

**class** JulgamentoPrisioneiroTeste {

@Test

**public** **void** testeDoisCulpados() {

JulgamentoPrisioneiro j = **new** JulgamentoPrisioneiro();

**int** resultadoReal = j.calculaPena("Culpado", "Culpado");

**int** resultadoEsperado = 15;

*assertEquals*(resultadoEsperado, resultadoReal);

}

@Test

**public** **void** testeCulpadoInocente() {

JulgamentoPrisioneiro j = **new** JulgamentoPrisioneiro();

**int** resultadoReal = j.calculaPena("Culpado", "Inocente");

**int** resultadoEsperado = 10;

*assertEquals*(resultadoEsperado, resultadoReal);

}

@Test

**public** **void** testeInocenteCulpado() {

JulgamentoPrisioneiro j = **new** JulgamentoPrisioneiro();

**int** resultadoReal = j.calculaPena("Inocente", "Culpado");

**int** resultadoEsperado = 10;

*assertEquals*(resultadoEsperado, resultadoReal);

}

@Test

**public** **void** testeInocenteInocente() {

JulgamentoPrisioneiro j = **new** JulgamentoPrisioneiro();

**int** resultadoReal = j.calculaPena("Inocente", "Inocente");

**int** resultadoEsperado = 11;

*assertEquals*(resultadoEsperado, resultadoReal);

}

}

Classe original do JulgamentoPrisioneiro, antes da correção dos erros:

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

**private** **int** PENA\_INOCENCIA = 10;

**private** **int** PENA\_CONDENACAO\_MUTUA = 15;

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

**private** **int** PENA\_CONDENACAO\_CUMPLICES = 11;

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

**if** (respostaPrisioneiroA == Resposta.DELACAO) {

**if** (respostaPrisioneiroB == Resposta.DELACAO) {

**return** PENA\_CONDENACAO\_MUTUA;

} **else** {

**return** PENA\_INOCENCIA;

}

} **else** {

**if** (respostaPrisioneiroB == Resposta.DELACAO) {

**return** PENA\_CONDENACAO\_INDIVIDUAL;

} **else** {

**return** PENA\_CONDENACAO\_CUMPLICES;

}

}

}

}

Obs.: Não foram detectados erros dentro da classe JUNIT, apenas na classe JulgamentoPrisioneiro.