# 实验五

#### #一、相关知识点

- 1. JDBC基本概念
- 2. JDBC简单查询、连接查询、嵌套查询、集函数查询等

## #二、实验目的

理解Statement对象、ResultSet对象。

## #三、实验内容

1、在booklib工程的BookManager类中增加如下函数(要求采用Statement完成相关查询),并在main函数中进行测试,在实验报告中将代码补上:

```
1
     public int getBookCount(String pubid) throws BaseException{
 2
          //要求返回该出版社的图书数量
3
          int result = 0;
          Connection conn = null;
4
 5
          try {
6
             conn = DBUtil.getConnection();
7
             String sql = "select * from beanpublisher where pubid=?";
             java.sql.PreparedStatement pst = conn.prepareStatement(sql);
8
9
             pst.setString(1, pubid);
             java.sql.ResultSet rs = pst.executeQuery();
10
             if (!rs.next()) throw new BusinessException("出版社不存在");
11
             sql = "select count(1) from beanbook where pubid = ?" +
12
             "group by pubid";
13
14
             pst = conn.prepareStatement(sql);
```

```
pst.setString(1, pubid);
15
16
             rs = pst.executeQuery();
             if (rs.next()) result = rs.getInt(1);
17
           } catch (SQLException e) {
18
19
             e.printStackTrace();
             throw new DbException(e);
20
21
           } finally {
             if (conn != null)
22
23
                try {
                   conn.close();
24
25
                } catch (SQLException e) {
                  // TODO Auto-generated catch block
26
                  e.printStackTrace();
27
                }
28
29
           }
           return result;
30
31
      }
```

```
mysql> select * from Beanbook;
+-----+
| barcode | bookname | pubid | price | state |
+-----+
| barcode1 | book1 | 1 | 10 | 在库 |
| barcode2 | book2 | 1 | 10 | 在库 |
| barcode3 | book3 | 2 | 10 | 在库 |
| barcode4 | book4 | 1 | 10 | 在库 |
+-----+
4 rows in set (0.00 sec)
```

![image

```
<u>278</u>9
          public static void main(String[] args) {
279
              BookManager pm = new BookManager();
280
                   String pubid = "1";
281
                   System.out.println(pm.getBookCount(pubid));
282
283
                   pubid = "10";
284
                   System.out.println(pm.getBookCount(pubid));
                     System.out.println("有图书的出版社数量为 " + pm.getPublisherCount());
System.out.println("没有图书的出版社数量为 " + pm.getNoneBookPublisherCount());
285 //
286 //
                     System.out.println("图书的平均价格为 " + pm.getBookAvgPrice());
287 //
 288
              } catch (BaseException e) {
289
                  // TODO Auto-generated catch block
290
                   e.printStackTrace();
291
292 //
               try {
293 //
                    pm.deletePublisher("testpubid");
294 //
               } catch (BaseException e) {
295 //
                   // TODO Auto-generated catch block
296 //
                    e.printStackTrace();
297 //
              }
298
299
300 }
301
Problems @ Javadoc Q Declaration □ 控制台 ×
<已終止> BookManager [Java 应用程序] C:\Users\Bexh0lder\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.2.v20220201-1201
<u>cn.edu.zucc.booklib.util.BusinessException</u>: 出版社不存在
        at cn.edu.zucc.booklib.control.BookManager.getBookCount(BookManager.java:26)
        at cn.edu.zucc.booklib.control.BookManager.main(BookManager.java:284)
```

```
1
      public int getPublisherCount() throws BaseException{
          //要求返回图书表中出现过的出版社数量
 2
 3
          int result = 0;
4
          Connection conn = null;
 5
          try {
 6
             conn = DBUtil.getConnection();
 7
             String sql = "select count(distinct pubid) from beanbook";
             java.sql.PreparedStatement pst = conn.prepareStatement(sql);
8
9
             java.sql.ResultSet rs = pst.executeQuery();
10
             if(rs.next()) result = rs.getInt(1);
          } catch (SQLException e) {
             e.printStackTrace();
12
13
             throw new DbException(e);
          } finally {
14
```

```
15
              if (conn != null)
16
                try {
17
                   conn.close();
18
                } catch (SQLException e) {
                   // TODO Auto-generated catch block
19
20
                   e.printStackTrace();
21
22
           }
23
           return result;
24
      }
```

```
mysql> select * from Beanbook;
                                    pubid | price | state
                   bookname
   barcode
  barcode1
                   book1
                                    1
                                                    10
  barcode2
                   book2
                                    1
                                                    10
  barcode3
                   book3
                                    2
                                                    10
  barcode4
                                    1
                                                    10
                   book4
  rows in set (0.00 sec)
 ZD9 //
                 System.out.printin(pm.getBookCount(pubia));
               System.out.println("有图书的出版社数量为 " + pm.getPublisherCount());
 260
 261
            } catch (BaseException e) {
 262
                // TODO Auto-generated catch block
 263
                e.printStackTrace();
 264
 265 //
             try {
                pm.deletePublisher("testpubid");
 266 //
 267 //
             } catch (BaseException e) {
 268 //
                // TODO Auto-generated catch block
 269 //
                e.printStackTrace();
  270
🥷 Problems @ Javadoc 🖳 Declaration 📮 控制台 🗵
<已終止> BookManager [Java 应用程序] C:\Users\Bexh0lder\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.fu
有图书的出版社数量为 2
```

```
1
     public int getNoneBookPublisherCount()throws BaseException{
 2
          //要求返回没有图书的出版社数量
 3
          int result = 0;
 4
          Connection conn = null;
 5
          try {
6
             conn = DBUtil.getConnection();
 7
             String sql = "select count(*) from beanpublisher";
8
             java.sql.PreparedStatement pst = conn.prepareStatement(sql);
9
             java.sql.ResultSet rs = pst.executeQuery();
10
             if(rs.next()) result = rs.getInt(1);
             result -= (new BookManager()).getPublisherCount();
11
```

```
12
           } catch (SQLException e) {
13
              e.printStackTrace();
              throw new DbException(e);
14
15
           } finally {
16
              if (conn != null)
17
                try {
18
                   conn.close();
19
                } catch (SQLException e) {
                  // TODO Auto-generated catch block
20
21
                  e.printStackTrace();
22
                }
23
           }
24
           return result;
25
```

```
mysql> select * from Beanbook;
             bookname | pubid | price | state
             book1
  barcode1
                                    10
  barcode2
             book2
                         1
                                    10
  barcode3
             book3
                         2
  barcode4
           book4
                       | 1
                                    10
4 rows in set (0.00 sec)
```

```
mysql> select * from Beanpublisher;
 pubid | publisherName |
        11
                          111
 2
        22
                          222
 3
         33
                          333
         12
 4
                          112
 5
          23
                          223
        34
                          334
6 rows in set (0.00 sec)
mysql> select count(*) from Beanpublisher;
 count(*)
 row in set (0.01 sec)
```

```
284 //
                System.out.println(pm.getBookCount(pubid));
System.out.println("有图书的出版社数量为 " + pm.getPublisherCount());
System.out.println("没有图书的出版社数量为 " + pm.getNoneBookPublisherCount());
285
286
                  System.out.println("有图书的出版社数量为 " + pm.getNoneBookPublisherCount());
287 //
288
            } catch (BaseException e) {
289
                // TODO Auto-generated catch block
290
                e.printStackTrace();
291
            }
292 //
293 //
                 pm.deletePublisher("testpubid");
 294 //
             } catch (BaseException e) {
295 //
                // TODO Auto-generated catch block
296 //
                 e.printStackTrace();
297 //
298
299
300 }
🧗 Problems @ Javadoc 🚇 Declaration 📮 控制台 🗵
<已終止> BookManager [Java 应用程序] C:\Users\Bexh0lder\,p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.2.v.
有图书的出版社数量为 2
没有图书的出版社数量为 4
  1
        public double getBookAvgPrice()throws BaseException{
  2
              //要求返回图书的评价价格
  3
              double result = 0;
  4
              Connection conn = null;
  5
              try {
  6
                conn = DBUtil.getConnection();
  7
                 String sql = "select avg(price) from Beanbook";
  8
                java.sql.PreparedStatement pst = conn.prepareStatement(sql);
  9
                java.sql.ResultSet rs = pst.executeQuery();
 10
                if(rs.next()) result = rs.getDouble(1);
 11
              } catch (SQLException e) {
 12
                 e.printStackTrace();
 13
                throw new DbException(e);
 14
              } finally {
 15
                if (conn != null)
 16
                   try {
 17
                       conn.close();
 18
                   } catch (SQLException e) {
                      // TODO Auto-generated catch block
 19
```

20

21

22

23

24

}

e.printStackTrace();

}

return result;

}

```
select * from Beanbook;
                        bookname | pubid | price | state
   barcode
   barcode1
                        book1
                                                                  10
   barcode2
                        book2
   barcode3
                        book3
   barcode4
                                                                  10
                        book4
  rows in set (0.00 sec)
                 System.out.println("月图书的出版社数重为 " + pm.getPublisherCount());
System.out.println("没有图书的出版社数量为 " + pm.getNoneBookPublisherCount());
286 //
287
               System.out.println("图书的平均价格为 " + pm.getBookAvgPrice());
288
           } catch (BaseException e) {
 289
               // TODO Auto-generated catch block
290
               e.printStackTrace();
 291
292 //
                pm.deletePublisher("testpubid");
293 //
294 //
            } catch (BaseException e) {
295 //
               // TODO Auto-generated catch block
                e.printStackTrace();
296 //
 297 //
298
299
 300 }
 301
Problems @ Javadoc Q Declaration □ 控制台 ×
<已終止> BookManager [Java 应用程序] C:\Users\Bexh0lder\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.2.v202
图书的平均价格为 10.0
```

2、在booklib工程的BookLendManager类中增加如下函数,并在main函数中进行测试,在实验报告中将代码补上:

```
1
     public String loadBookLendOperator(String barcode)throws BaseException{
         //参数为图书条码,返回这本图书最近一次被借出时的操作员姓名,要求采用连
 2
     接查询实现。难点:如何识别出最近一次?假设不允许用mysql的limit关键字,也不能
     用嵌套查询,应该如何完成?
3
         BeanBook book = (new BookManager()).loadBook(barcode);
         if (book == null) throw new BusinessException("图书不存在");
4
5
         String result = "";
         Connection conn = null;
6
7
         try {
8
           conn = DBUtil.getConnection();
           String sql = "select * from BeanBookLendRecord where bookBarcode=?";
9
           java.sql.PreparedStatement pst = conn.prepareStatement(sql);
10
11
           pst.setString(1, barcode);
12
           java.sql.ResultSet rs = pst.executeQuery();
13
           if (!rs.next()) {
14
             throw new BusinessException("该图书没有借阅记录");
15
           }
```

```
16
              sql = "select username,max(lendDate)" +
                   " from beanbooklendrecord" +
17
                   "left outer join beansystemuser on lendOperUserid = userid" +
18
19
                   " where bookBarcode=?";
20
             sql += " group by bookBarcode";
21
             pst = conn.prepareStatement(sql);
22
             pst.setString(1, barcode);
23
             rs = pst.executeQuery();
24
             if(rs.next()) result = rs.getString(1);
           } catch (SQLException e) {
25
26
              e.printStackTrace();
27
             throw new DbException(e);
28
           } finally {
29
             if (conn != null)
30
                try {
31
                   conn.close();
32
                } catch (SQLException e) {
                   // TODO Auto-generated catch block
33
                   e.printStackTrace();
34
                }
35
36
           }
37
           return result;
38
```

```
mysql> select * from beanbooklendrecord;

| id | readerid | bookBarcode | LendDate | returnDate | lendOperUserid | returnOperUserid | penalSum |
| 6 | 32001272 | barcode1 | 2022-05-09 17:36:52 | 2022-05-09 17:37:04 | admin | admin | 0 |
| 7 | 32001003 | barcode1 | 2022-05-09 17:37:19 | 2022-05-09 17:37:44 | admin | admin | 0 |
| 8 | 32001002 | barcode1 | 2022-05-09 17:38:04 | 2022-05-09 17:38:14 | admin | admin | 0 |
| 9 | 32001272 | barcode2 | 2022-05-09 17:41:20 | 2022-05-09 17:41:41 | admin | admin | 0 |
| 10 | 32001002 | barcode3 | 2022-05-09 17:41:32 | 2022-05-09 17:41:47 | admin | admin | 0 |
| 5 rows in set (0.00 sec)
```

```
352⊝
          public static void main(String[] args) {
 353
              BookLendManager pm = new BookLendManager();
 354
355
                   System.out.println("最近一次被借出时的操作员姓名为 " + pm.loadBookLendOperator("barcode1"));
System.out.println("最近一次被借出时的操作员姓名为 " + pm.loadBookLendOperator("barcode4"));
356
 357 //
                     System.out.println("最近一次被借出时的操作员姓名为 " + pm.loadBookLendOperator("barcode5"));
358 //
 359
              } catch (BaseException e) {
360
                   // TODO Auto-generated catch block
361
                   e.printStackTrace();
 362
363 //
                try {
 364 //
                     pm.deletePublisher("testpubid");
365 //
                } catch (BaseException e) {
                    // TODO Auto-generated catch block
366 //
                    e.printStackTrace();
 367 //
                }
368 //
 369
 370
371 }
372
🖳 Problems @ Javadoc 🚇 Declaration 🕒 控制台 🗵
<已終止>BookLendManager [Java 应用程序] C:\Users\Bexh0lder\,p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86 64 17.0.2.v20220201-1
最近一次被借出时的操作员姓名为 管理员
```

#### 3、完成题2中的功能,要求采用嵌套查询实现。

```
1
        public String loadBookLendOperator(String barcode)throws BaseException{
 2
          //参数为图书条码,返回这本图书最近一次被借出时的操作员姓名,要求采用连
     接查询实现。难点:如何识别出最近一次?假设不允许用mysql的limit关键字,也不能
     用嵌套查询,应该如何完成?
 3
          BeanBook book = (new BookManager()).loadBook(barcode);
          if (book == null) throw new BusinessException("图书不存在");
 4
          String result = "";
 5
          Connection conn = null;
 6
 7
          try {
 8
            conn = DBUtil.getConnection();
 9
            String sql = "select * from BeanBookLendRecord where bookBarcode=?";
            java.sql.PreparedStatement pst = conn.prepareStatement(sql);
10
            pst.setString(1, barcode);
11
            java.sql.ResultSet rs = pst.executeQuery();
12
13
            if (!rs.next()) {
               throw new BusinessException("该图书没有借阅记录");
14
15
            }
16
            //连接查询
17
     //
             sql = "select username,max(lendDate)" +
                  "from beanbooklendrecord" +
18
     //
                  "left outer join beansystemuser on lendOperUserid = userid" +
19
     //
     //
                  "where bookBarcode=?";
20
             sql += " group by bookBarcode";
21
            //嵌套查询
22
            sql = "select username from beansystemuser where userid in (" +
23
24
                 "select lendOperUserid from beanbooklendrecord where lendDate in ("
                 "select max(lendDate) from beanbooklendrecord where bookBarcode =
25
     ? group by bookBarcode))";
            pst = conn.prepareStatement(sql);
26
27
            pst.setString(1, barcode);
28
            rs = pst.executeQuery();
            if(rs.next()) result = rs.getString(1);
29
          } catch (SQLException e) {
30
31
            e.printStackTrace();
            throw new DbException(e);
32
          } finally {
33
            if (conn != null)
34
35
              try {
36
                 conn.close();
```

```
220
 357€
            public static void main(String[] args) {
 358
                  BookLendManager pm = new BookLendManager();
 359
 360

      System.out.println("最近一次被借出时的操作员姓名为" + pm.loadBookLendOperator("barcode1"));

      System.out.println("最近一次被借出时的操作员姓名为" + pm.loadBookLendOperator("barcode4"));

      System.out.println("最近一次被借出时的操作员姓名为" + pm.loadBookLendOperator("barcode5"));

 361
 362 //
 363 //
 364
                  } catch (BaseException e) {
@365
                       // TODO Auto-generated catch block
                        e.printStackTrace();
 366
 367
 368 //
                    try {
                         pm.deletePublisher("testpubid");
 369 //
 370 //
                   } catch (BaseException e) {
2371 //
                        // TODO Auto-generated catch block
372 /,
373 //
}
                         e.printStackTrace();
 375
 376 }
 377

  Problems @ Javadoc ❷ Declaration ■ 控制台 ×

<已終止> BookLendManager [Java 应用程序] C:\Users\Bexh0lder\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.2.v20220201-120
```

4、在booklib工程的BookLendManager类中增加如下函数,并在main函数中进行测试,在实验报告中将代码补上:

最近一次被借出时的操作员姓名为 管理员

```
1
     public void showAllLendRecord(){
         //通过System.out.println方法,输出所有借阅记录的明细数据,要求结果中包括
2
     读者姓名、图书名称、所属出版社名称、借阅操作员姓名、归还操作员姓名、借阅时
     间、归还时间等
3
         //注意:需要注意未归还图书的情况
4
         Connection conn = null;
5
         try {
6
           List<BeanBookLendRecord> result = new ArrayList<BeanBookLendRecord>
     ();
7
           conn = DBUtil.getConnection();
           String sql = "select * from BeanBookLendRecord";
8
9
           java.sql.PreparedStatement pst = conn.prepareStatement(sql);
           java.sql.ResultSet rs = pst.executeQuery();
10
11
           while(rs.next()) {
12
             String readerName = "";
13
             String bookName = "";
```

```
14
                String publisherName = "";
15
                String lendOperUserName = null;
                String returnOperUserName = null;
16
17
                BeanBookLendRecord r = new BeanBookLendRecord();
18
                r.setId(rs.getInt(1));
                r.setReaderid(rs.getString(2));
19
20
                r.setBookBarcode(rs.getString(3));
21
                r.setLendDate(rs.getTimestamp(4));
22
                r.setReturnDate(rs.getTimestamp(5));
                r.setLendOperUserid(rs.getString(6));
23
                r.setReturnOperUserid(rs.getString(7));
24
25
                r.setPenalSum(rs.getDouble(8));
26
                result.add(r);
27
                sql = "select readerName from beanreader where readerid = ?";
                pst = conn.prepareStatement(sql);
28
29
                pst.setString(1, r.getReaderid());
30
                java.sql.ResultSet xs = pst.executeQuery();
31
                if(xs.next()) readerName = xs.getNString(1);
                sql = "select bookName, publisherName from beanbook,beanpublisher
32
      where barcode = ? and beanpublisher.pubid = beanbook.pubid";
33
                pst = conn.prepareStatement(sql);
                pst.setString(1, r.getBookBarcode());
34
                xs = pst.executeQuery();
35
36
                if(xs.next()){
37
                  readerName = xs.getNString(1);
38
                  publisherName = xs.getNString(2);
39
                }
                if(r.getLendOperUserid() != null)
40
41
                  sql = "select username from beansystemuser where userid = ?";
42
43
                  pst = conn.prepareStatement(sql);
                  pst.setString(1, r.getLendOperUserid());
44
                  xs = pst.executeQuery();
45
46
                  if(xs.next()) lendOperUserName = xs.getNString(1);
47
                }
48
                if(r.getReturnOperUserid() != null)
49
                  sql = "select username from beansystemuser where userid = ?";
50
51
                  pst = conn.prepareStatement(sql);
52
                  pst.setString(1, r.getReturnOperUserid());
                  xs = pst.executeQuery();
53
                  if(xs.next()) returnOperUserName = xs.getNString(1);
54
```

```
55
               System.out.println("读者姓名:" + readerName + "\t图书名称:" +
56
                    bookName + "\t所属出版社名称:" + publisherName + "\t借阅操作员姓
57
      名:" + lendOperUserName
58
                    + "\t归还操作员姓名:" + returnOperUserName + "\t借阅时间:" +
                   r.getLendDate() + "\t归还时间:" + r.getReturnDate());
59
60
61
          } catch (SQLException e) {
62
             e.printStackTrace();
              throw new DbException(e);
63
     //
64
          } finally {
65
            if (conn != null)
66
               try {
67
                 conn.close();
               } catch (SQLException e) {
68
                 // TODO Auto-generated catch block
69
70
                 e.printStackTrace();
71
               }
72
          }
73
```

```
public static void main(String[] args) {
 432
                 BookLendManager pm = new BookLendManager();
                   try {
 433 //
 434
435 //
                       pm.showAllLendRecord();
                   pm.$howAllLendRecOrd();
System.out.println(電通一人按備出時的操作员姓名为 " + pm.loadBookLendOperator("barcode1"));
System.out.println("最近一次接備出時的操作员姓名为 " + pm.loadBookLendOperator("barcode4"));
System.out.println("最近一次接借出時的操作员姓名为 " + pm.loadBookLendOperator("barcode5"));
Catch (BaseException e) {
// TODO Auto-generated catch block
436 //
437 //
438 //
2439 //
440 //
441 //
442 //
443 //
444 //
                          e.printStackTrace();
                  pm.deletePublisher("testpubid");
} catch (BaseException e) {
   // TODO Auto-generated catch block
 446 //
447 //
448
                         e.printStackTrace():
 449
 450 }
451

▼ Problems @ Javadoc @ Declaration ■ 控制台 ×
                                                     se (2022年5月9日下午9:58:38-下午9:58:4
                      图书名称: 所属出版社名称:11 图书名称: 所属出版社名称:11
读者姓名:book1
读者姓名:book2
                      图书名称:
                                所属出版社名称:11
读者姓名:book3
                      图书名称:
                                所属出版社名称:22
                                                      借阅操作员姓名:管理员 归还操作员姓名:null 借阅时间:2022-05-09 20:51:38.0
```

5、在booklib工程的BookManager类中增加如下函数,并在main函数中进行测试,在实验报告中将代码补上:

```
public void showTop5Books(){

//通过System.out.println方法,输出借阅次数最多的5本图书及其借阅次数

Connection conn = null;

try {

conn = DBUtil.getConnection();

String sql = "select bookname, count(1) as x "
```

```
7
                  + "from beanbooklendrecord, beanbook "
 8
                  + "where barcode = bookBarcode "
                  + "group by bookname order by x DESC limit 5;";
 9
10
             java.sql.PreparedStatement pst = conn.prepareStatement(sql);
11
             java.sql.ResultSet rs = pst.executeQuery();
             while(rs.next())
12
13
14
                System.out.println("图书名称" + rs.getString(1) + "\t借阅次数" +
      rs.getInt(2));
15
             }
           } catch (SQLException e) {
16
17
             e.printStackTrace();
      //
              throw new DbException(e);
18
           } finally {
19
             if (conn != null)
20
21
                try {
22
                  conn.close();
23
                } catch (SQLException e) {
                  // TODO Auto-generated catch block
24
25
                  e.printStackTrace();
26
                }
27
           }
28
     }
```

```
public void showTop5Publisher(){
 1
          //通过System.out.println方法,输出被借阅图书次数最多的5个出版名称及其总借
 2
     阅次数和被借阅过的图书次数
          Connection conn = null;
 3
4
          try {
 5
            conn = DBUtil.getConnection();
6
            String sql = "select publishername,count(*) as x, count(distinct bookbarcode)
 7
                 + "from beanbooklendrecord "
8
                 + "left outer join (beanbook left outer join beanpublisher on
     beanbook.pubid = beanpublisher.pubid) on bookbarcode = barcode "
9
                 + "group by beanbook.pubid order by x desc limit 5;";
            java.sql.PreparedStatement pst = conn.prepareStatement(sql);
10
            java.sql.ResultSet rs = pst.executeQuery();
11
12
            while(rs.next())
13
14
              System.out.println("出版社名称" + rs.getString(1) + "\t总借阅次数" +
     rs.getInt(2) + "\t被借阅过的图书数量" + rs.getInt(3));
```

```
15
           } catch (SQLException e) {
16
              e.printStackTrace();
17
18
      //
               throw new DbException(e);
19
           } finally {
              if (conn != null)
20
21
                try {
22
                   conn.close();
23
                } catch (SQLException e) {
24
                   // TODO Auto-generated catch block
25
                   e.printStackTrace();
26
                }
27
           }
28
```

```
334⊜
         public static void main(String[] args) {
 335
             BookManager pm = new BookManager();
 336 //
               try {
                  pm.showTop5Publisher();
 337
 338 //
                  String pubid = "1"
 339 //
                   System.out.println(pm.getBookCount(pubid));
                    pubid = "10";
 340 //
 341 //
                   System.out.println(pm.getBookCount(pubid));
                   System.out.println("有图书的出版社数量为 " + pm.getPublisherCount());
System.out.println("没有图书的出版社数量为 " + pm.getNoneBookPublisherCount());
 342 //
 343 //
                   System.out.println("图书的平均价格为 " + pm.getBookAvgPrice());
 344 //
 345 //
              } catch (BaseException e) {
                 // TODO Auto-generated catch block
@346
 347 //
                    e.printStackTrace();
 348 //
 349 //
               try {
 350 //
                   pm.deletePublisher("testpubid");
 351 //
              } catch (BaseException e) {
2352 //
                 // TODO Auto-generated catch block
 353 //
                  e.printStackTrace();
 354 //
 355
 356
 357 }
 358

♀ Problems @ Javadoc ❷ Declaration ❷ 控制台 ×
<已終止>BookManager [Java 应用程序] C:\Users\Bexh0lder\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.2.v20220201-
            总借阅次数8
出版社名称11
                                 被借阅过的图书数量3
出版社名称22
                总借阅次数2
                                  被借阅过的图书数量1
出版社名称33
                                  被借阅过的图书数量1
               总借阅次数2
出版社名称23
                总借阅次数2
                                  被借阅过的图书数量1
出版社名称34
                总借阅次数2
                                  被借阅过的图书数量1
```

6、在BookLendManager中增加函数public void printDateLendRecord(String date)throws DbException,并在main函数中调用测试;要求通过该函数输出指定日期的所有借阅记录,,输出格式如下:

```
readerId=*,bookBarcode=**,lendDate=2020-05-01 15:17:01,returnDate=未归还 readerId=*,bookBarcode=**,lendDate=2020-05-01 15:17:01,returnDate=2020-05-12 12:00:00
```

说明:每个借阅记录1行输出,如果returnDate为空,则输出:"未归还"注:时间的输出格式请使用java.text.SimpleDateFormat类实现请提供函数代码及运行结果截图:

```
public void printDateLendRecord(String date)throws DbException{
 1
 2
           Connection conn = null;
 3
           SimpleDateFormat DateFor = new SimpleDateFormat("yyyy-MM-dd
      hh:mm:ss");
 4
           try {
 5
             conn = DBUtil.getConnection();
 6
             String sql = "select readerid, bookBarcode, lendDate, returnDate from
      beanbooklendrecord where lendDate = ?";
 7
             java.sql.PreparedStatement pst = conn.prepareStatement(sql);
 8
             pst.setString(1, date);
             java.sql.ResultSet rs = pst.executeQuery();
 9
10
             if(rs.next())
                if(rs.getDate(4) != null)
11
                  System.out.println("readerId= " + rs.getString(1) + ",bookBarcode= " +
12
      rs.getString(2) + ",lendDate= " + DateFor.format(rs.getDate(3)) + ",returnDate=" +
      DateFor.format(rs.getDate(4)));
13
                else
                  System.out.println("readerId= " + rs.getString(1) + ",bookBarcode= " +
14
      rs.getString(2) + ",lendDate= " + DateFor.format(rs.getDate(3)) + ",未归还");
15
           } catch (SQLException e) {
16
             e.printStackTrace();
             throw new DbException(e);
17
18
           } finally {
             if (conn != null)
19
20
                try {
21
                  conn.close();
22
                } catch (SQLException e) {
23
                  // TODO Auto-generated catch block
                  e.printStackTrace();
24
25
               }
           }
26
27
     }
```

```
public static void main(String[] args) {
    BookLendManager pm = new BookLendManager();
 459⊜
  460
  461
                           try {
                                   1
pm.printDateLendRecord("2022-05-09 17:36:52");
pm.printDateLendRecord("2022-05-09 22:55:23");
System.out.println("最近一次被借出时的操作员姓名为 " + pm.loadBookLendOperator("barcode1"));
System.out.println("最近一次被借出时的操作员姓名为 " + pm.loadBookLendOperator("barcode4"));
System.out.println("最近一次被借出时的操作员姓名为 " + pm.loadBookLendOperator("barcode5"));
  462
  463
  464 //
  465 //
466 //
  467
                           } catch (BaseException e) {
    // TODO Auto-generated
                                              TODO Auto-generated catch block
  468 //
  469
                                    e.printStackTrace();
469
470
471 //
472 //
473 //
2474 //
475 //
                             try {
                                      pm.deletePublisher("testpubid");
                             } catch (BaseException e) {
   // TODO Auto-generated catch block
   e.printStackTrace();
  476 //
477
  478
 479 }
  480
<已終止>BookLendManager [Java 应用程序] C.\Users\BexhOlder\,p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.2.v20220201-1208\jre\bin\javaw.readerId= 32001272,bookBarcode= barcode1,lendDate= 2022-05-09_12:00:00,returnDate=2022-05-09_12:00:00
readerId= 32001272,bookBarcode= barcode1,lendDate= 2022-05-09_12:00:00,未归还
```