Jonathan Perry

BCIS 3680
Enterprise-Oriented Programming
Final Project

Perry, Jonathan

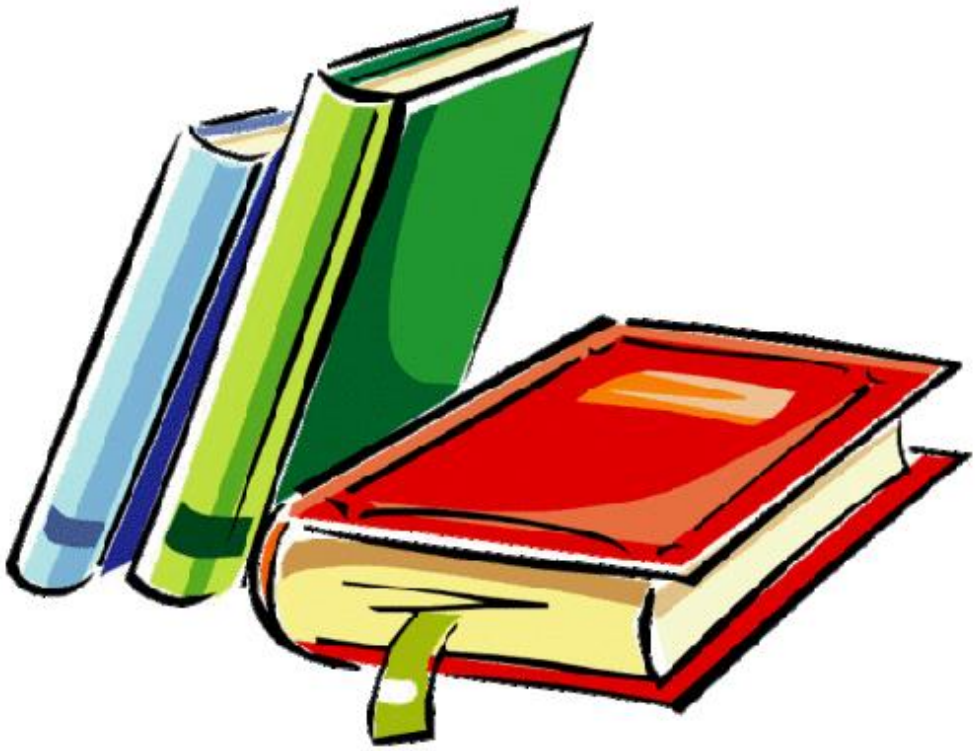
Contents

Default.html

Webpage:

Books Galore Library System

Login

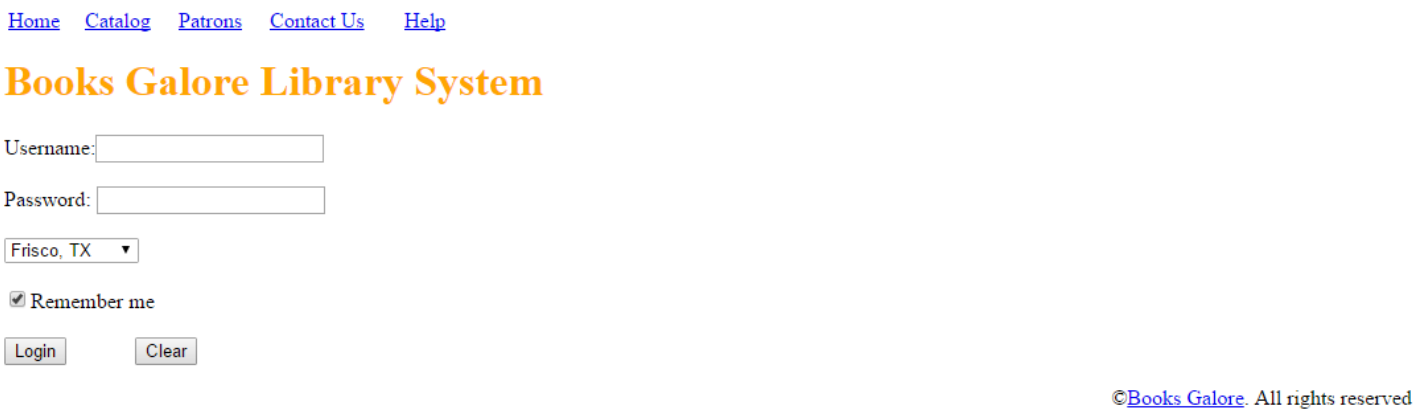


Code:

```
<html>
    <head>
        <title>Welcome to Books Galore</title>
    </head>
    <body>
    <h1><font color = "orange"> Books Galore Library System</font></h1>
    <h3><a href="login.jsp">Login</a></h3>
    <br>
    <a href="login.jsp"><img src="images/books.jpg"></a>
    </body>
</html>
```

Login.jsp

Webpage:



Code:

```
<html>
    <head><title>User Login</title>
    <script type = "text/javascript">
function validateForm(form)
{
    var u = document.login.username.value;
    var p = document.login.password.value;

    if(u == "")
    {
        alert("You must enter your user name!");
        return false;
    }
    else if(pass == "")
    {
    alert("You must enter your password!");
    return false;
    }
    else
    {
        response.sendRedirect("main.htm");
    }

}
</script>
    </head>
    <body>
        <%@ include file="header.htm" %></br>
        <h1><font color = "orange">Books Galore Library System</font></h1>
        <form id="login" name="login" method="post" action="main.jsp" onSubmit="return
validateForm()">
        Username:<INPUT TYPE="text" name = "username"></INPUT><BR><BR>
        Password: <INPUT TYPE="password" name = "password"></INPUT><BR><BR>
        <select name ="location"><option value ="4">Kalua, HI</option>
                                 <option value="3">Sanoma, CA</option>
                                 <option value ="2" selected>Frisco, TX</option>
                                 <option value ="1">Oviedo, FL</option></select>
                                 </br></br>
        <input type="checkbox" NAME = "checkbox" checked/>Remember me<BR><BR>
        <INPUT type="submit" VALUE=
"Login"</INPUT>            
        <INPUT type="reset" value= "Clear" </INPUT>
        </FORM>
        <%@ include file="footer.htm" %>
    </body>
</html>
```

main.jsp

Webpage:

Home Catalog Patrons Contact Us Help

Books Galore Library System

John Doe, welcome to the library system!

Your time zone is Central. Current time is Jan 10, 2016 4:30 PM

Manage Catalog

Code:

```
<%@ page import="java.util.GregorianCalendar, wu.andy.DateAssistant" %>

<html>
    <head><title>Library System Main Page</title></head>
    <body> <%@ include file="header.htm" %>
    <br>
    <h1><font color = "orange"> Books Galore Library System</font></h1>
<%
GregorianCalendar current = new GregorianCalendar();
String name = request.getParameter("username");
String pass = request.getParameter("password");
String location = request.getParameter("location");

if ( pass.equals("JPerry"))
{
out.println(name + ", welcome to the library system!");
out.println("<br /><br /> Your time zone is ");
session.setAttribute("name", name);
switch (Integer.parseInt(location))
{
case 1:
out.println("Eastern. Current time is ");
out.println(DateAssistant.formatDate(current, 1) + " ");
break;
case 2:
out.println("Central. Current time is ");
out.println(DateAssistant.formatDate(current, 2) + " ");
break;
case 3:
out.println("Pacific. Current time is ");
out.println(DateAssistant.formatDate(current, 3) + " ");
break;
case 4:
out.println("Hawaii. Current time is ");
out.println(DateAssistant.formatDate(current, 4) + " ");
break;
}
}
else
{
response.sendRedirect("login.jsp");
}
%>
<br>
<br>
<a href="catalog.jsp">Manage Catalog</a>
<%@ include file="footer.htm" %>
    </body>
</html>
```

catalog.jsp

Webpage:

Home Catalog Patrons Contact Us Help

Hi, John Doe!

Library Catalog

	Bar Code	Call Number	Book Title	Available
○	101	LB2351.2 .F475 2011	Crazy U	Yes
○	102	HV8079.C65 C93 2008	Cyber Forensics	Yes
○	103	HA32.S572 2004	SPSS for Introductory Statistics	Yes
○	104	HG181.S37 1999	Securities Technology Handbook	No
○	105	HN90.I56.L37 2005	Darknet	No
○	106	BCJ971.B8 W5513 2010	Hagakure	Yes
○	107	QA76.9 .A25 M567 2005	The Art of Intrusion	Yes
○	108	BF76.7 .P83 2001	APA Publication Manual 5e	No

Add New Book Delete Selected

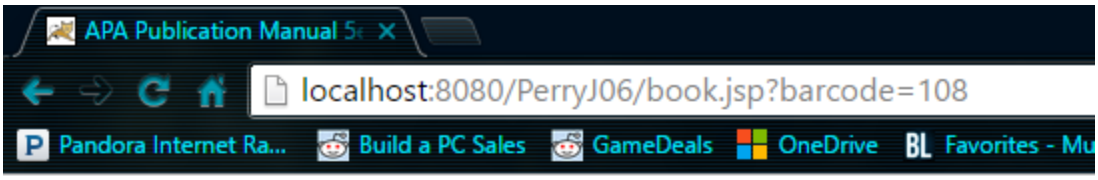
Code:

```
<%@ page import="java.sql.*" %>
<html>
<head>
<title>Libary Catalog</title>
<%@ include file="header.htm" %>
<script>
function validate(object)
{
    if(object.value == "Add New Book")
    {
        document.forms.main.action="newbook.jsp";
    }
    else if (object.value == "Delete Selected")
    {
        document.forms.main.action="deleted.jsp";
    }
    document.forms.main.submit();
}
</script>
</head>
<br>
<br>
<body>
<h3>Hi,  <%= (session.getAttribute("name"))%>!</h3>
<h1><font color="orange">Library Catalog</font></h1>
<%
            Class.forName("com.mysql.jdbc.Driver").newInstance();
            String connection1 = "jdbc:mysql://192.168.0.21:3306/mysql?";
            ServletContext sc = getServletContext();
            Connection connection = DriverManager.getConnection(connection1,
            sc.getInitParameter("userName"), sc.getInitParameter("passWord"));
            Statement statement = connection.createStatement();

            ResultSet rs;
            String barcode = "";
            String callNum;
            String bookTitle;
            int checkedOut;

            String sqlBooks = "SELECT * FROM book";

            rs = statement.executeQuery(sqlBooks);
%>

<br>
<form name = "main" method = "post">
<table border="1" cellpadding="7">
            <tr>
                  <th> </th>
                  <th>Bar Code</th>
                  <th>Call Number</th>
                  <th>Book Title</th>
                  <th>Available</th>
```

```
            </tr>

<%
      while (rs.next())
      {
            barcode = rs.getString("barcode");
            callNum = rs.getString("callno");
            bookTitle = rs.getString("title");
            checkedOut = rs.getInt("onloan");

            out.print("<td><input type = radio name = selectOption value = " + barcode +
            "></td>");
            out.print("<td align = center><a href = book.jsp");
            out.print("?barcode=" + barcode + ">" + barcode);
            out.print("</a></td>");
            out.print("<td>" + callNum + "</td>");
            out.print("<td>" + bookTitle + "</td>");
            if (checkedOut == 0)
            {out.println("<td align = center> No </td></tr>");}
            else
            {out.println("<td align = center> Yes </td></tr>");}
      }
%>
</table>
      <br> <br>
<input type = "submit" name = "add" value = "Add New Book" onclick = "validate(this)">
           
<input type = "submit" name = "delete" value = "Delete Selected" onclick =
"validate(this)"></p>

</form>
<br>
<br>
<%@ include file="footer.htm" %>
</body>
</html>
```

book.jsp

Webpage:



Home Catalog Patrons Contact Us Help

APA Publication Manual 5e

Barcode: 108
Call No: BF76.7 .P83 2001
Binding: Paperback
Price: $49.95
Publisher: APA

Loan History

Patron	Type	Out	Due	In
Mandy Ellis	Faculty	09/28/2012	12/27/2012	12/03/2012

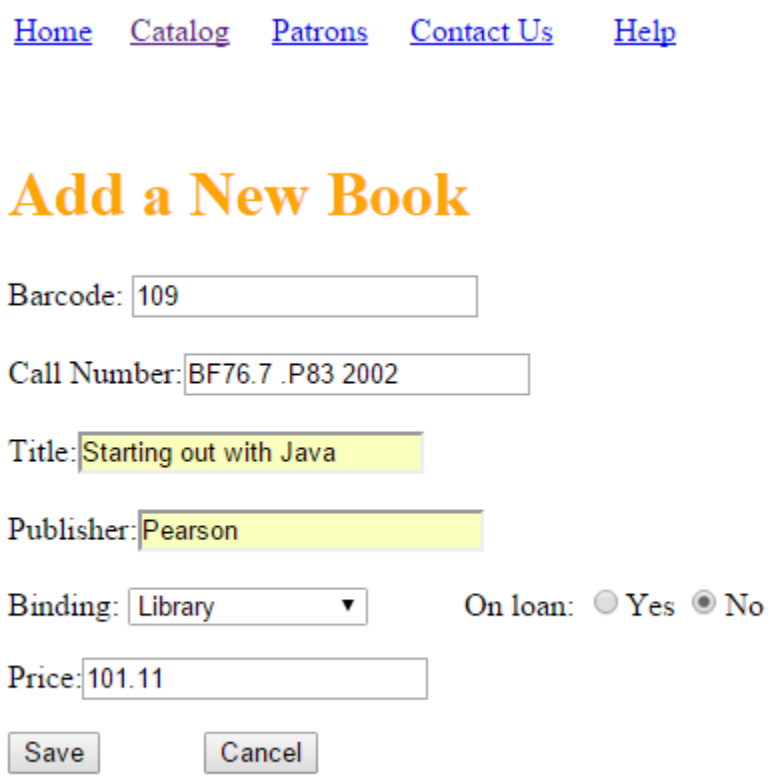
Code:

```
<%@ page import="java.sql.*, java.text.DecimalFormat, java.text.SimpleDateFormat,
wu.andy.DateAssistant" %>
<%
SimpleDateFormat sdf = new SimpleDateFormat("MM/dd/yyyy");
            String barcode = request.getParameter("barcode");
            Class.forName("com.mysql.jdbc.Driver").newInstance();
            ServletContext sc = getServletContext();
            Connection connection = DriverManager.getConnection(connection1,
            sc.getInitParameter("userName"), sc.getInitParameter("passWord"));
            Statement statement = connection.createStatement();

            ResultSet rs, rs1;
            String bookTitle, bookcode, callNo, bookBinding, publisher;
            double bookPrice;

            DecimalFormat df = new DecimalFormat("$#,##0.00");

                  String sqlBooks = "SELECT * FROM book ";
                  sqlBooks += " WHERE barcode = " + barcode + ";";

            rs = statement.executeQuery(sqlBooks);
            rs.next();
            bookTitle = rs.getString("title");
%>
<html>
<head>
<title><%= bookTitle %> Details </title>
<%@ include file="header.htm" %>
</head>
<body>
<br>
<%
bookcode = rs.getString("barcode");
callNo = rs.getString("callno");
bookBinding = rs.getString("binding");
bookPrice = rs.getDouble("price");
publisher = rs.getString("pub");
out.print("<h1><font color=\"orange\">" + bookTitle + "</font></h1>");
out.print("Barcode: " + barcode + "<br />");
out.print("Call No: " + callNo + "<br />");
out.print("Binding: " + bookBinding + "<br />");
out.print("<p1>Price: " + df.format(bookPrice) + "<br />");
out.print("<p1>Publisher: " + publisher + "<br />");
%>
<br>
<HR>
<h3><font color=orange>Loan History</font></h3>
<table cellpadding="4">
            <tr>
                  <th align = "left">Patron</th>
                  <th align = "left">Type</th>
                  <th align = "left">Out</th>
                  <th align = "left">Due</th>
                  <th align = "left">In</th>
```

```
            </tr>
<%
      String lastName, firstName, wholeName, schoolType;
      int interval;
      Date loanDate, returnDate;

                  String sqlBooks1 = "SELECT p.lname, p.fname, p.type, b.loandate,
                  b.returndate  FROM bookpatron b,  patron p";
                  sqlBooks1 += " WHERE b.pid = p.pid ";
                  sqlBooks1 += " AND b.barcode = " + barcode + ";";

            rs1 = statement.executeQuery(sqlBooks1);
while (rs1.next())
      {
lastName = rs1.getString("lname");
firstName = rs1.getString("fname");
wholeName = firstName.concat(" " + lastName);
schoolType = rs1.getString("type");
            if (schoolType.equals("undergrad"))
        {interval = 30;}
        else if (schoolType.equals("graduate"))
        {interval = 60;}
        else
        {interval = 90;}
loanDate = rs1.getDate("loandate");
returnDate = rs1.getDate("returnDate");

            out.print("<td>" + wholeName + "</td>");
            out.print("<td>" + schoolType.substring(0,1).toUpperCase() +
            schoolType.substring(1) + "</td>");
            out.print("<td>" + DateAssistant.convertDate(loanDate) + "</td>");
            out.print("<td>" + DateAssistant.calcNewDate(loanDate, interval) + "</td>");
            if (returnDate != null )
            out.print("<td>" + DateAssistant.convertDate(returnDate) + "</td></tr>");
            else
                  out.print("<td>On Loan </td></tr>");
      }
      connection.close();
%>
</table>
</body>
<%@ include file="footer.htm" %>
</html>
```

newbook.jsp

Webpage:



Code:

```
<%@ page import="java.sql.*" %>
<html>
<head><title>Add a New Book</title>
<%@ include file="header.htm" %>
<SCRIPT LANGUAGE = "JAVASCRIPT">
function validate()
{

    if (document.forms.main.callNumber.value == "")
    {
        alert("Call number cannot be empty.");
        return false;
    }
    else if (document.forms.main.title.value == "")
    {
        alert("Title cannot be empty.");
        return false;
    }
    else if (document.forms.main.publisher.value == "")
    {
```

```
            alert("Publisher cannot be empty.");
            return false;
        }
        else if (document.forms.main.binding.value <=1)
        {
            alert("You must select a value for binding.");
            return false;
        }
        else if (document.forms.main.price.value < 0)
        {
            alert("You must enter in a positive number.")
            return false;
        }
        else if (document.forms.main.price.value == "")
        {
            alert("Price cannot be empty.")
            return false;
        }
        return true;
}
</SCRIPT>
</script>
</head>
<br>
<br>
<body>
<%
                Class.forName("com.mysql.jdbc.Driver").newInstance();
                String connection1 = "jdbc:mysql://192.168.0.21:3306/mysql?";
                ServletContext sc = getServletContext();
                Connection connection = DriverManager.getConnection(connection1,
                sc.getInitParameter("userName"), sc.getInitParameter("passWord"));
                Statement statement = connection.createStatement();

                ResultSet rs = statement.executeQuery("SELECT max( barcode ) AS \"maxBook\"
                FROM book");
                int maxBookId = 0;
                while (rs.next())
                {
                      maxBookId = rs.getInt("maxBook");
                }
                session.setAttribute("maxBookID", maxBookId + 1);
%>
<br />
<h1><font color="orange">Add a New Book</font></h1>
<FORM NAME = "main" ACTION = "added.jsp" METHOD ="POST">
Barcode: </td><INPUT TYPE="text" name = "barCode" value ="<%= String.valueOf(maxBookId+1)
%>" readonly></INPUT><br><br>
Call Number:<INPUT TYPE="text" name = "callNumber"></INPUT><br><br>
Title:<INPUT TYPE="text" name = "title"></INPUT><br><br>
Publisher:<INPUT TYPE="text" name = "publisher"></INPUT><br>
<p>Binding:  <select name ="binding"><option value ="4">Hardcover</option>
                                                <option value="3">Library</option>
                                                <option value ="2">Paperback</option>
                                                <option value ="1"selected>Select a
```

```
binding</option></select>           
On loan:
<input type="radio" name="yesNo"  value="1">Yes</input>
<input type="radio" name="yesNo" value="0"checked>No</input></p>
Price:<INPUT TYPE="text" name = "price"></INPUT>
<p>
<input type = "submit" VALUE = "Save" onclick ="return validate()"></input>    
       
<input type = "reset" VALUE = "Cancel" ></input></p>
</form>

<br />
<br />
<%@ include file="footer.htm" %>
</body>
</html>
```

added.jsp

Webpage:

Home Catalog Patrons Contact Us Help

Catalog Update

New Book Updated Successfully.

Code:

```
<%@ page import="java.sql.*, java.text.DecimalFormat, java.text.SimpleDateFormat,
wu.andy.DateAssistant" %>
<%
            String barCodeDB = String.valueOf(session.getAttribute("maxBookID"));
            Class.forName("com.mysql.jdbc.Driver").newInstance();
            String connection1 = "jdbc:mysql://192.168.0.21:3306/mysql?";
            ServletContext sc = getServletContext();
            Connection connection = DriverManager.getConnection(connection1,
            sc.getInitParameter("userName"), sc.getInitParameter("passWord"));
            PreparedStatement ps = null;

            String callNumberDB = request.getParameter("callNumber");
            String titleDB = request.getParameter("title");
            String publisherDB = request.getParameter("publisher");
            String bindingDB = request.getParameter("binding");
            int newBindingDB = Integer.parseInt(bindingDB);
            String bindingType = "";
                  if (newBindingDB == 2)
                  {
                        bindingType = "Hardcover";
                  }
                  else if (newBindingDB == 3)
                  {
                        bindingType = "Library";
                  }
                  else if (newBindingDB == 4)
                  {
                        bindingType = "Paperback";
                  }
            String yesNoDB = request.getParameter("yesNo");
            String priceDB = request.getParameter("price");
            String insertStatement = "INSERT into book (barcode, callno, title, pub,
            binding, onloan, price) values (?,?,?,?,?,?,?)";
                  ps = connection.prepareStatement(insertStatement);
                  ps.setString(1, barCodeDB);
                  ps.setString(2, callNumberDB);
                  ps.setString(3, titleDB);
                  ps.setString(4, publisherDB);
                  ps.setString(5, bindingType);
                  ps.setString(6, yesNoDB);
                  ps.setString(7, priceDB);
                  int i = 0;
                  i = ps.executeUpdate();

%>
<html>
<head><title>New Book Added </title>
<%@ include file="header.htm" %>
</head>
<body>
<br>
<h2><font color=orange>Catalog Update</font></h2><br><br>
<%
      String message = "";
```

```
        if ( i == 0)
        {
              message = "Databased failed to update. Please try again.";
        }
        else if (i == 1)
        message = "New Book Updated Successfully.";
              
        out.print(message);
%>
</body>
<%@ include file="footer.htm" %>
</html>
```

Navigation From catalog.jsp to deleted.jsp

catalog.jsp webpage:

Home Catalog Patrons Contact Us Help

Hi, John Doe!

Library Catalog

	Bar Code	Call Number	Book Title	Available
○	101	LB2351.2 .F475 2011	Crazy U	Yes
○	102	HV8079.C65 C93 2008	Cyber Forensics	Yes
○	103	HA32.S572 2004	SPSS for Introductory Statistics	Yes
○	104	HG181.S37 1999	Securities Technology Handbook	No
○	105	HN90.I56.L37 2005	Darknet	No
○	106	BCJ971.B8 W5513 2010	Hagakure	Yes
○	107	QA76.9 .A25 M567 2005	The Art of Intrusion	Yes
○	108	BF76.7 .P83 2001	APA Publication Manual 5e	No
◉	109	BF76.7 .P83 2002	Starting out with Java	No

Add New Book Delete Selected

deleted.jsp webpage:

Home Catalog Patrons Contact Us Help

Catalog Update

Book deleted successfully.

Code:

```
<%@ page import="java.sql.*, java.text.DecimalFormat, java.text.SimpleDateFormat,
wu.andy.DateAssistant" %>

<html>
<head><title> Book Deleted</title>
<%@ include file="header.htm" %>
</head>
<body>
<br><h2><font color=orange>Catalog Update</font></h2><br>
<%
            String selectedOption = request.getParameter("selectOption");
            Class.forName("com.mysql.jdbc.Driver").newInstance();
            String connection1 = "jdbc:mysql://192.168.0.21:3306/mysql?";
            ServletContext sc = getServletContext();
            Connection connection = DriverManager.getConnection(connection1,
            sc.getInitParameter("userName"), sc.getInitParameter("passWord"));
            String deleteSQL = "DELETE FROM book WHERE barcode =" + selectedOption;
            PreparedStatement ps = connection.prepareStatement(deleteSQL);
            int i = 0;
            i = ps.executeUpdate();
            if (i == 0)
            {out.print("<br>Book deletion was unsuccessful.");}
            else if (i ==1)
            {out.print("<br>Book deleted successfully.");}

%>
<br>
</body>
<%@ include file="footer.htm" %>
</html>
```

DateAssistant.java

```
/*  Demo for: Custom-Made Classes
    DateAssistant.class */

package wu.andy;

import java.util.GregorianCalendar;
import java.util.Locale;
import java.util.Date;
import java.util.TimeZone;
import java.text.DateFormat;
import java.util.Calendar;
import java.text.SimpleDateFormat;


public class DateAssistant
{

    public static GregorianCalendar convertDateString(String dateString)
    {
        /* This method converts an String argument that is in the "mm/dd/yyyy"
        format into a GregorianCalendar object. */
```

```
        // Declare the GregorianCalendar object to be returned
        GregorianCalendar gc;

        // To call the constructor of GregorianCalendar with month, day, and
        // year arguments, three int variables are needed.
        String monthString, dayString, yearString;
        int month, day, year;

        /* The substring method of the String class returns part of a String as
        a "substring". The charaters included in the substring are determined
        by their indexes in the original string, an index being the position of
        the character.
        For example: in the date string
        character:   m | m | / | d | d | / | y | y | y | y
        position:    0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

        The substring() method is overloaded. One version takes two position
        indexes as arguments: the first one marks the start position of the
        substring and the scond, ending. The tricky part is that the first
        index is inclusive whereas the second is exclusive. Another version
        takes only one position index as argument. In that case, this index
        marks the start position of the substring, and it will run all the way
        to tne end of the original string.
         */

        // Call the substring() method with two arguments. Since the month part
        // is the first two characters of the date string, the start and ending
        // indexes are 0 and 2.
        monthString = dateString.substring(0, 2);
        month = Integer.parseInt(monthString);
        //month = Integer.parseInt(dateString.substring(0,2)); // better

        // Get the day part.
        dayString = dateString.substring(3, 5);
        day = Integer.parseInt(dayString);

        // Get the year part. We may use the other version with one argument.
        yearString = dateString.substring(6);
        year = Integer.parseInt(yearString);

        // Now we have all three arguments for calling one of the constructors
        // of the GregorianCalendar class. Note that the GregorianCalendar class
        // counts months from 0 (i.e., January = Month 0; December = Month 11).
        gc = new GregorianCalendar(year, month - 1, day);

        return gc;
    }

    public static String formatDate(GregorianCalendar gc, int zone)
    {
        /* This method takes a GregorianCalendar object that represents a given
           point in time and displays it in a format that is determined by time
           zones:
                  Hawaii, Alaska, Pacific, Mountain, Central, Eastern
```

```
        To specify which time zone to use, pass one of the above six strings
        as the second argument when calling this method.
    */
    // Variables
    String dateDisplay;
    TimeZone tz;

    // Convert the GregorianCalendar object to Date object.
    Date d = gc.getTime();

    // Create the DateFormat object and set date display in U.S. style
    DateFormat df = null;
    df = DateFormat.getDateTimeInstance(
                    DateFormat.MEDIUM, DateFormat.SHORT, Locale.US);

    // Set default time zone as Central
    // This is a catch-all option so that if the user doesn't enter a string
    // that matches any of the six below, the display is set as Central
    tz = TimeZone.getTimeZone("America/Chicago");

    // Adjust the time zone for date display based on the zone names
    // passed into this method
     switch (zone)
    {
     case 1:
        tz = TimeZone.getTimeZone("America/New_York");
        break;

        case 2:
        tz = TimeZone.getTimeZone("America/Chicago");
        break;

        case 3:
        tz = TimeZone.getTimeZone("America/Los_Angeles");
        break;

        case 4:
        tz = TimeZone.getTimeZone("Pacific/Honolulu");
        break;
    }

    // Actaully set time zone for the DateFormat object
    df.setTimeZone(tz);

    // Create the display string
    dateDisplay = df.format(d);

    // Return date display string
    return dateDisplay;
}

public static int calcInterval(String date1, String date2)
{
    /* This method takes two dates in the "mm/dd/yyyy" format and
    calculates the number of days between them. */
```

```
        // Variables
        int interval;
        GregorianCalendar gc1, gc2;
        long time1, time2;

        // Convert date strings to GregorianCalendar objects
        gc1 = convertDateString(date1);
        gc2 = convertDateString(date2);

        /* The getTimeInMillis() is an example of the methods the
        GregorianCalendar class inherits from the Calendar class. Similar
        to the Date class, it represents the number of milliseconds from the
        "epoch" (midnight, GMT, Jan. 1, 1970) as a long value. This is handy
        for:
        1. Comparing two dates, or
        2. Calculating the time interval between two dates.
        We use it for both in this method.
         */
        // Convert GregorianCalendar objects to longs
        time1 = gc1.getTimeInMillis();
        time2 = gc2.getTimeInMillis();

        // Calculate time interval in days
        // 1 day = 24 hours * 60 mins * 60 seconds * 1000 millisecs
        interval = (int) ((Math.max(time1, time2) - Math.min(time1, time2))
                    / (24 * 60 * 60 * 1000));
        return interval;
    }
    public static String calcNewDate(Date date, int days)
    {
    SimpleDateFormat sdf = new SimpleDateFormat("MM/dd/yyyy");
        Calendar cal = Calendar.getInstance();
        cal.setTime(date);
        cal.add(Calendar.DATE, days);
        String output = sdf.format(cal.getTime());
        return output;
    }
    public static String convertDate(Date day)
{
SimpleDateFormat formatter = new SimpleDateFormat("MM/dd/yyyy");
String date = formatter.format(day);
return date;
}
}
```