TESTINLJ

```
1  package query;
2
3  import database.Database;
4  import database.DatabaseException;
5  import databox.*;
6  import index.BPlusTree;
7  import index.BPlusTreeException;
8  import table.*;
9
10 import org.junit.After;
11 import org.junit.Before;
12 import org.junit.Test;
13 import org.junit.Rule;
14
15 import org.junit.rules.TemporaryFolder;
16
17 import java.io.File;
18 import java.io.IOException;
19 import java.nio.charset.Charset;
20 import java.nio.file.Files;
21 import java.nio.file.Paths;
22 import java.util.*;
23
24 import static org.junit.Assert.assertEquals;
25
26 public class TestINLJ {
27     public static final String TestDir = "testDatabase";
28     private Database db;
29     private String filename;
30     private File file;
31     private String btree_filename = "TestBPlusTree";
32
33     @Rule
34     public TemporaryFolder tempFolder = new TemporaryFolder();
35
36     @Before
37     public void beforeEach() throws Exception {
38         File testDir = tempFolder.newFolder(TestDir);
39         this.filename = testDir.getAbsolutePath();
40         this.db = new Database(filename);
41         this.db.deleteAllTables();
42         this.file = tempFolder.newFile(btree_filename);
43     }
44
45     @After
46     public void afterEach() {
47         this.db.deleteAllTables();
48         this.db.close();
49     }
50
51     private BPlusTree getBPlusTree(Type keySchema, int order) throws
       BPlusTreeException {
52         return new BPlusTree(file.getAbsolutePath(), keySchema, order);
53     }
54
55
56     @Test
57     public void testINLJ_SJoinE() throws DatabaseException, BPlusTreeException,
       IOException, QueryPlanException {
58
59
60         // create second table
61         String table1Name = "student";
62         String table2Name = "enrollment";
63
64         Database.Transaction t1 = db.beginTransaction();
65
```

```
66          BPlusTree rightBtree = loadStudent(t1);
67          loadEnrollment(t1);
68
69          // ******************** WRITE YOUR CODE BELOW ************************
70          // init INLJ Operator
71
72          // loop and print result
73          // ******************** WRITE YOUR CODE ABOVE ************************
74
75          t1.end();
76
77          throw new UnsupportedOperationException("TODO: implement");
78
79
80      }
81
82      @Test
83      public void testINLJ_SJoinEJoinC() throws DatabaseException,
        BPlusTreeException, IOException, QueryPlanException {
84
85          // create second table
86          String table1Name = "student";
87          String table2Name = "enrollment";
88
89          Database.Transaction t1 = db.beginTransaction();
90
91          BPlusTree rightBtree = loadStudent(t1);
92          loadEnrollment(t1);
93
94
95          // ******************** WRITE YOUR CODE BELOW ************************
96          // init BtreeIndexScanOperator
97
98          // init INLJ Operator
99
100         // loop and print result
101         // ******************** WRITE YOUR CODE ABOVE ************************
102
103
104         // ******************** WRITE YOUR CODE BELOW ************************
105         // use TestSourceOperator create a new DataSource that contains the join
            result
106         // ******************** WRITE YOUR CODE ABOVE ************************
107
108
109         // ******************** WRITE YOUR CODE BELOW ************************
110         // init BtreeIndexScanOperator
111
112         // init INLJ
113
114         // loop and print result
115         // ******************** WRITE YOUR CODE ABOVE ************************
116
117         t1.end();
118         throw new UnsupportedOperationException("TODO: implement");
119
120     }
121
122     private BPlusTree loadStudent(Database.Transaction t1) throws
        DatabaseException, BPlusTreeException, IOException{
123         // Create student table/Schema
124
125         // create b+ tree on id
126
127         // create table
128
129
```

```
            // read from csv file
1
2           // add each line to record and create a b+tree
3
134         throw new UnsupportedOperationException("TODO: implement");
135     }
136
137     private void loadEnrollment(Database.Transaction t1) throws  DatabaseException,
        BPlusTreeException, IOException{
138         // Create student table/Schema
139
140         // create b+ tree on id
141
142         // create table
143
144
145         // read from csv file
146
147         // add each line to record (you can create a tree here, change the return
            type)
148
149         throw new UnsupportedOperationException("TODO: implement");     }
150
151     private BPlusTree loadCourse(Database.Transaction t1) throws
        DatabaseException, BPlusTreeException, IOException{
152         // Create student table/Schema
153
154         // create b+ tree on id
155
156         // create table
157
158
159         // read from csv file
160
161         // add each line to record and create a b+tree
162
163         throw new UnsupportedOperationException("TODO: implement");     }
164 }
165
```