

08

Mão na massa: Tratamento de exceções

Vamos forçar agora uma exceção em nossa classe `Fluxo` do projeto `java-pilha`.

1) Dentro do `for` do `metodo2()`, faça o seguinte:

```
for(int i = 1; i <= 5; i++) {
    System.out.println(i);
    int a = i / 0;
}
```

Isso deverá retornar uma `Arithmeticption`!

2) Para evitar que a exceção caia na nossa pilha, vamos utilizar o bloco `try` e `catch`:

```
try{
    int a = i / 0;
} catch(ArithmeticException ex) {
    System.out.println("ArithmeticException");
}
```

3) Porém podemos fazer a utilização do `try` e `catch` durante a chamada do `metodo2()`, dentro do `metodo1()`, ficando do seguinte modo:

```
try{
    metodo2();
} catch(ArithmeticException ex) {
    System.out.println("ArithmeticException");
}
```

4) Porém podemos subir mais uma vez o nosso `try` e `catch` para o nosso método `main`, no momento de chamada do `metodo1()`. Além disso, podemos definir algumas coisas de nossa exceção, como pegar a mensagem:

```
try{
    metodo1();
} catch(ArithmeticException ex) {
    String msg = ex.getMessage();
    System.out.println("ArithmeticException " + msg);
}
```

5) Outra coisa que podemos fazer é pegar o rastro da nossa exceção:

```
try{
    metodo1();
} catch(ArithmeticException ex) {
    String msg = ex.getMessage();
    System.out.println("ArithmeticException " + msg);
}
```

```
    ex.printStackTrace();
}
```

6) Agora vamos fazer outro teste. Para isso crie a classe `Conta` do seguinte modo:

```
public class Conta {  
  
    void deposita() {}  
  
}
```

7) E dentro do `for` do `metodo2()`, faremos o seguinte teste:

```
for(int i = 1; i <= 5; i++) {  
    System.out.println(i);  
    Conta c = null;  
    c.deposita();  
}
```

8) Mas a exce  o causada    uma `NullPointerException`, para pegarmos essa exce  o, precisamos coloc  -la dentro do `catch` no m  todo `main`:

```
try {  
    metodo1();  
} catch(ArithmeticException | NullPointerException ex) {  
    String msg = ex.getMessage();  
    System.out.println("Exception " + msg);  
    ex.printStackTrace();  
}
```