Test Case

Class under test: AddBookController

# Check ISBN:

* Input: ISBN (String).
* Output: result (boolean).

## Black box test:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test Case | 1 | 2 | 3 | 4 | 5 | 6 |
| String length | 10-13 | 10-13 | < 10 | < 10 | > 13 | > 13 |
| All characters | Are Digits | Not Digits | Are Digits | Not Digits | Are Digits | Not Digits |
| Output | true | false | false | false | false | false |

@Test  
public void checkISBNTestCase1() throws Exception {  
 *assertEquals*(true, addBookController.checkISBN("12345678901"));  
}  
  
@Test  
public void checkISBNTestCase2() throws Exception {  
 *assertEquals*(false, addBookController.checkISBN("123456789abc"));  
}  
  
@Test  
public void checkISBNTestCase3() throws Exception {  
 *assertEquals*(false, addBookController.checkISBN("12345678"));  
}  
  
@Test  
public void checkISBNTestCase4() throws Exception {  
 *assertEquals*(false, addBookController.checkISBN("123456ab"));  
}  
  
@Test  
public void checkISBNTestCase5() throws Exception {  
 *assertEquals*(false, addBookController.checkISBN("12345689012345"));  
}  
  
@Test  
public void checkISBNTestCase6() throws Exception {  
 *assertEquals*(false, addBookController.checkISBN("1234567890123ab"));  
}

## White box test:

* C0 100% Coverage test:

@Test  
public void checkISBNTestCase1() throws Exception {  
 *assertEquals*(true, addBookController.checkISBN("12345678901"));  
}

* C1 100% Coverage test:

@Test  
public void checkISBNTestCase1() throws Exception {  
 *assertEquals*(true, addBookController.checkISBN("12345678901"));  
}  
  
@Test  
public void checkISBNTestCase2() throws Exception {  
 *assertEquals*(false, addBookController.checkISBN("123456789abc"));  
}  
  
@Test  
public void checkISBNTestCase3() throws Exception {  
 *assertEquals*(false, addBookController.checkISBN("12345678"));  
}  
  
@Test  
public void checkISBNTestCase4() throws Exception {  
 *assertEquals*(false, addBookController.checkISBN("123456ab"));  
}  
  
@Test  
public void checkISBNTestCase5() throws Exception {  
 *assertEquals*(false, addBookController.checkISBN("12345689012345"));  
}  
  
@Test  
public void checkISBNTestCase6() throws Exception {  
 *assertEquals*(false, addBookController.checkISBN("1234567890123ab"));  
}

# Check copy type:

* Input: type (String).
* Output: result (boolean).

## Black box test:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Case | 1 | 2 | 3 | 4 |
| String length | = 1 | = 1 | != 1 | != 1 |
| First char | ‘B’ || ‘R’ | Not ‘B’ & Not ‘R’ | ‘B’ || ‘R’ | Not ‘B’ & Not ‘R’ |
| Output | true | false | false | false |

@Test  
public void checkTypeTestCase1() throws Exception {  
 *assertEquals*(true, addBookController.checkType("B"));  
}  
  
@Test  
public void checkTypeTestCase2() throws Exception {  
 *assertEquals*(false, addBookController.checkType("A"));  
}  
  
@Test  
public void checkTypeTestCase3() throws Exception {  
 *assertEquals*(false, addBookController.checkType("B12ab"));  
}  
  
@Test  
public void checkTypeTestCase4() throws Exception {  
 *assertEquals*(false, addBookController.checkType(""));  
}

## White box test:

* C0 100% Coverage test:

@Test  
public void checkTypeTestCase1() throws Exception {  
 *assertEquals*(true, addBookController.checkType("B"));  
}

* C1 100% Coverage test:

@Test  
public void checkTypeTestCase1() throws Exception {  
 *assertEquals*(true, addBookController.checkType("B"));  
}  
  
@Test  
public void checkTypeTestCase2() throws Exception {  
 *assertEquals*(false, addBookController.checkType("A"));  
}  
  
@Test  
public void checkTypeTestCase3() throws Exception {  
 *assertEquals*(false, addBookController.checkType("B12ab"));  
}  
  
@Test  
public void checkTypeTestCase4() throws Exception {  
 *assertEquals*(false, addBookController.checkType(""));  
}

# Check copy price:

* Input: price (String).
* Output: result (boolean).

## Black box test:

|  |  |  |
| --- | --- | --- |
| Test Case | 1 | 2 |
| All characters | Are Digits | Not Digits |
| Output | true | false |

@Test  
public void checkPriceTestCase1() throws Exception {  
 *assertEquals*(true, addBookController.checkPrice("700000"));  
}  
  
@Test  
public void checkPriceTestCase2() throws Exception {  
 *assertEquals*(false, addBookController.checkPrice("700abc00"));  
}

## White box test:

* C0 100% Coverage test:

@Test  
public void checkPriceTestCase1() throws Exception {  
 *assertEquals*(true, addBookController.checkPrice("700000"));  
}

* C1 100% Coverage test:

@Test  
public void checkPriceTestCase1() throws Exception {  
 *assertEquals*(true, addBookController.checkPrice("700000"));  
}  
  
@Test  
public void checkPriceTestCase2() throws Exception {  
 *assertEquals*(false, addBookController.checkPrice("700abc00"));  
}