

课程回顾

1 javaWeb实现CRUD实现，步骤具体有

- 1 1. 分析实现需求，底层使用SQL语句
- 2 新增: `insert 表 values(?,?,?,?)`
- 3 2. 表--->domain包实体类
- 4 2-1 属性名和查询SQL语句查询结果集列名不一致
- 5 查询SQL: `select book_code code`
- 6 实体类: `book_Code`
- 7
- 8 提供有参构造方法:
- 9 没有手动添加无参构造方法
- 10 3. dao层: JDBC代码
- 11 接口+实现类
- 12 4. 建议: 单元测试, 确认dao没有问题
- 13 5. service层:
- 14 接口+实现类
- 15 6. 建议: 单元测试, 确认service没有问题
- 16
- 17 7. 设计出来, JSP
- 18 JSP需要将数据提交Servlet
- 19
- 20 8. Servlet处理请求:
- 21 取
- 22 调
- 23 `response`写响应体
- 24 存
- 25 转

课程目标

1 新增 ==== 掌握

扩展: 简化数据提交时, 在Servlet获取数据代码

BeanUtils工具

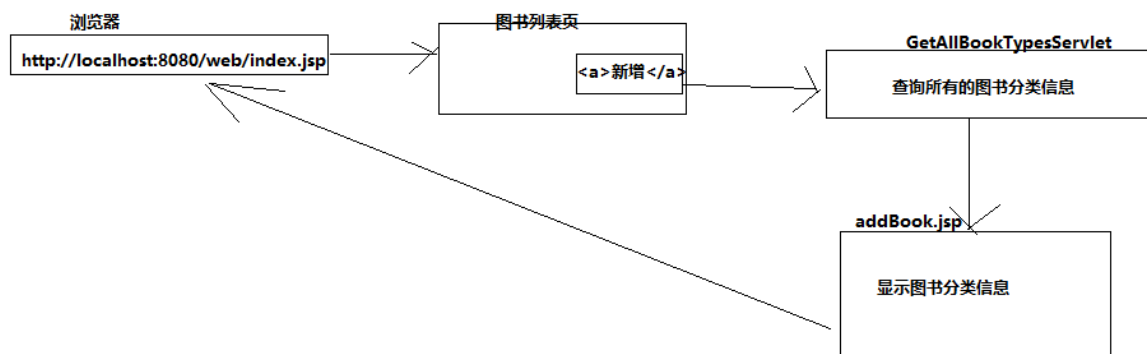
2 修改 ==== 难点

3 模糊查询+分页

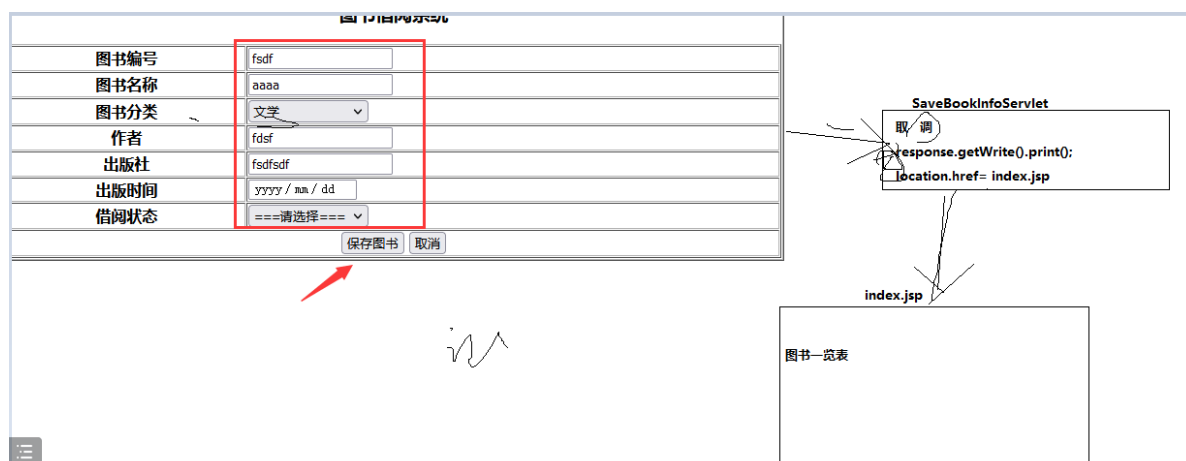
课程实施

1 新增

1-1 新增业务流程



1-2 进入新增jsp页面的代码



daoの图书分类查询

```
1 package cn.kgc.dao.impl;
2
3 import cn.kgc.dao.BookTypeDao;
4 import cn.kgc.domain.BookType;
5 import cn.kgc.util.JDBCUtil;
6 import org.apache.commons.dbutils.QueryRunner;
7 import org.apache.commons.dbutils.handlers.BeanListHandler;
8
9 import java.sql.SQLException;
10 import java.util.List;
11
12 /**
13  * @Author: lc
14  * @Date: 2022/5/20
15  * @Description: cn.kgc.dao.impl
16  * @Version: 1.0
17  */
18 public class BookTypeDaoImpl implements BookTypeDao {
19     private QueryRunner qr=new QueryRunner(JDBCUtil.datasource);
20     @Override
21     public List<BookType> selectAll() {
22         try {
23             return qr.query("select id,type_name from book_type",
24                             new BeanListHandler<>(BookType.class));
25         } catch (SQLException e) {
26             throw new RuntimeException(e);
27         }
28     }
29 }
```

```

27     }
28 }
29 }

```

serviceの获取所有的图书分类

```

1  package cn.kgc.service.impl;
2
3  import cn.kgc.dao.BookTypeDao;
4  import cn.kgc.dao.impl.BookTypeDaoImpl;
5  import cn.kgc.domain.BookType;
6  import cn.kgc.service.BookTypeService;
7
8  import java.util.List;
9
10 /**
11  * @Author: lc
12  * @Date: 2022/5/20
13  * @Description: cn.kgc.service.impl
14  * @Version: 1.0
15  */
16 public class BookTypeServiceImpl implements BookTypeService {
17     private BookTypeDao dao=new BookTypeDaoImpl();
18     @Override
19     public List<BookType> getAllBookTypes() {
20         return dao.selectAll();
21     }
22 }

```

Servletの获取所有的图书分类转发给新增jsp显示

```

1  package cn.kgc.controller; /**
2  * @Author: lc
3  * @Date: 2022/5/20
4  * @Description: ${PACKAGE_NAME}
5  * @Version: 1.0
6  */
7
8  import cn.kgc.domain.BookType;
9  import cn.kgc.service.impl.BookTypeServiceImpl;
10
11 import javax.servlet.*;
12 import javax.servlet.http.*;
13 import javax.servlet.annotation.*;
14 import java.io.IOException;
15 import java.util.List;
16
17 @WebServlet("/GetBookTypesServlet")
18 public class GetBookTypesServlet extends HttpServlet {
19     @Override
20     protected void doGet(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
21         //取 略
22
23         //调 查询所有的图书分类的信息
24         //实现jsp上面，外键约束的列值来源于主表的主键数据

```



```

39         <c:forEach var="type" items="${bookTypes}">
40             <option
value="${type.id}">${type.type_name}</option>
41         </c:forEach>
42     </select>
43 </td>
44 </tr>
45 <tr>
46     <th>作者</th>
47     <td>
48         <input type="text" name="book_author"/>
49     </td>
50 </tr>
51 <tr>
52     <th>出版社</th>
53     <td>
54         <input type="text" name="publish_press"/>
55     </td>
56 </tr>
57 <tr>
58     <th>出版时间</th>
59     <td>
60         <!--
61         date是表单项，提供日期选择的标签
62         --%>
63         <input type="date" name="publish_date"/>
64     </td>
65 </tr>
66 <tr>
67     <th>借阅状态</th>
68     <td>
69         <select name="is_borrow">
70             <option value="">===请选择===</option>
71             <option value="false">未借阅</option>
72             <option value="true">已借阅</option>
73         </select>
74     </td>
75 </tr>
76 <tr>
77     <td colspan="2" align="center">
78         <input type="submit" value="保存图书" />
79         <input type="reset" value="取消" />
80     </td>
81 </tr>
82 </table>
83 </td>
84 </tr>
85 </table>
86 </form>
87 </body>
88 </html>

```

1-3 实现新增功能的代码

dao的执行insert语句

```
1  @Override
2      public int insert(BookInfo book) {
3          StringBuilder sb=new StringBuilder();
4          sb.append("insert book_info values ");
5          sb.append(" (NULL,?,?,?,?,?,?)");//主键: null默认填入自增长 手动填入主
      键
6          try {
7              return qr.update(sb.toString(),
8                  book.getBook_code(),
9                  book.getBook_name(),
10                 book.getBook_type(),
11                 book.getBook_author(),
12                 book.getPublish_press(),
13                 book.getPublish_date(),
14                 //isIs_borrow getter针对boolean列自动生成的方法命名规范
15                 book.isIs_borrow());
16         } catch (SQLException e) {
17             throw new RuntimeException(e);
18         }
19     }
```

service的实现新增功能

```
1  @Override
2      public boolean add(BookInfo book) {
3          return dao.insert(book)==1;
4      }
```

Servletの获取表单数据，执行insert功能

```
1  package cn.kgc.controller; /**
2      * @Author: lc
3      * @Date: 2022/5/20
4      * @Description: ${PACKAGE_NAME}
5      * @Version: 1.0
6      */
7
8  import cn.kgc.domain.BookInfo;
9  import cn.kgc.service.impl.BookInfoServiceImpl;
10
11 import javax.servlet.ServletException;
12 import javax.servlet.annotation.WebServlet;
13 import javax.servlet.http.HttpServlet;
14 import javax.servlet.http.HttpServletRequest;
15 import javax.servlet.http.HttpServletResponse;
16 import java.io.IOException;
17 import java.io.PrintWriter;
18 import java.text.SimpleDateFormat;
19 import java.util.Date;
20
21 @WebServlet("/AddBookInfoServlet")
```

```

22 public class AddBookInfoServlet extends HttpServlet {
23     @Override
24     protected void doGet(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
25         request.setCharacterEncoding("utf-8");
26         response.setContentType("text/html;charset=utf-8");
27         //取
28         String book_code = request.getParameter("book_code");
29         String book_name = request.getParameter("book_name");
30         String book_typeStr = request.getParameter("book_type");
31         String book_author = request.getParameter("book_author");
32         String publish_press = request.getParameter("publish_press");
33         String publish_dateStr = request.getParameter("publish_date");
34         String is_borrowStr = request.getParameter("is_borrow");
35
36         //String-->其他类型
37         Integer book_type = null;
38         try {
39             book_type = Integer.valueOf(book_typeStr);
40             //Integer.parseInt(book_typeStr);
41         } catch (Exception e) {
42             //空着
43         }
44
45         Date publish_date = null;
46         try {
47             SimpleDateFormat sdf=new SimpleDateFormat("yyyy/MM/dd");
48             publish_date = sdf.parse(publish_dateStr);
49         } catch (Exception e) {
50
51         }
52         Boolean is_borrow=null;
53         try {
54             is_borrow = Boolean.valueOf(is_borrowStr);
55         } catch (Exception e) {
56
57         }
58         // 调
59         BookInfo book=new BookInfo();
60
61         book.setBook_code(book_code);
62         book.setBook_name(book_name);
63         book.setPublish_press(publish_press);
64         book.setBook_author(book_author);
65         book.setBook_type(book_type);
66         book.setPublish_date(publish_date);
67         book.setIs_borrow(is_borrow);
68
69         boolean bool = new BookInfoServiceImpl().add(book);
70
71         //响应体的方式, alert()
72         PrintWriter writer = response.getWriter();
73         writer.print("<script type='text/javascript'>alert('"+(bool?"添加成
功":"添加失败")+"'");</script>");
74         if(bool){
75             //重定向 302
76             //getServletContext().getContextPath()获取服务器上发布项目的名称

```

```

77 //response.sendRedirect(getServletContext().getContextPath()+"/index.jsp");
78     writer.print("<script
type='text/javascript'>location.href='"+getServletContext().getContextPath()
+ "/index.jsp';</script>");
79     }
80 }
81
82 @Override
83 protected void doPost(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
84     doGet(request, response);
85 }
86 }

```

扩展：表单数据的快速封装

日期转换器

```

1 //针对日期格式，日期格式转换器
2 DateConverter converter=new DateConverter();
3 converter.setPatterns(new String[]{"yyyy-MM-dd","yyyyMMdd","yyyy年MM月dd
日","yyyy/MM/dd"});
4 ConvertUtils.register(converter,java.util.Date.class);
5
6 BeanUtils.populate(book,request.getParameterMap());

```

实现form数据快速封装为Bean对象

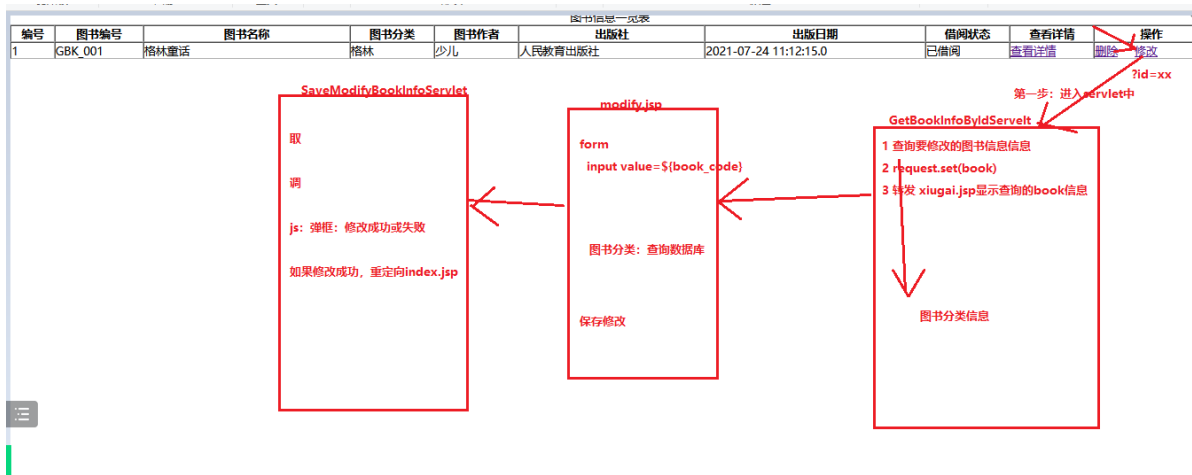
```

1 BookInfo book=new BookInfo();
2 //亲力亲为赋值，借助BeanUtils对象，简化亲力亲为赋值的过程
3 try {
4     //针对日期格式，日期格式转换器
5     DateConverter converter=new DateConverter();
6     converter.setPatterns(new String[]{"yyyy-MM-dd","yyyyMMdd","yyyy年MM月dd
日","yyyy/MM/dd"});
7     ConvertUtils.register(converter,java.util.Date.class);
8
9     BeanUtils.populate(book,request.getParameterMap());
10    System.out.println(book);
11 } catch (Exception e) {
12     throw new RuntimeException(e);
13 }

```

2 修改

2-1 修改实现流程



2-2 实现修改图书的数据显示

daoの根据主键查询要修改的图书信息

```

1  @Override
2  public BookInfo selectById(Integer id) {
3      StringBuilder sb=new StringBuilder();
4      sb.append("SELECT book_info.book_id id,book_info.book_code,");
5      sb.append("book_info.book_name,book_info.book_type,");
6      sb.append("book_info.book_author,book_info.publish_press,");
7      sb.append("book_info.publish_date,book_info.is_borrow ");
8      sb.append(" FROM book_info ");
9      sb.append(" where book_info.book_id=?");
10     try {
11         return qr.query(sb.toString(),new BeanHandler<>
12         (BookInfo.class),id);
13     } catch (SQLException e) {
14         throw new RuntimeException(e);
15     }
16 }
  
```

serviceの获取修改的图书信息

```

1  package cn.kgc.service.impl;
2
3  import cn.kgc.dao.BookInfoDao;
4  import cn.kgc.dao.impl.BookInfoDaoImpl;
5  import cn.kgc.domain.BookInfo;
6  import cn.kgc.service.BookInfoService;
7
8  import java.util.List;
9
10 /**
11  * @Author: lc
12  * @Date: 2022/5/17
13  * @Description: cn.kgc.service.impl
14  * @Version: 1.0
15  */
16 public class BookInfoServiceImpl implements BookInfoService {
17     private BookInfoDao dao=new BookInfoDaoImpl();
18
19     @Override
  
```

```

20         public BookInfo findBookInfoBy(Integer id) {
21             return dao.selectByPkId(id);
22         }
23     }
24

```

Servletの获取图书分类、要修改的图书信息存入域对象，转发给修改jsp显示

```

1  package cn.kgc.controller; /**
2      * @Author: lc
3      * @Date: 2022/5/20
4      * @Description: ${PACKAGE_NAME}
5      * @Version: 1.0
6      */
7
8  import cn.kgc.domain.BookInfo;
9  import cn.kgc.domain.BookType;
10 import cn.kgc.service.impl.BookInfoServiceImpl;
11 import cn.kgc.service.impl.BookTypeServiceImpl;
12
13 import javax.servlet.*;
14 import javax.servlet.http.*;
15 import javax.servlet.annotation.*;
16 import java.io.IOException;
17 import java.util.List;
18
19 @WebServlet("/GetModifyBookInfoServlet")
20 public class GetModifyBookInfoServlet extends HttpServlet {
21     @Override
22     protected void doGet(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
23         //取
24         String idStr = request.getParameter("id");
25         //套路:
26         Integer id=null;
27         try {
28             id = Integer.valueOf(idStr);
29         } catch (Exception e) {
30         }
31         //调用
32         //1-1 获取要显示图书信息
33         BookInfo book = new BookInfoServiceImpl().findBookInfoBy(id);
34         //1-2 获取图书分类信息
35         List<BookType> allBookTypes = new
BookTypeServiceImpl().getAllBookTypes();
36
37         //存
38         request.setAttribute("book",book);
39         request.setAttribute("types",allBookTypes);
40
41         //转发
42
43         request.getRequestDispatcher("/modify.jsp").forward(request,response);
44     }
45     @Override

```

modify.jspの显示要修改的图书信息

[illegible]

```

45         </c:forEach>
46     </select>
47 </td>
48 </tr>
49 <tr>
50     <th>作者</th>
51     <td>
52         <input type="text" name="book_author"
value="${book.book_author}"/>
53     </td>
54 </tr>
55 <tr>
56     <th>出版社</th>
57     <td>
58         <input type="text" name="publish_press"
value="${book.publish_press}"/>
59     </td>
60 </tr>
61 <tr>
62     <th>出版时间</th>
63     <td>
64         <input type="date" name="publish_date"
value="${book.publish_date}"/>
65     </td>
66 </tr>
67 <tr>
68     <th>借阅状态</th>
69     <td>
70         <select name="is_borrow">
71             <option value="">===请选择===</option>
72             <option value="false"
73                 <c:if test="${book.is_borrow==false}">
74                     selected
75                 </c:if>
76             >未借阅</option>
77             <option value="true"
78                 <c:if test="${book.is_borrow==true}">
79                     selected
80                 </c:if>
81             >已借阅</option>
82         </select>
83     </td>
84 </tr>
85 <tr>
86     <td colspan="2" align="center">
87         <input type="submit" value="保存修改" />
88         <input type="reset" value="取消"/>
89     </td>
90 </tr>
91 </table>
92 </td>
93 </tr>
94 </table>
95 </form>
96 </body>
97 </html>
98

```

2-3 实现修改保存数据

daoの根据主键修改图书信息

```
1  @Override
2      public int update(BookInfo book) {
3          StringBuilder sb=new StringBuilder();
4          sb.append("update book_info set ");
5          sb.append(" book_code=?,book_name=?,book_type=?,book_author=?");
6          sb.append(" publish_press=?,publish_date=?,is_borrow=? where
book_id=?");
7          try {
8              return qr.update(sb.toString(),book.getBook_code(),
9                  book.getBook_name(),
10                 book.getBook_type(),
11                 book.getBook_author(),
12                 book.getPublish_press(),
13                 book.getPublish_date(),
14                 //isIs_borrow getter针对boolean列自动生成的方法命名规范
15                 book.isIs_borrow(),
16                 book.getId());
17          } catch (SQLException e) {
18              throw new RuntimeException(e);
19          }
20      }
```

service的实现图书修改

```
1      package cn.kgc.service.impl;
2
3      import cn.kgc.dao.BookInfoDao;
4      import cn.kgc.dao.impl.BookInfoDaoImpl;
5      import cn.kgc.domain.BookInfo;
6      import cn.kgc.service.BookInfoService;
7
8      import java.util.List;
9
10     /**
11      * @Author: lc
12      * @Date: 2022/5/17
13      * @Description: cn.kgc.service.impl
14      * @Version: 1.0
15      */
16     public class BookInfoServiceImpl implements BookInfoService {
17         private BookInfoDao dao=new BookInfoDaoImpl();
18
19         @Override
20         public boolean modify(BookInfo book) {
21             return dao.update(book)==1;
22         }
23     }
24
```

modify.jsp页面设计的隐藏域的设计

```
1  <%@ page contentType="text/html;charset=UTF-8" language="java" %>
2  <%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
3  <html>
4  <head>
5      <meta charset="UTF-8">
6      <title>修改图书信息</title>
7  </head>
8
9  <body>
10 <form
11     action="${pageContext.servletContext.contextPath}/SaveModifyBookInfoServlet"
12     method="post">
13     <!--隐藏域-->
14     <input type="hidden" value="${book.id}" name="id">
15     <table align="center" border="1px" cellspacing="0px" width="800px">
16         <tr>
17             <td>
18                 <h3 align="center">图书借阅系统</h3>
19                 <table border="1px" cellspacing="1px" width="800px">
20                     <tr>
21                         <th>图书编号</th>
22                         <td>
23                             <input type="text" name="book_code"
24                             value="${book.book_code}" />
25                         </td>
26                     </tr>
27                     <tr>
28                         <th>图书名称</th>
29                         <td>
30                             <input type="text" name="book_name"
31                             value="${book.book_name}" />
32                         </td>
33                     </tr>
34                     <tr>
35                         <th>图书分类</th>
36                         <td>
37                             <select name="book_type">
38                                 <option value="">===请选择===</option>
39                                 <!--
40                                 1.循环，将所有的图书分类显示
41                                 2.根据修改的图书分类，设置默认的选中项
42                                 --%>
43                                 <c:forEach items="${types}" var="type">
44                                     <option value="${type.id}"
45                                     <c:if
46                                     test="${book.book_type==type.id}">
47                                         selected
48                                     </c:if>
49                                     >${type.type_name}</option>
50                                 </c:forEach>
51                             </select>
52                         </td>
53                     </tr>
54                     <tr>
55                         <th>作者</th>
```

```

51         <td>
52             <input type="text" name="book_author"
value="${book.book_author}"/>
53         </td>
54     </tr>
55     <tr>
56         <th>出版社</th>
57         <td>
58             <input type="text" name="publish_press"
value="${book.publish_press}"/>
59         </td>
60     </tr>
61     <tr>
62         <th>出版时间</th>
63         <td>
64             <input type="date" name="publish_date"
value="${book.publish_date}"/>
65         </td>
66     </tr>
67     <tr>
68         <th>借阅状态</th>
69         <td>
70             <select name="is_borrow">
71                 <option value="">===请选择===</option>
72                 <option value="false"
73                     <c:if test="${book.is_borrow==false}">
74                         selected
75                     </c:if>
76                 >未借阅</option>
77                 <option value="true"
78                     <c:if test="${book.is_borrow==true}">
79                         selected
80                     </c:if>
81                 >已借阅</option>
82             </select>
83         </td>
84     </tr>
85     <tr>
86         <td colspan="2" align="center">
87             <input type="submit" value="保存修改" />
88             <input type="reset" value="取消"/>
89         </td>
90     </tr>
91 </table>
92 </td>
93 </tr>
94 </table>
95 </form>
96 </body>
97 </html>

```

Servletの获取修改后的图书信息并修改

```

1 package cn.kgc.controller; /**
2  * @Author: lc
3  * @Date: 2022/5/20
4  * @Description: ${PACKAGE_NAME}

```

```

5  * @Version: 1.0
6  */
7
8  import cn.kgc.domain.BookInfo;
9  import cn.kgc.service.impl.BookInfoServiceImpl;
10 import org.apache.commons.beanutils.BeanUtils;
11 import org.apache.commons.beanutils.ConvertUtils;
12 import org.apache.commons.beanutils.converters.DateConverter;
13
14 import javax.servlet.ServletException;
15 import javax.servlet.annotation.WebServlet;
16 import javax.servlet.http.HttpServlet;
17 import javax.servlet.http.HttpServletRequest;
18 import javax.servlet.http.HttpServletResponse;
19 import java.io.IOException;
20 import java.io.PrintWriter;
21
22 @WebServlet("/SaveModifyBookInfoServlet")
23 public class SaveModifyBookInfoServlet extends HttpServlet {
24     @Override
25     protected void doGet(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
26         request.setCharacterEncoding("utf-8");
27         response.setContentType("text/html;charset=utf-8");
28         //扩展:
29         BookInfo book=new BookInfo();
30         //亲力亲为赋值, 借助BeanUtils对象, 简化亲力亲为赋值的过程
31         try {
32             //针对日期格式, 日期格式转换器
33             DateConverter converter=new DateConverter();
34             converter.setPatterns(new String[]{"yyyy-MM-dd", "MM-dd-
yyyy", "yyyyMMdd", "yyyy年MM月dd日", "yyyy/MM/dd"});
35             ConvertUtils.register(converter, java.util.Date.class);
36
37             BeanUtils.populate(book, request.getParameterMap());
38             System.out.println(book);
39         } catch (Exception e) {
40             throw new RuntimeException(e);
41         }
42         boolean bool= new BookInfoServiceImpl().modify(book);
43
44         //响应体的方式, alert()
45         PrintWriter writer = response.getWriter();
46         writer.print("<script type='text/javascript'>alert('"+(bool?"修改成
功":"修改失败")+"'");</script>");
47         if(bool){
48             //重定向 302
49             //getServletContext().getContextPath()获取服务器上发布项目的名称
50
51             //response.sendRedirect(getServletContext().getContextPath()+"/index.jsp");
52             writer.print("<script
type='text/javascript'>location.href='"+getServletContext().getContextPath()
+ "/index.jsp';</script>");
53         }
54     }
55     @Override

```



```
56     protected void doPost(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
57         doGet(request, response);
58     }
59 }
60
```

常见异常

```
java.lang.RuntimeException: java.sql.SQLException: You have an error in your SQL syntax; check the manual that corresponds to your MySQL
server version for the right syntax to use near 'FROM book_info where book_info.book_id=1' at line 1 Query: SELECT book_info.book_id
id,book_info.book_code,book_info.book_name,book_info.book_type,book_info.book_author,book_info.publish_press,book_info.publish_date,
book_info.is_borrow, FROM book_info where book_info.book_id=? Parameters: [1]

    at cn.kgc.dao.impl.BookInfoDaoImpl.selectByPkId(BookInfoDaoImpl.java:55)
    at cn.kgc.service.impl.BookInfoServiceImpl.findBookInfoBy(BookInfoServiceImpl.java:25)
    at cn.kgc.test.BookInfoTester.testFindById(BookInfoTester.java:45) <25 internal lines>
Caused by: java.sql.SQLException: You have an error in your SQL syntax; check the manual that corresponds to your MySQL
server version for the right syntax to use near 'FROM book_info where book_info.book_id=1' at line 1 Query: SELECT book_info.book_id
id,book_info.book_code,book_info.book_name,book_info.book_type,book_info.book_author,book_info.publish_press,book_info.publish_date,
book_info.is_borrow, FROM book_info where book_info.book_id=? Parameters: [1]

    at org.apache.commons.dbutils.AbstractQueryRunner.rethrow(AbstractQueryRunner.java:392)
    at org.apache.commons.dbutils.QueryRunner.query(QueryRunner.java:351)
    at org.apache.commons.dbutils.QueryRunner.query(QueryRunner.java:289)
    at cn.kgc.dao.impl.BookInfoDaoImpl.selectByPkId(BookInfoDaoImpl.java:53)
    ... 27 more
```

HTTP Status 500 - org.apache.commons.beanutils.ConversionException: DateConverter does not support default String to 'Date' conversion.

type Exception report

message org.apache.commons.beanutils.ConversionException: DateConverter does not support default String to 'Date' conversion.

description The server encountered an internal error that prevented it from fulfilling this request.

exception

```
java.lang.RuntimeException: org.apache.commons.beanutils.ConversionException: DateConverter does not support default String to 'Date' conversion.
    cn.kgc.controller.AddBookInfoServlet.doGet(AddBookInfoServlet.java:73)
    cn.kgc.controller.AddBookInfoServlet.doPost(AddBookInfoServlet.java:90)
    javax.servlet.http.HttpServlet.service(HttpServlet.java:647)
    javax.servlet.http.HttpServlet.service(HttpServlet.java:728)
```

BeanUtils实现form封装为book对象时，无法自主处理日期格式
程序员手动添加：日期格式转换器

root cause

```
org.apache.commons.beanutils.ConversionException: DateConverter does not support default String to 'Date' conversion.
    org.apache.commons.beanutils.converters.DateConverter.toDate(DateConverter.java:468)
    org.apache.commons.beanutils.converters.DateConverter.convertToType(DateConverter.java:343)
    org.apache.commons.beanutils.converters.AbstractConverter.convert(AbstractConverter.java:156)
    org.apache.commons.beanutils.converters.ConverterFacade.convert(ConverterFacade.java:60)
    org.apache.commons.beanutils.ConvertUtilsBean.convert(ConvertUtilsBean.java:470)
    org.apache.commons.beanutils.BeanUtilsBean.setProperty(BeanUtilsBean.java:1008)
    org.apache.commons.beanutils.BeanUtilsBean.populate(BeanUtilsBean.java:830)
    org.apache.commons.beanutils.BeanUtils.populate(BeanUtils.java:433)
    cn.kgc.controller.AddBookInfoServlet.doGet(AddBookInfoServlet.java:70)
    cn.kgc.controller.AddBookInfoServlet.doPost(AddBookInfoServlet.java:90)
    javax.servlet.http.HttpServlet.service(HttpServlet.java:647)
    javax.servlet.http.HttpServlet.service(HttpServlet.java:728)
```

课程总结

1 图书新增功能

2 图书修改功能

3 BeanUtils.populate()功能

课程预习

1.Session和Cookie

2.文件上传和下载

3 监听器和过滤器

4 ajax
