Test Case

Class under test: BookController

# Generate book number:

* Input: book classification (String).
* Output: book number (String).

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case | 1 | 2 | 3 |
| Input | "Science" | "Information Technology" | "Mathematics" |
| Output | “SC0003” | “IT0001” | “MA0002” |
| Passed | Pass | Pass | Pass |

@Test  
public void generateBookNumberTestCase1() throws Exception {  
 String classification = "Science";  
 String type = classification.substring(0, 2).toUpperCase();  
 String number = new DecimalFormat("0000").format((getHighestBookNumber(classification) + 1) % 10000);  
 String bookNumber = type + number;  
 *assertEquals*(bookNumber, bookController.generateBookNumber(classification));  
}  
  
@Test  
public void generateBookNumberTestCase2() throws Exception {  
 String classification = "Information Technology";  
 String type = classification.substring(0, 2).toUpperCase();  
 String number = new DecimalFormat("0000").format((getHighestBookNumber(classification) + 1) % 10000);  
 String bookNumber = type + number;  
 *assertEquals*(bookNumber, bookController.generateBookNumber(classification));  
}  
  
@Test  
public void generateBookNumberTestCase3() throws Exception {  
 String classification = "Mathematics";  
 String type = classification.substring(0, 2).toUpperCase();  
 String number = new DecimalFormat("0000").format((getHighestBookNumber(classification) + 1) % 10000);  
 String bookNumber = type + number;  
 *assertEquals*(bookNumber, bookController.generateBookNumber(classification));  
}  
  
private int getHighestBookNumber(String classification) {  
 List<Book> bookList = DataHandler.*requestBookList*();  
 int max = 0;  
 for (Book book : bookList) {  
 if (book.getBooknumber().substring(0, 2).toLowerCase()

.equals(classification.substring(0, 2).toLowerCase())) {  
 int tmp = Integer.*parseInt*(book.getBooknumber().substring(2));  
 if (tmp > max) {  
 max = tmp;  
 }  
 }  
 }  
 return max;  
}