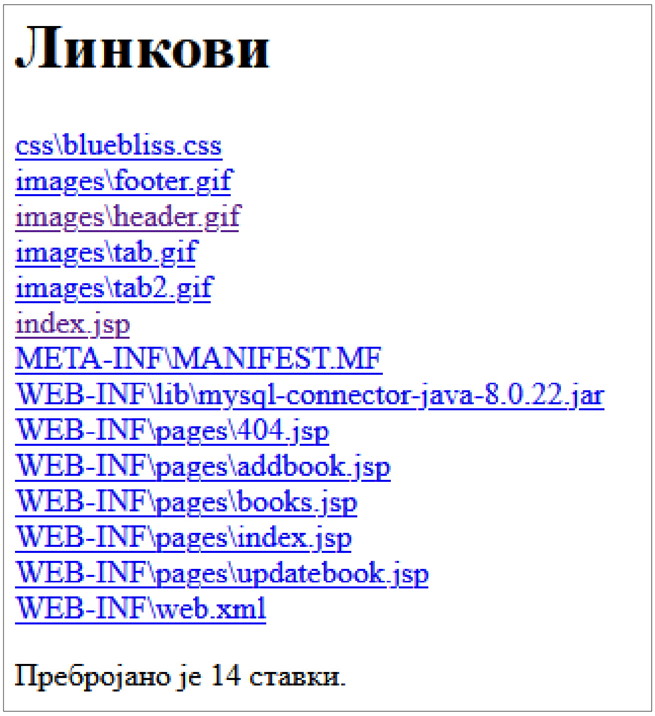
ИНТЕРНЕТ ПРОГРАМИРАЊЕ
ПРЕДАВАЊА – 07 – ЈСП М2 МОДЕЛ
АНАЛИЗА И ПРГЛЕД ПРИМЈЕРА

РЕЗИМЕ

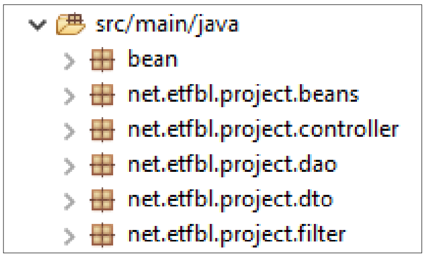


Линкови

css\bluebliss.css
images\footer.gif
images\header.gif
images\tab.gif
images\tab2.gif
index.jsp
META-INF\MANIFEST.MF
WEB-INF\lib\mysql-connector-java-8.0.22.jar
WEB-INF\pages\404.jsp
WEB-INF\pages\addbook.jsp
WEB-INF\pages\books.jsp
WEB-INF\pages\index.jsp
WEB-INF\pages\updatebook.jsp
WEB-INF\web.xml

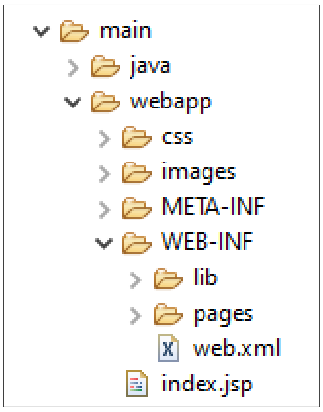
Пребројано је 14 ставки.

ЈАВА



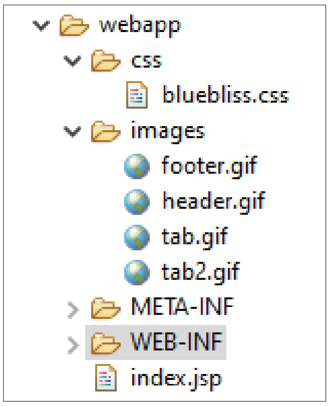
```
src/main/java
    bean
    net.etfbl.project.beans
    net.etfbl.project.controller
    net.etfbl.project.dao
    net.etfbl.project.dto
    net.etfbl.project.filter
```

ВЕБ



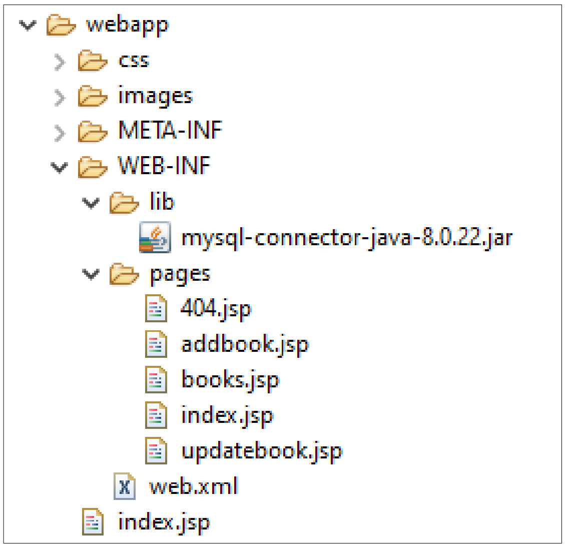
```
main
    java
    webapp
        css
        images
        META-INF
        WEB-INF
            lib
            pages
            web.xml
        index.jsp
```

## СТИЛОВИ И СЛИКЕ



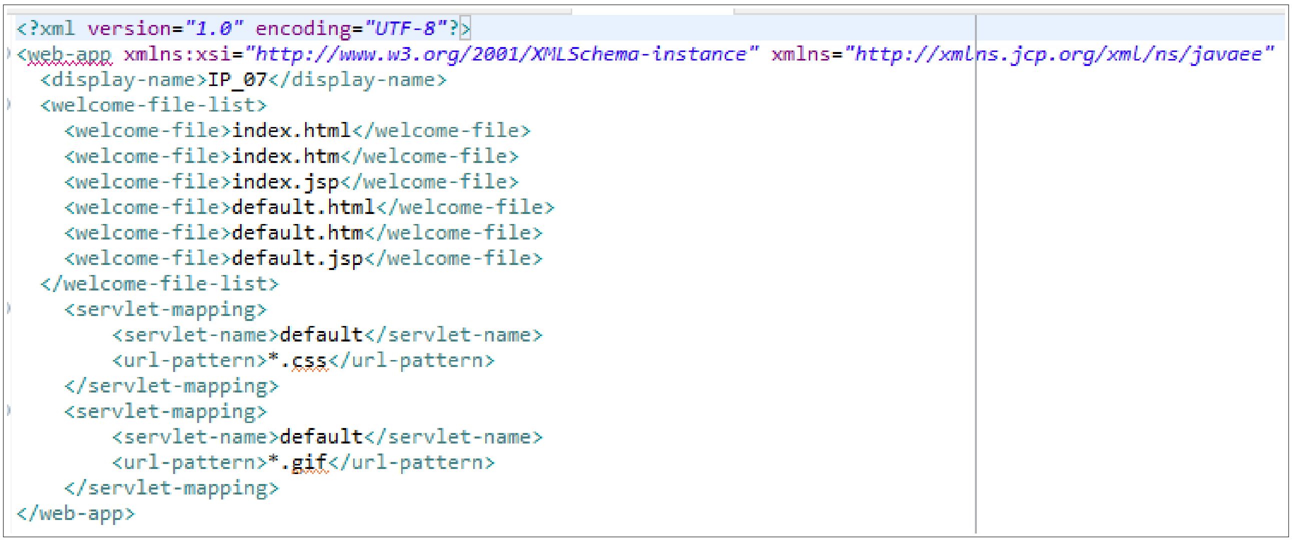
```
webapp
  css
    bluebliss.css
  images
    footer.gif
    header.gif
    tab.gif
    tab2.gif
  META-INF
  WEB-INF
  index.jsp
```

## БИБЛИОТЕКЕ И СТРАНИЦЕ



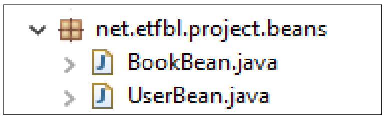
```
webapp
  css
  images
  META-INF
  WEB-INF
    lib
      mysql-connector-java-8.0.22.jar
    pages
      404.jsp
      addbook.jsp
      books.jsp
      index.jsp
      updatebook.jsp
    web.xml
  index.jsp
```

## WEB.INF



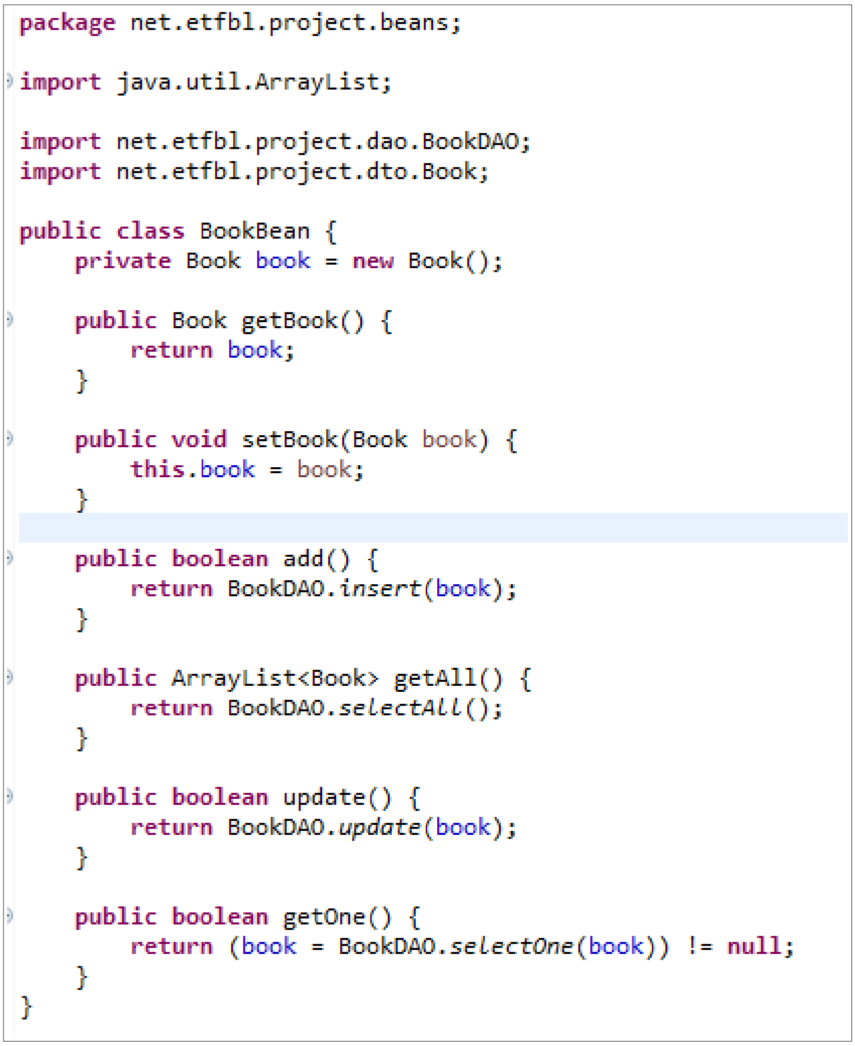
```
<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://xmlns.jcp.org/xml/ns/javaee"
  <display-name>IP_07</display-name>
  <welcome-file-list>
    <welcome-file>index.html</welcome-file>
    <welcome-file>index.htm</welcome-file>
    <welcome-file>index.jsp</welcome-file>
    <welcome-file>default.html</welcome-file>
    <welcome-file>default.htm</welcome-file>
    <welcome-file>default.jsp</welcome-file>
  </welcome-file-list>
    <servlet-mapping>
        <servlet-name>default</servlet-name>
        <url-pattern>*.css</url-pattern>
    </servlet-mapping>
    <servlet-mapping>
        <servlet-name>default</servlet-name>
        <url-pattern>*.gif</url-pattern>
    </servlet-mapping>
</web-app>
```

## BEANS



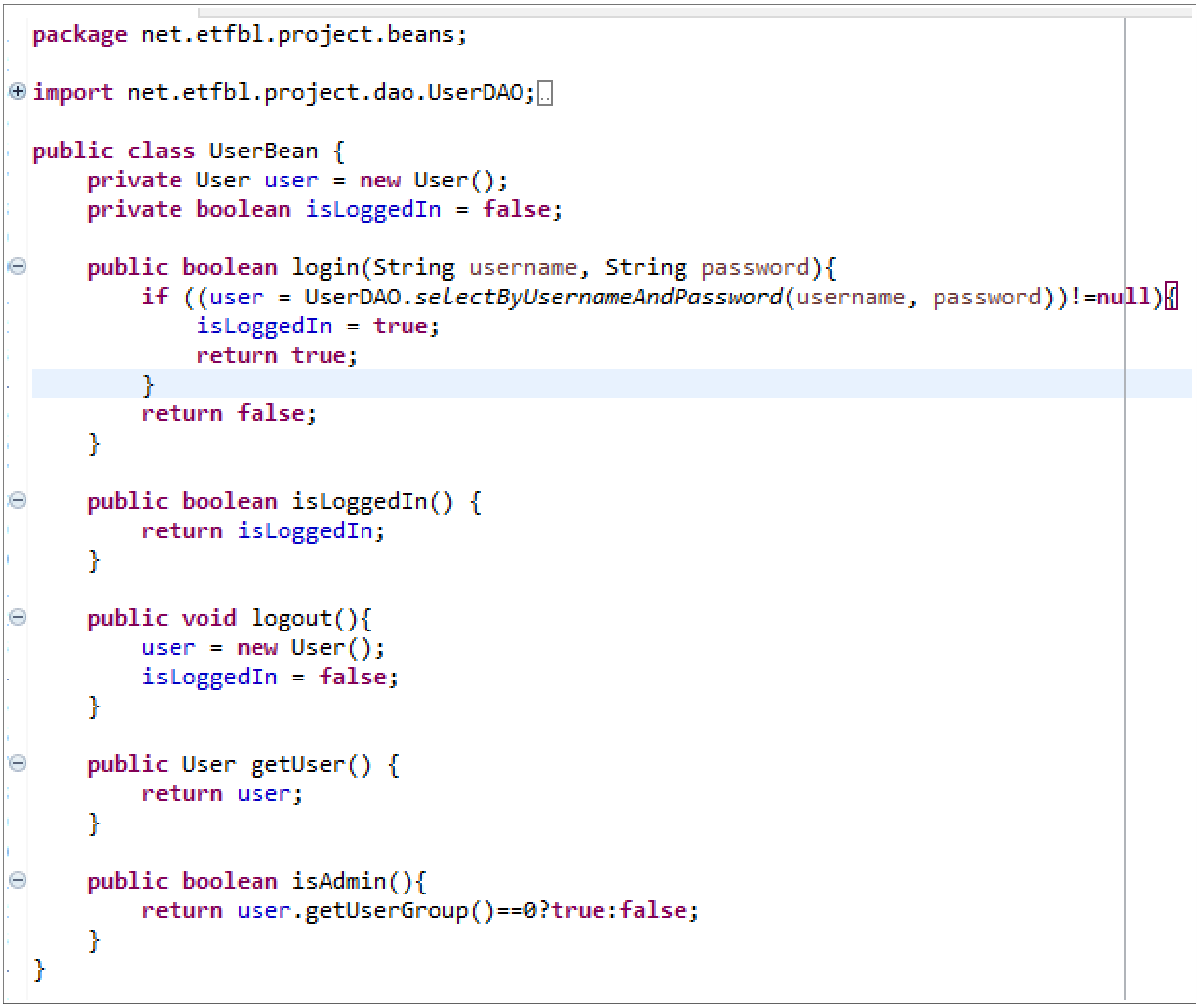
net.etfbl.project.beans
BookBean.java
UserBean.java

## BOOK_BEAN.JAVA



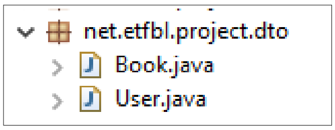
```java
package net.etfbl.project.beans;

import java.util.ArrayList;

import net.etfbl.project.dao.BookDAO;
import net.etfbl.project.dto.Book;

public class BookBean {
    private Book book = new Book();

    public Book getBook() {
        return book;
    }

    public void setBook(Book book) {
        this.book = book;
    }

    public boolean add() {
        return BookDAO.insert(book);
    }

    public ArrayList<Book> getAll() {
        return BookDAO.selectAll();
    }

    public boolean update() {
        return BookDAO.update(book);
    }

    public boolean getOne() {
        return (book = BookDAO.selectOne(book)) != null;
    }
}
```

USERS_BEAN.JAVA



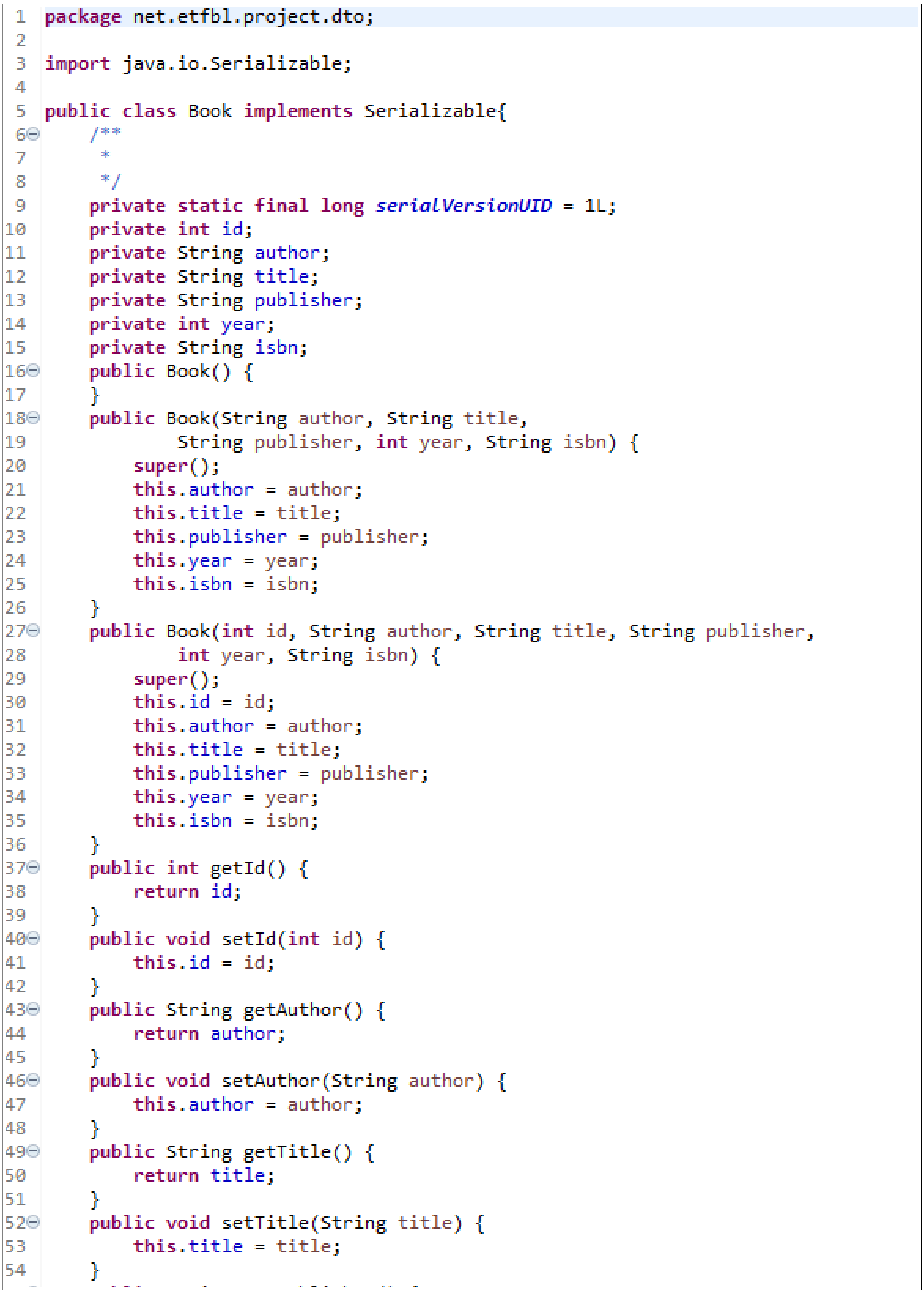
```java
package net.etfbl.project.beans;

import net.etfbl.project.dao.UserDAO;

public class UserBean {
    private User user = new User();
    private boolean isLoggedIn = false;

    public boolean login(String username, String password){
        if ((user = UserDAO.selectByUsernameAndPassword(username, password))!=null){
            isLoggedIn = true;
            return true;
        }
        return false;
    }

    public boolean isLoggedIn() {
        return isLoggedIn;
    }

    public void logout(){
        user = new User();
        isLoggedIn = false;
    }

    public User getUser() {
        return user;
    }

    public boolean isAdmin(){
        return user.getUserGroup()==0?true:false;
    }
}
```

DATA OBJECTS

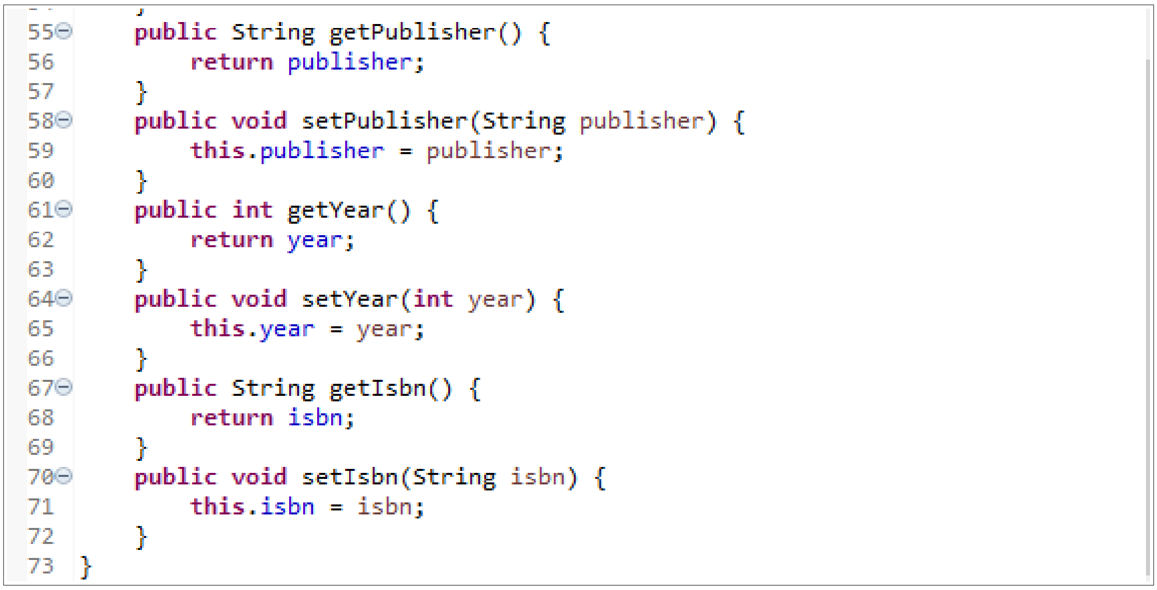


```
net.etfbl.project.dto
    Book.java
    User.java
```

BOOK.JAVA

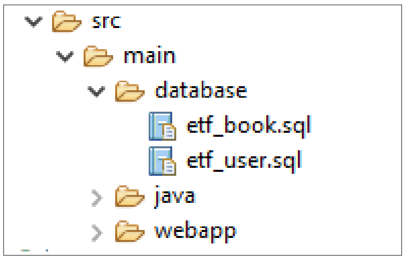


```java
package net.etfbl.project.dto;

import java.io.Serializable;

public class Book implements Serializable{
    /**
     * 
     */
    private static final long serialVersionUID = 1L;
    private int id;
    private String author;
    private String title;
    private String publisher;
    private int year;
    private String isbn;
    public Book() {
    }
    public Book(String author, String title,
            String publisher, int year, String isbn) {
        super();
        this.author = author;
        this.title = title;
        this.publisher = publisher;
        this.year = year;
        this.isbn = isbn;
    }
    public Book(int id, String author, String title, String publisher,
            int year, String isbn) {
        super();
        this.id = id;
        this.author = author;
        this.title = title;
        this.publisher = publisher;
        this.year = year;
        this.isbn = isbn;
    }
    public int getId() {
        return id;
    }
    public void setId(int id) {
        this.id = id;
    }
    public String getAuthor() {
        return author;
    }
    public void setAuthor(String author) {
        this.author = author;
    }
    public String getTitle() {
        return title;
    }
    public void setTitle(String title) {
        this.title = title;
    }
```



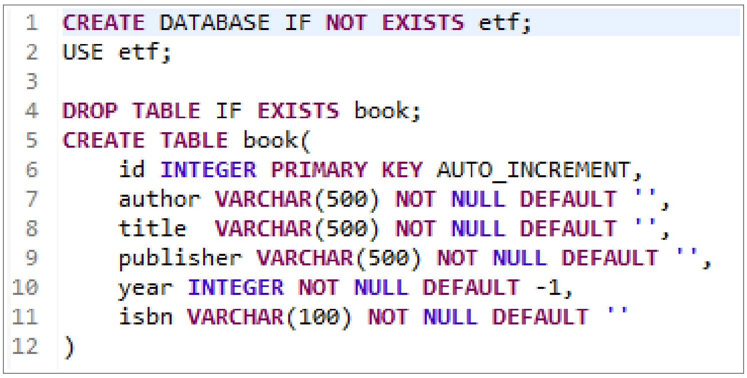
```java
    public String getPublisher() {
        return publisher;
    }
    public void setPublisher(String publisher) {
        this.publisher = publisher;
    }
    public int getYear() {
        return year;
    }
    public void setYear(int year) {
        this.year = year;
    }
    public String getIsbn() {
        return isbn;
    }
    public void setIsbn(String isbn) {
        this.isbn = isbn;
    }
}
```

DATABASE



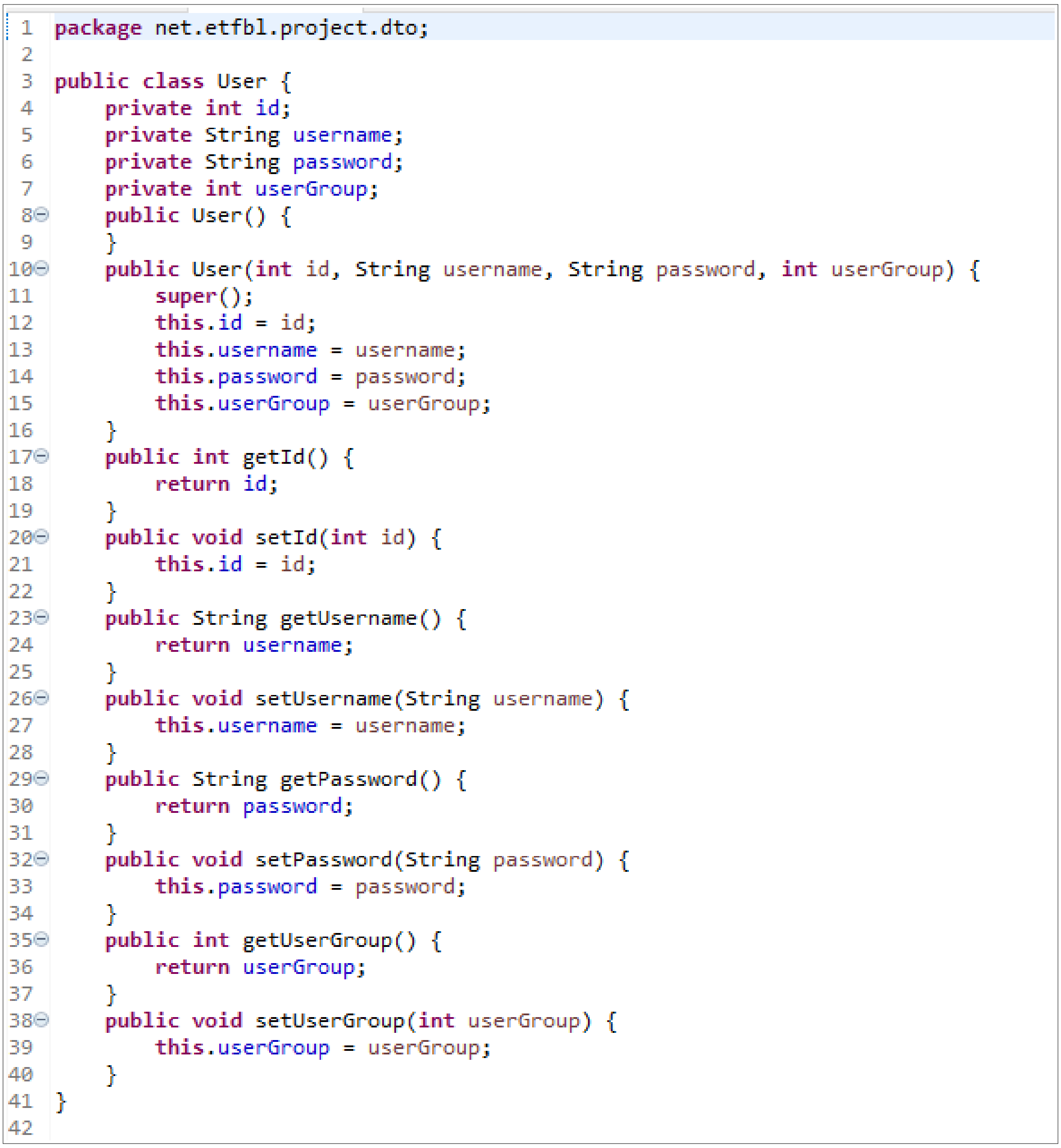
```
src
  main
    database
      etf_book.sql
      etf_user.sql
    java
    webapp
```

DATABASE – BOOK



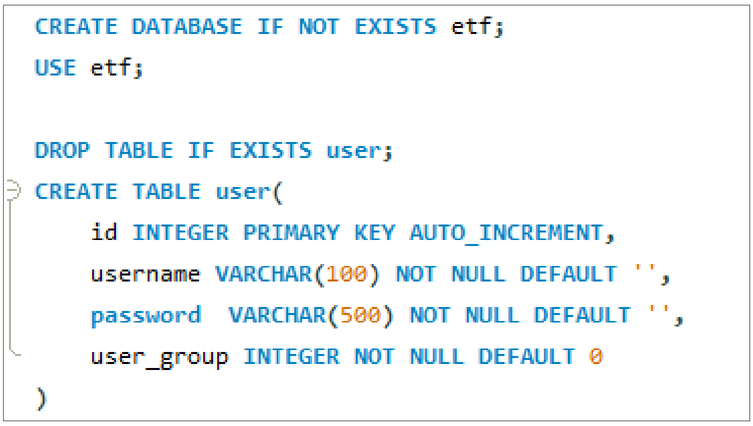
```sql
CREATE DATABASE IF NOT EXISTS etf;
USE etf;

DROP TABLE IF EXISTS book;
CREATE TABLE book(
    id INTEGER PRIMARY KEY AUTO_INCREMENT,
    author VARCHAR(500) NOT NULL DEFAULT '',
    title  VARCHAR(500) NOT NULL DEFAULT '',
    publisher VARCHAR(500) NOT NULL DEFAULT '',
    year INTEGER NOT NULL DEFAULT -1,
    isbn VARCHAR(100) NOT NULL DEFAULT ''
)
```

USER.JAVA



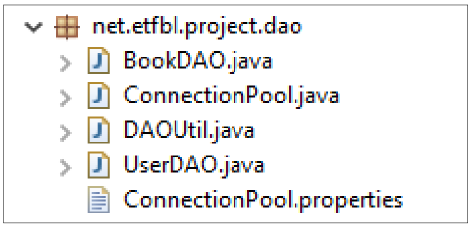
```java
package net.etfbl.project.dto;

public class User {
    private int id;
    private String username;
    private String password;
    private int userGroup;
    public User() {
    }
    public User(int id, String username, String password, int userGroup) {
        super();
        this.id = id;
        this.username = username;
        this.password = password;
        this.userGroup = userGroup;
    }
    public int getId() {
        return id;
    }
    public void setId(int id) {
        this.id = id;
    }
    public String getUsername() {
        return username;
    }
    public void setUsername(String username) {
        this.username = username;
    }
    public String getPassword() {
        return password;
    }
    public void setPassword(String password) {
        this.password = password;
    }
    public int getUserGroup() {
        return userGroup;
    }
    public void setUserGroup(int userGroup) {
        this.userGroup = userGroup;
    }
}

```

DATABASE – USER



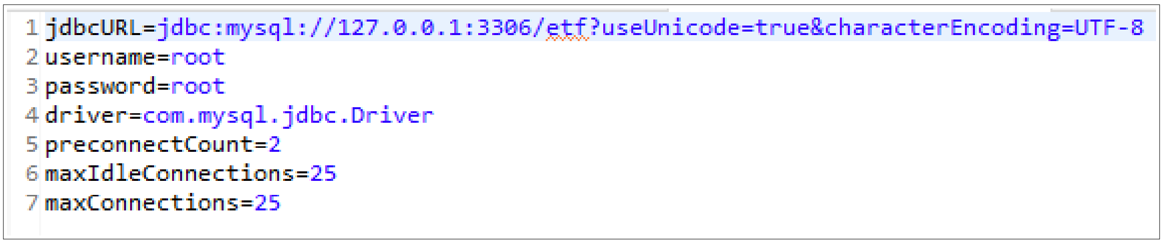
```sql
CREATE DATABASE IF NOT EXISTS etf;
USE etf;

DROP TABLE IF EXISTS user;
CREATE TABLE user(
    id INTEGER PRIMARY KEY AUTO_INCREMENT,
    username VARCHAR(100) NOT NULL DEFAULT '',
    password  VARCHAR(500) NOT NULL DEFAULT '',
    user_group INTEGER NOT NULL DEFAULT 0
)
```

## DATA ADAPTER



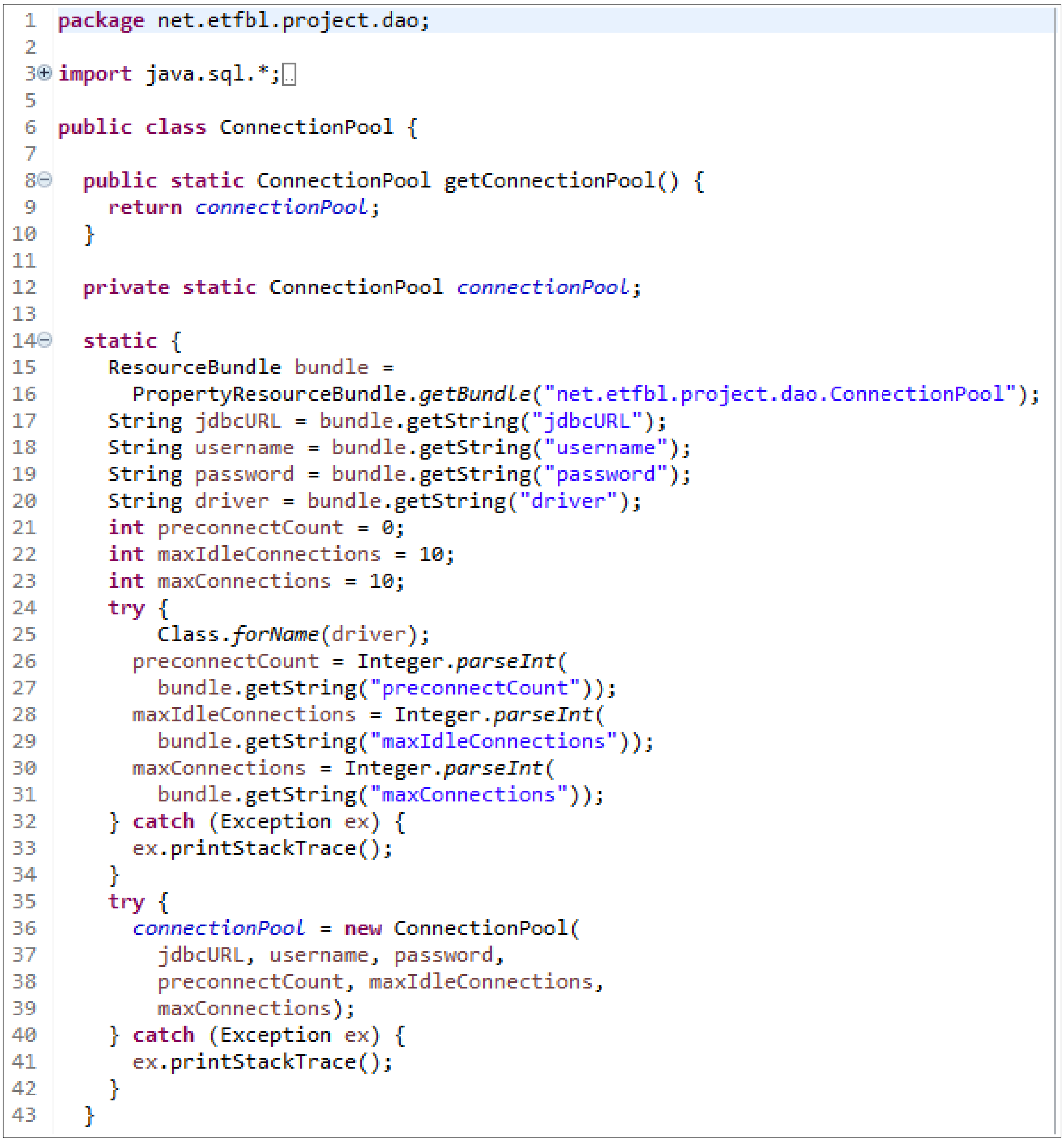
```
net.etfbl.project.dao
    BookDAO.java
    ConnectionPool.java
    DAOUtil.java
    UserDAO.java
    ConnectionPool.properties
```

## CONNECTION_POOL.PROPERTIES

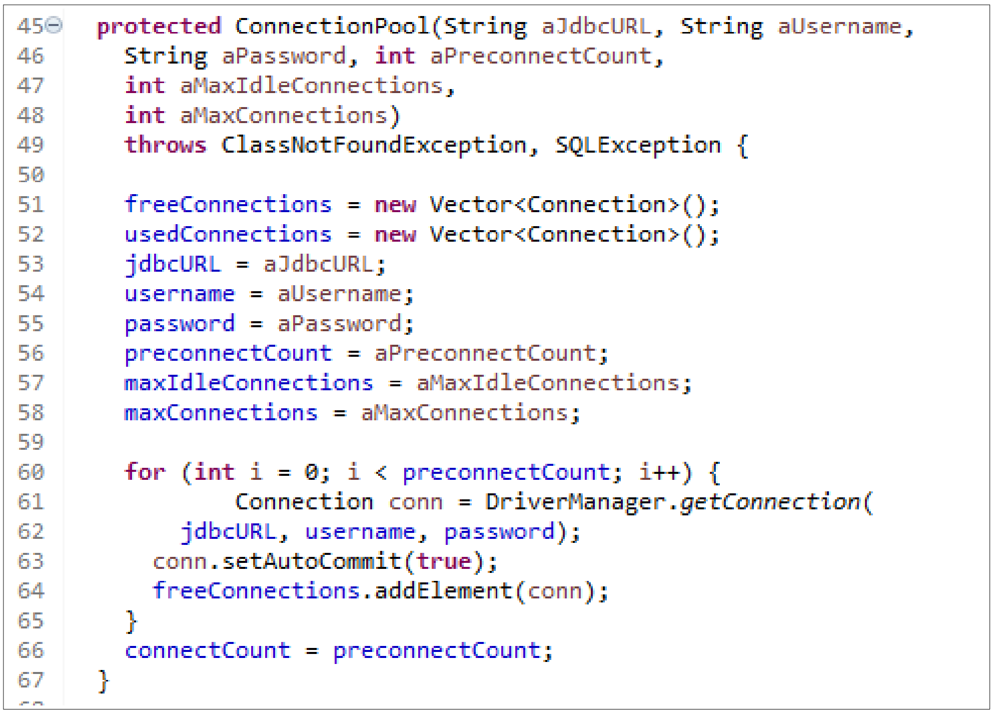


```
jdbcURL=jdbc:mysql://127.0.0.1:3306/etf?useUnicode=true&characterEncoding=UTF-8
username=root
password=root
driver=com.mysql.jdbc.Driver
preconnectCount=2
maxIdleConnections=25
maxConnections=25
```

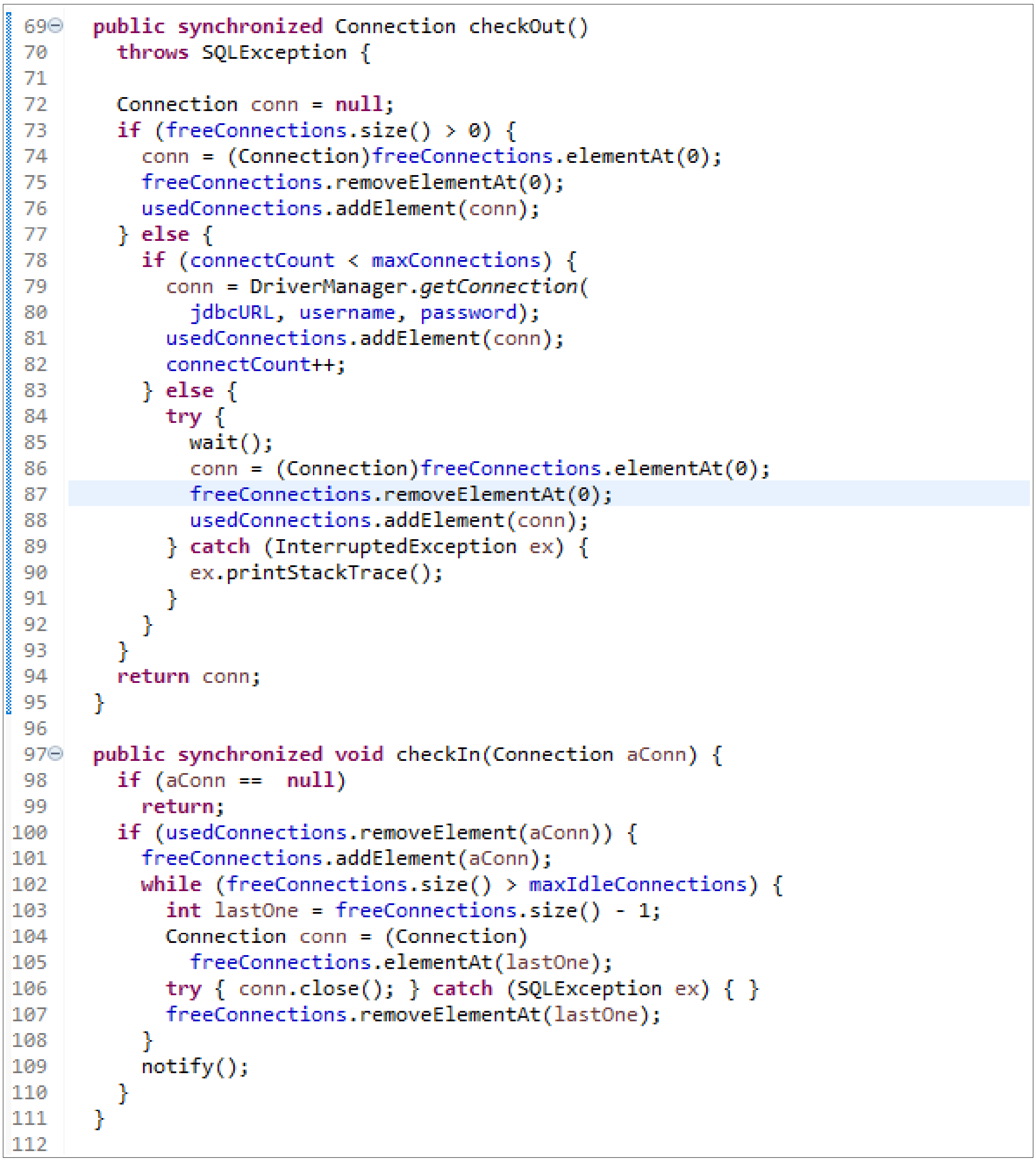
CONNECTION_POOL.JAVA



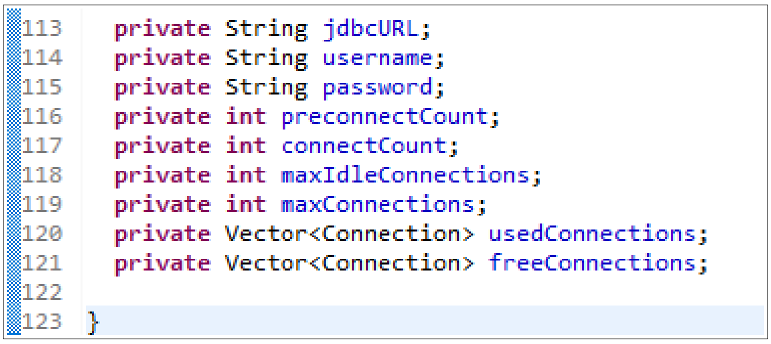
```java
package net.etfbl.project.dao;

import java.sql.*;

public class ConnectionPool {

  public static ConnectionPool getConnectionPool() {
    return connectionPool;
  }

  private static ConnectionPool connectionPool;

  static {
    ResourceBundle bundle =
      PropertyResourceBundle.getBundle("net.etfbl.project.dao.ConnectionPool");
    String jdbcURL = bundle.getString("jdbcURL");
    String username = bundle.getString("username");
    String password = bundle.getString("password");
    String driver = bundle.getString("driver");
    int preconnectCount = 0;
    int maxIdleConnections = 10;
    int maxConnections = 10;
    try {
        Class.forName(driver);
      preconnectCount = Integer.parseInt(
        bundle.getString("preconnectCount"));
      maxIdleConnections = Integer.parseInt(
        bundle.getString("maxIdleConnections"));
      maxConnections = Integer.parseInt(
        bundle.getString("maxConnections"));
    } catch (Exception ex) {
      ex.printStackTrace();
    }
    try {
      connectionPool = new ConnectionPool(
        jdbcURL, username, password,
        preconnectCount, maxIdleConnections,
        maxConnections);
    } catch (Exception ex) {
      ex.printStackTrace();
    }
  }
```



```java
protected ConnectionPool(String aJdbcURL, String aUsername,
  String aPassword, int aPreconnectCount,
  int aMaxIdleConnections,
  int aMaxConnections)
  throws ClassNotFoundException, SQLException {

  freeConnections = new Vector<Connection>();
  usedConnections = new Vector<Connection>();
  jdbcURL = aJdbcURL;
  username = aUsername;
  password = aPassword;
  preconnectCount = aPreconnectCount;
  maxIdleConnections = aMaxIdleConnections;
  maxConnections = aMaxConnections;

  for (int i = 0; i < preconnectCount; i++) {
          Connection conn = DriverManager.getConnection(
      jdbcURL, username, password);
    conn.setAutoCommit(true);
    freeConnections.addElement(conn);
  }
  connectCount = preconnectCount;
}
```

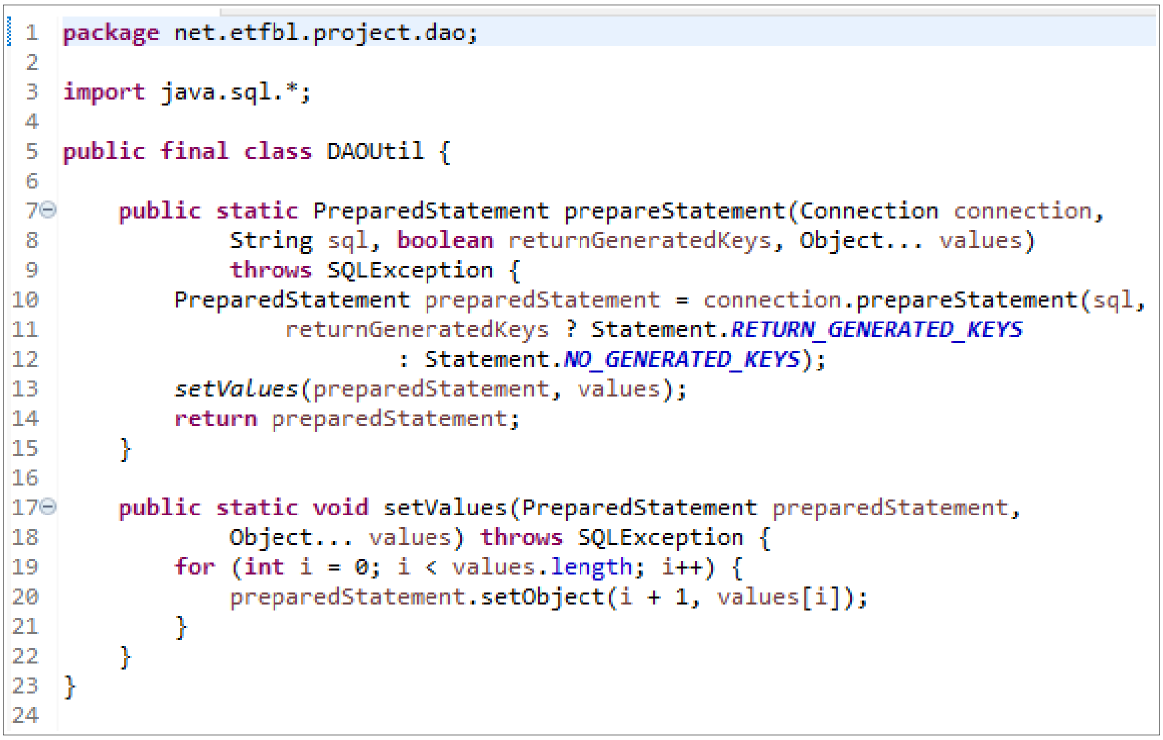


```java
  public synchronized Connection checkOut()
    throws SQLException {

    Connection conn = null;
    if (freeConnections.size() > 0) {
      conn = (Connection)freeConnections.elementAt(0);
      freeConnections.removeElementAt(0);
      usedConnections.addElement(conn);
    } else {
      if (connectCount < maxConnections) {
        conn = DriverManager.getConnection(
          jdbcURL, username, password);
        usedConnections.addElement(conn);
        connectCount++;
      } else {
        try {
          wait();
          conn = (Connection)freeConnections.elementAt(0);
          freeConnections.removeElementAt(0);
          usedConnections.addElement(conn);
        } catch (InterruptedException ex) {
          ex.printStackTrace();
        }
      }
    }
    return conn;
  }

  public synchronized void checkIn(Connection aConn) {
    if (aConn ==  null)
      return;
    if (usedConnections.removeElement(aConn)) {
      freeConnections.addElement(aConn);
      while (freeConnections.size() > maxIdleConnections) {
        int lastOne = freeConnections.size() - 1;
        Connection conn = (Connection)
          freeConnections.elementAt(lastOne);
        try { conn.close(); } catch (SQLException ex) { }
        freeConnections.removeElementAt(lastOne);
      }
      notify();
    }
  }

```



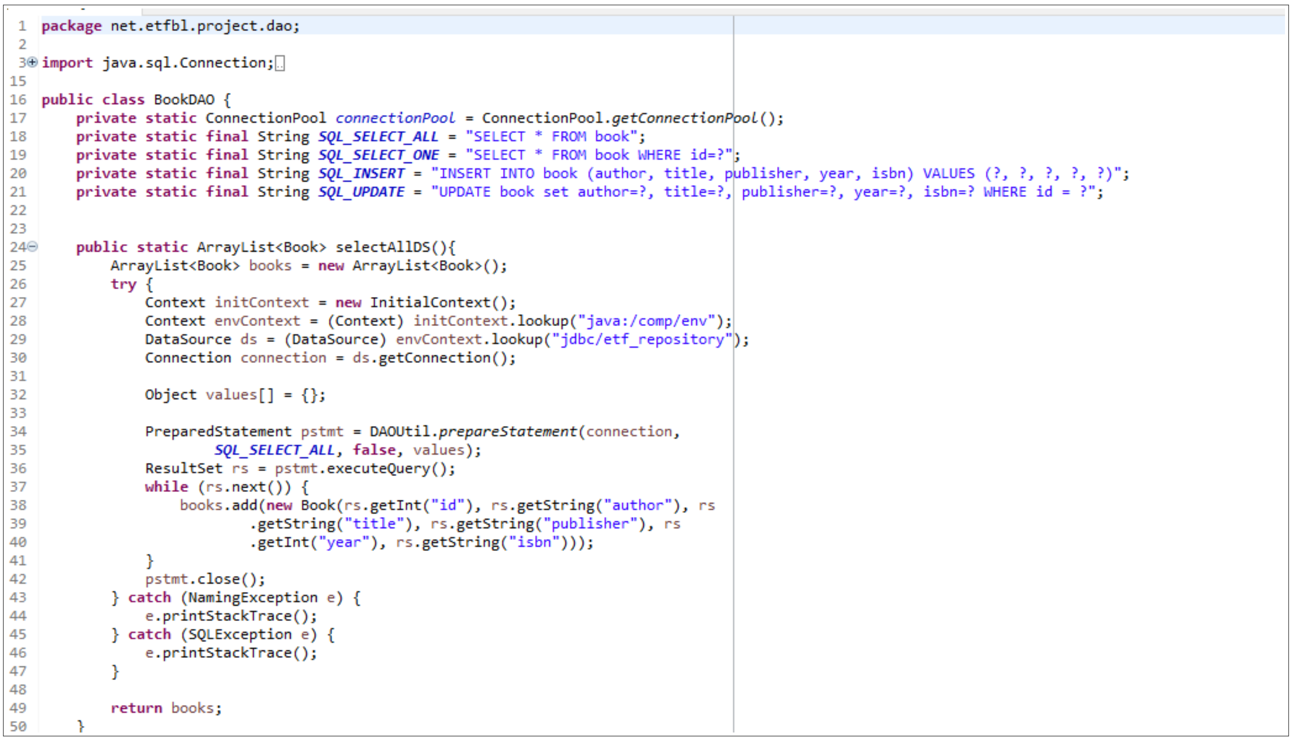
```java
  private String jdbcURL;
  private String username;
  private String password;
  private int preconnectCount;
  private int connectCount;
  private int maxIdleConnections;
  private int maxConnections;
  private Vector<Connection> usedConnections;
  private Vector<Connection> freeConnections;

}
```

DAO_UTIL.JAVA

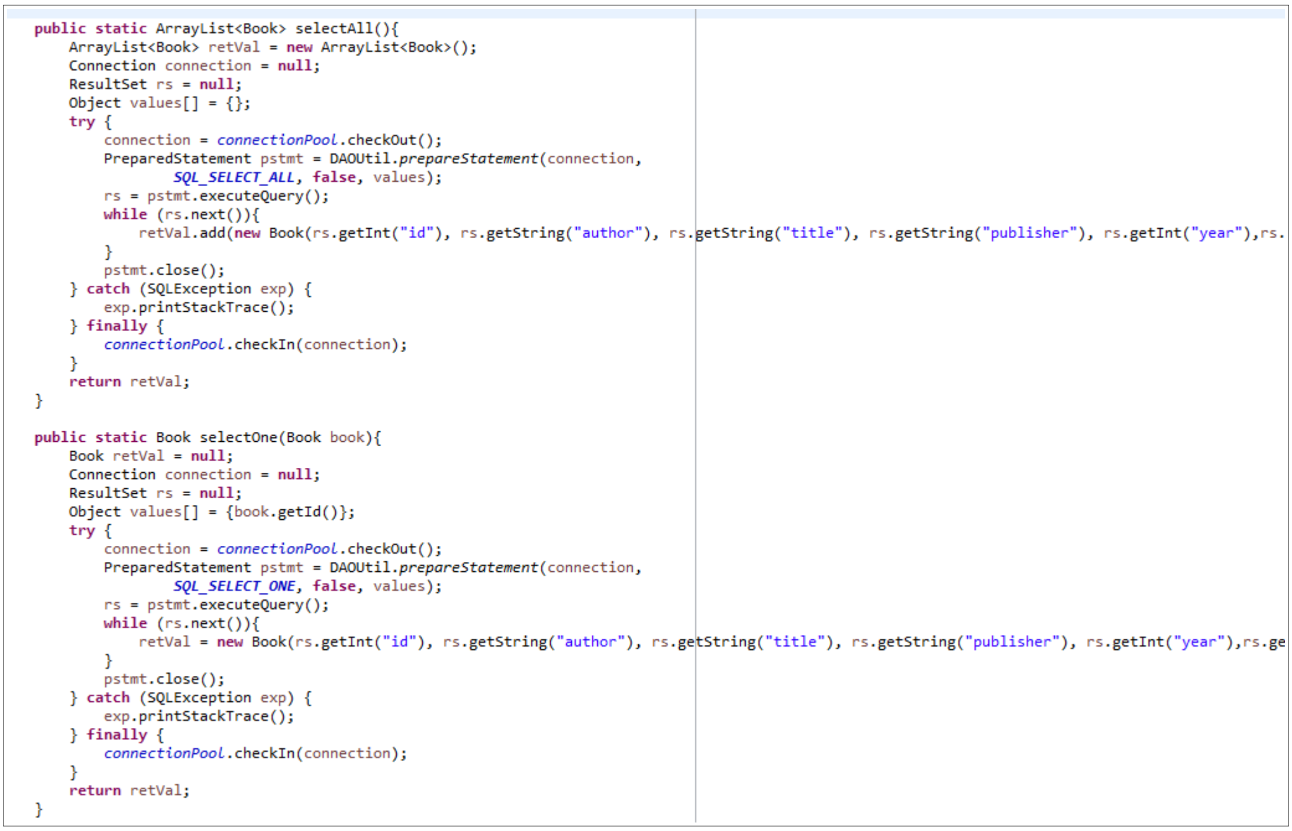


```java
package net.etfbl.project.dao;

import java.sql.*;

public final class DAOUtil {

    public static PreparedStatement prepareStatement(Connection connection,
            String sql, boolean returnGeneratedKeys, Object... values)
            throws SQLException {
        PreparedStatement preparedStatement = connection.prepareStatement(sql,
                returnGeneratedKeys ? Statement.RETURN_GENERATED_KEYS
                        : Statement.NO_GENERATED_KEYS);
        setValues(preparedStatement, values);
        return preparedStatement;
    }

    public static void setValues(PreparedStatement preparedStatement,
            Object... values) throws SQLException {
        for (int i = 0; i < values.length; i++) {
            preparedStatement.setObject(i + 1, values[i]);
        }
    }
}
```

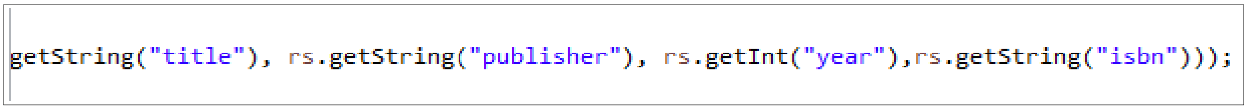
# BOOK_DAO.JAVA



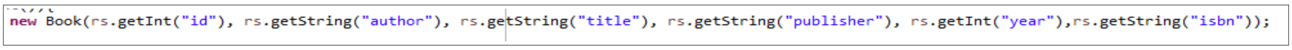
```java
package net.etfbl.project.dao;

import java.sql.Connection;
public class BookDAO {
    private static ConnectionPool connectionPool = ConnectionPool.getConnectionPool();
    private static final String SQL_SELECT_ALL = "SELECT * FROM book";
    private static final String SQL_SELECT_ONE = "SELECT * FROM book WHERE id=?";
    private static final String SQL_INSERT = "INSERT INTO book (author, title, publisher, year, isbn) VALUES (?, ?, ?, ?, ?)";
    private static final String SQL_UPDATE = "UPDATE book set author=?, title=?, publisher=?, year=?, isbn=? WHERE id = ?";


    public static ArrayList<Book> selectAllDS(){
        ArrayList<Book> books = new ArrayList<Book>();
        try {
            Context initContext = new InitialContext();
            Context envContext = (Context) initContext.lookup("java:/comp/env");
            DataSource ds = (DataSource) envContext.lookup("jdbc/etf_repository");
            Connection connection = ds.getConnection();

            Object values[] = {};

            PreparedStatement pstmt = DAOUtil.prepareStatement(connection,
                    SQL_SELECT_ALL, false, values);
            ResultSet rs = pstmt.executeQuery();
            while (rs.next()) {
                books.add(new Book(rs.getInt("id"), rs.getString("author"), rs
                        .getString("title"), rs.getString("publisher"), rs
                        .getInt("year"), rs.getString("isbn")));
            }
            pstmt.close();
        } catch (NamingException e) {
            e.printStackTrace();
        } catch (SQLException e) {
            e.printStackTrace();
        }

        return books;
    }
```



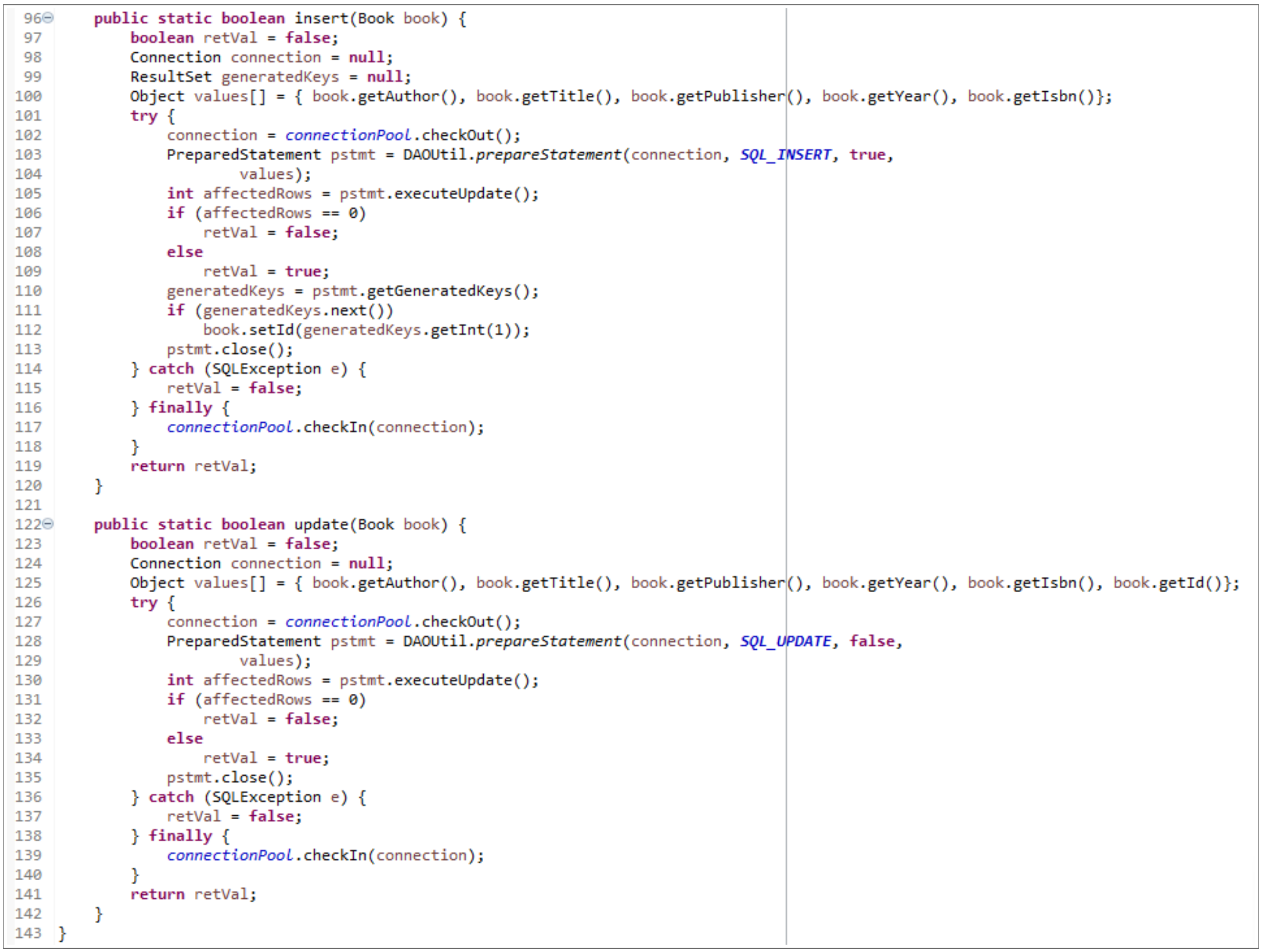
```java
    public static ArrayList<Book> selectAll(){
        ArrayList<Book> retVal = new ArrayList<Book>();
        Connection connection = null;
        ResultSet rs = null;
        Object values[] = {};
        try {
            connection = connectionPool.checkOut();
            PreparedStatement pstmt = DAOUtil.prepareStatement(connection,
                    SQL_SELECT_ALL, false, values);
            rs = pstmt.executeQuery();
            while (rs.next()){
                retVal.add(new Book(rs.getInt("id"), rs.getString("author"), rs.getString("title"), rs.getString("publisher"), rs.getInt("year"),rs.
            }
            pstmt.close();
        } catch (SQLException exp) {
            exp.printStackTrace();
        } finally {
            connectionPool.checkIn(connection);
        }
        return retVal;
    }

    public static Book selectOne(Book book){
        Book retVal = null;
        Connection connection = null;
        ResultSet rs = null;
        Object values[] = {book.getId()};
        try {
            connection = connectionPool.checkOut();
            PreparedStatement pstmt = DAOUtil.prepareStatement(connection,
                    SQL_SELECT_ONE, false, values);
            rs = pstmt.executeQuery();
            while (rs.next()){
                retVal = new Book(rs.getInt("id"), rs.getString("author"), rs.getString("title"), rs.getString("publisher"), rs.getInt("year"),rs.ge
            }
            pstmt.close();
        } catch (SQLException exp) {
            exp.printStackTrace();
        } finally {
            connectionPool.checkIn(connection);
        }
        return retVal;
    }
```



```java
getString("title"), rs.getString("publisher"), rs.getInt("year"),rs.getString("isbn")));
```

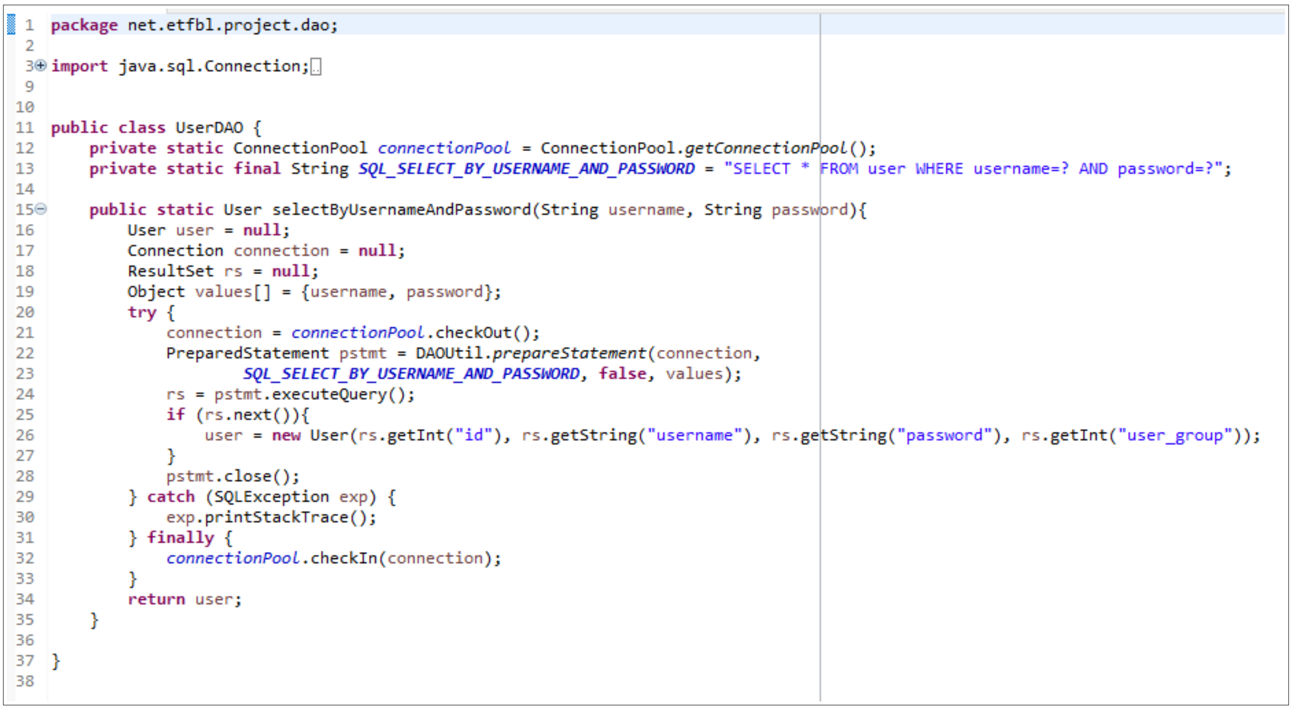


```java
new Book(rs.getInt("id"), rs.getString("author"), rs.getString("title"), rs.getString("publisher"), rs.getInt("year"),rs.getString("isbn"));
```



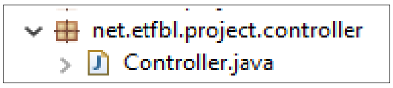
```java
    public static boolean insert(Book book) {
        boolean retVal = false;
        Connection connection = null;
        ResultSet generatedKeys = null;
        Object values[] = { book.getAuthor(), book.getTitle(), book.getPublisher(), book.getYear(), book.getIsbn()};
        try {
            connection = connectionPool.checkOut();
            PreparedStatement pstmt = DAOUtil.prepareStatement(connection, SQL_INSERT, true,
                    values);
            int affectedRows = pstmt.executeUpdate();
            if (affectedRows == 0)
                retVal = false;
            else
                retVal = true;
            generatedKeys = pstmt.getGeneratedKeys();
            if (generatedKeys.next())
                book.setId(generatedKeys.getInt(1));
            pstmt.close();
        } catch (SQLException e) {
            retVal = false;
        } finally {
            connectionPool.checkIn(connection);
        }
        return retVal;
    }

    public static boolean update(Book book) {
        boolean retVal = false;
        Connection connection = null;
        Object values[] = { book.getAuthor(), book.getTitle(), book.getPublisher(), book.getYear(), book.getIsbn(), book.getId()};
        try {
            connection = connectionPool.checkOut();
            PreparedStatement pstmt = DAOUtil.prepareStatement(connection, SQL_UPDATE, false,
                    values);
            int affectedRows = pstmt.executeUpdate();
            if (affectedRows == 0)
                retVal = false;
            else
                retVal = true;
            pstmt.close();
        } catch (SQLException e) {
            retVal = false;
        } finally {
            connectionPool.checkIn(connection);
        }
        return retVal;
    }
}
```

# USER_DAO.JAVA

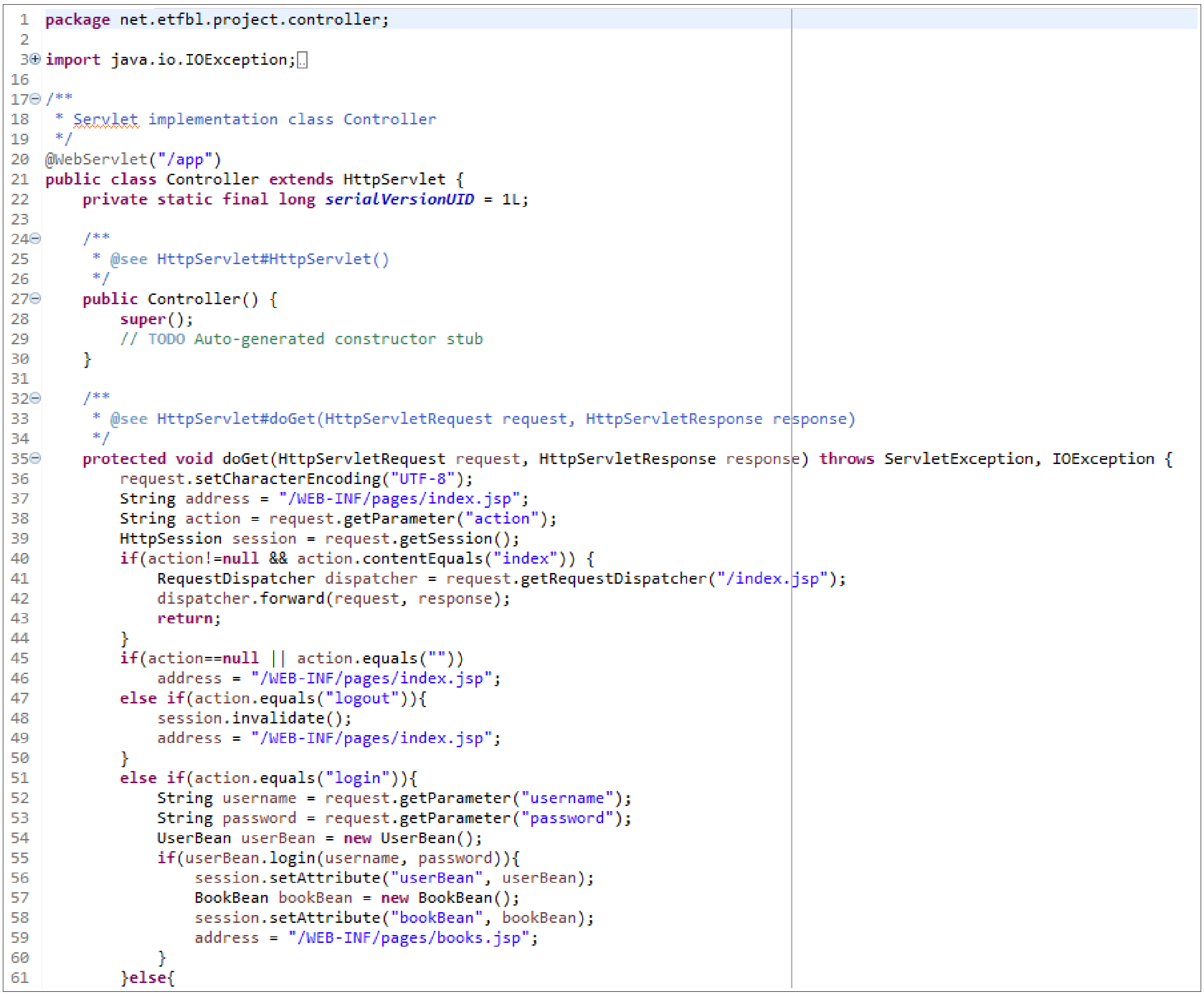


```java
package net.etfbl.project.dao;

import java.sql.Connection;


public class UserDAO {
    private static ConnectionPool connectionPool = ConnectionPool.getConnectionPool();
    private static final String SQL_SELECT_BY_USERNAME_AND_PASSWORD = "SELECT * FROM user WHERE username=? AND password=?";

    public static User selectByUsernameAndPassword(String username, String password){
        User user = null;
        Connection connection = null;
        ResultSet rs = null;
        Object values[] = {username, password};
        try {
            connection = connectionPool.checkOut();
            PreparedStatement pstmt = DAOUtil.prepareStatement(connection,
                    SQL_SELECT_BY_USERNAME_AND_PASSWORD, false, values);
            rs = pstmt.executeQuery();
            if (rs.next()){
                user = new User(rs.getInt("id"), rs.getString("username"), rs.getString("password"), rs.getInt("user_group"));
            }
            pstmt.close();
        } catch (SQLException exp) {
            exp.printStackTrace();
        } finally {
            connectionPool.checkIn(connection);
        }
        return user;
    }

}
```

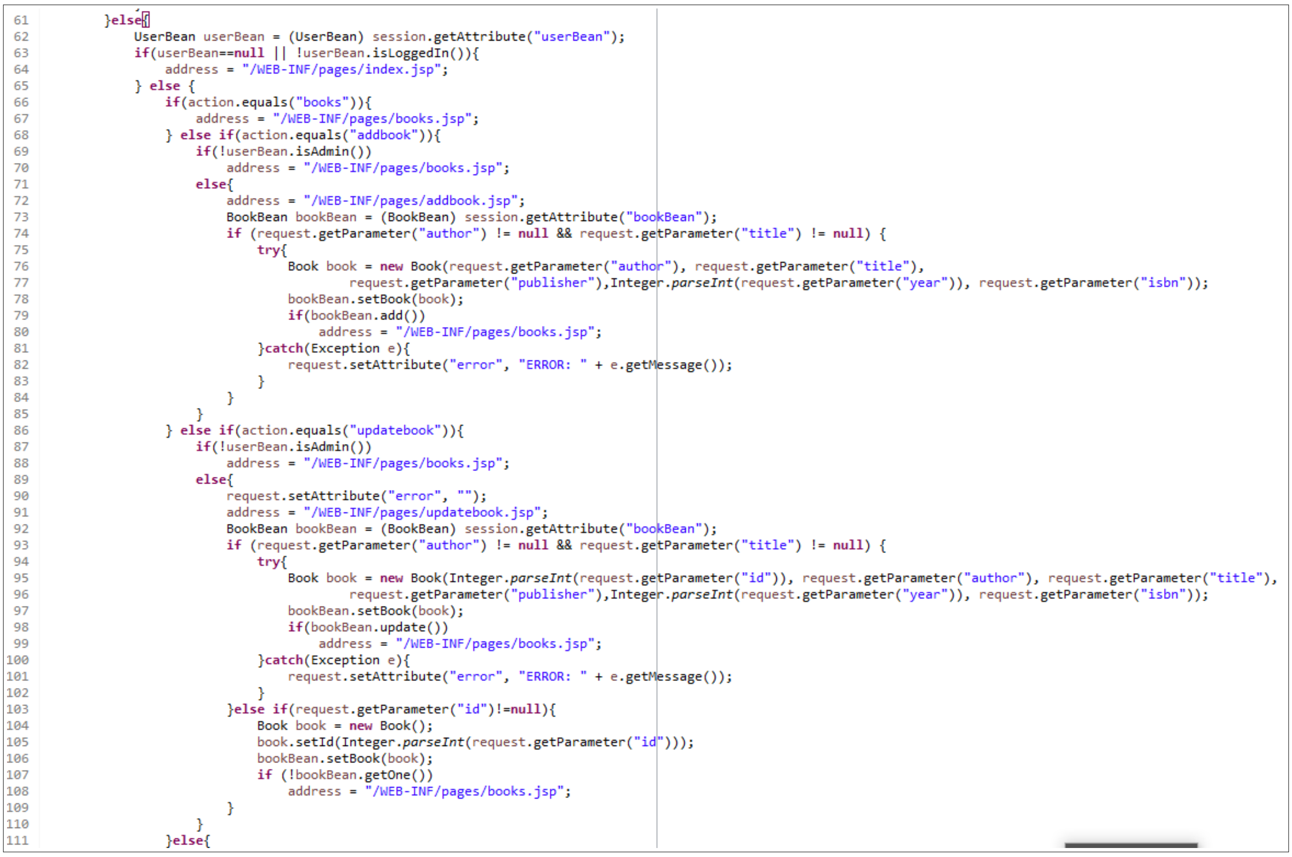
# CONTROLLER



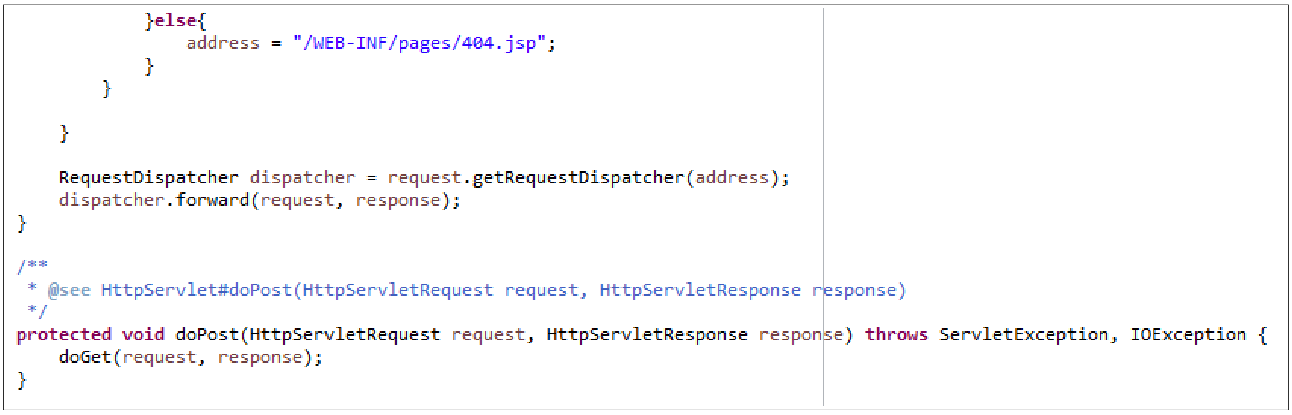
# CONTROLLER.JAVA



```java
package net.etfbl.project.controller;

import java.io.IOException;

/**
 * Servlet implementation class Controller
 */
@WebServlet("/app")
public class Controller extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public Controller() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
     */
    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        request.setCharacterEncoding("UTF-8");
        String address = "/WEB-INF/pages/index.jsp";
        String action = request.getParameter("action");
        HttpSession session = request.getSession();
        if(action!=null && action.contentEquals("index")) {
            RequestDispatcher dispatcher = request.getRequestDispatcher("/index.jsp");
            dispatcher.forward(request, response);
            return;
        }
        if(action==null || action.equals(""))
            address = "/WEB-INF/pages/index.jsp";
        else if(action.equals("logout")){
            session.invalidate();
            address = "/WEB-INF/pages/index.jsp";
        }
        else if(action.equals("login")){
            String username = request.getParameter("username");
            String password = request.getParameter("password");
            UserBean userBean = new UserBean();
            if(userBean.login(username, password)){
                session.setAttribute("userBean", userBean);
                BookBean bookBean = new BookBean();
                session.setAttribute("bookBean", bookBean);
                address = "/WEB-INF/pages/books.jsp";
            }
        }else{
```

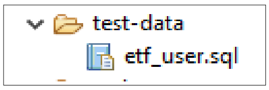


```java
        }else{
            UserBean userBean = (UserBean) session.getAttribute("userBean");
            if(userBean==null || !userBean.isLoggedIn()){
                address = "/WEB-INF/pages/index.jsp";
            } else {
                if(action.equals("books")){
                    address = "/WEB-INF/pages/books.jsp";
                } else if(action.equals("addbook")){
                    if(!userBean.isAdmin())
                        address = "/WEB-INF/pages/books.jsp";
                    else{
                        address = "/WEB-INF/pages/addbook.jsp";
                        BookBean bookBean = (BookBean) session.getAttribute("bookBean");
                        if (request.getParameter("author") != null && request.getParameter("title") != null) {
                            try{
                                Book book = new Book(request.getParameter("author"), request.getParameter("title"),
                                        request.getParameter("publisher"),Integer.parseInt(request.getParameter("year")), request.getParameter("isbn"));
                                bookBean.setBook(book);
                                if(bookBean.add())
                                    address = "/WEB-INF/pages/books.jsp";
                            }catch(Exception e){
                                request.setAttribute("error", "ERROR: " + e.getMessage());
                            }
                        }
                    }
                } else if(action.equals("updatebook")){
                    if(!userBean.isAdmin())
                        address = "/WEB-INF/pages/books.jsp";
                    else{
                        request.setAttribute("error", "");
                        address = "/WEB-INF/pages/updatebook.jsp";
                        BookBean bookBean = (BookBean) session.getAttribute("bookBean");
                        if (request.getParameter("author") != null && request.getParameter("title") != null) {
                            try{
                                Book book = new Book(Integer.parseInt(request.getParameter("id")), request.getParameter("author"), request.getParameter("title"),
                                        request.getParameter("publisher"),Integer.parseInt(request.getParameter("year")), request.getParameter("isbn"));
                                bookBean.setBook(book);
                                if(bookBean.update())
                                    address = "/WEB-INF/pages/books.jsp";
                            }catch(Exception e){
                                request.setAttribute("error", "ERROR: " + e.getMessage());
                            }
                        }else if(request.getParameter("id")!=null){
                            Book book = new Book();
                            book.setId(Integer.parseInt(request.getParameter("id")));
                            bookBean.setBook(book);
                            if (!bookBean.getOne())
                                address = "/WEB-INF/pages/books.jsp";
                        }
                    }
                }else{
```



```java
                }else{
                    address = "/WEB-INF/pages/404.jsp";
                }
            }

        }

        RequestDispatcher dispatcher = request.getRequestDispatcher(address);
        dispatcher.forward(request, response);
    }

    /**
     * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)
     */
    protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        doGet(request, response);
    }
```

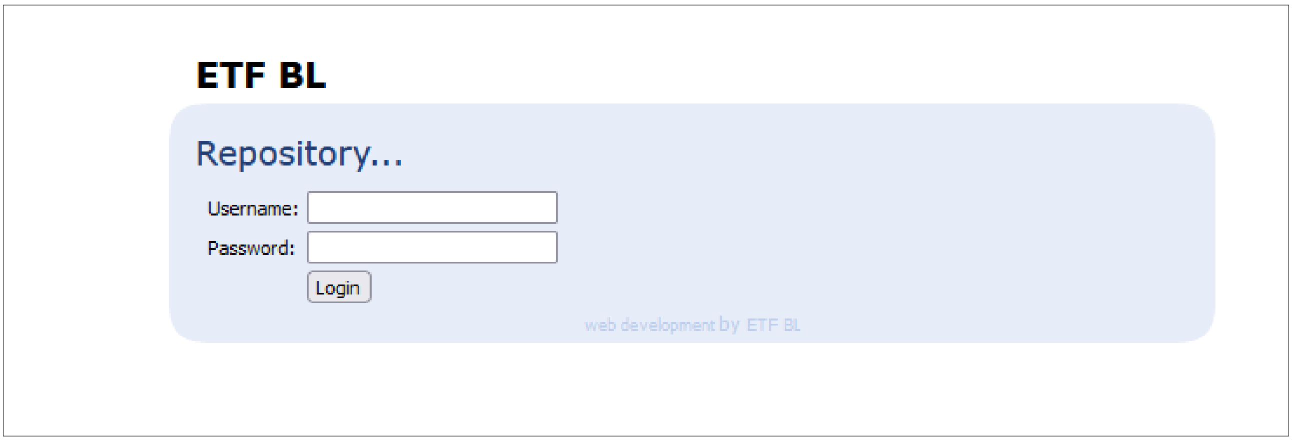
TEST DATA



test-data
etf_user.sql

```
INSERT INTO etf.user(username, password, user_group) VALUES('marko', 'Marko123', 0);
INSERT INTO etf.user(username, password, user_group) VALUES('janko', 'Janko123', 1);
```

ИЗГЛЕД

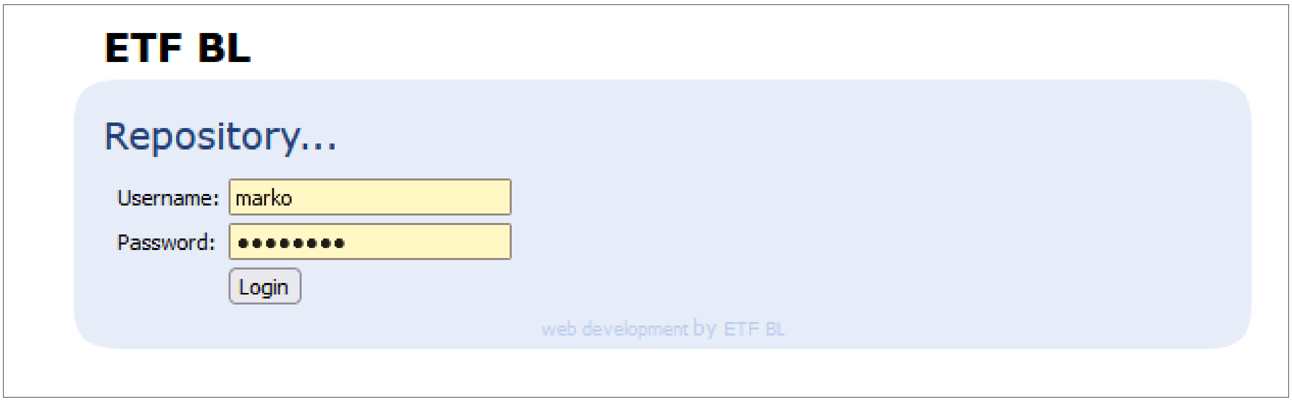


ETF BL

Repository...

Username:

Password:

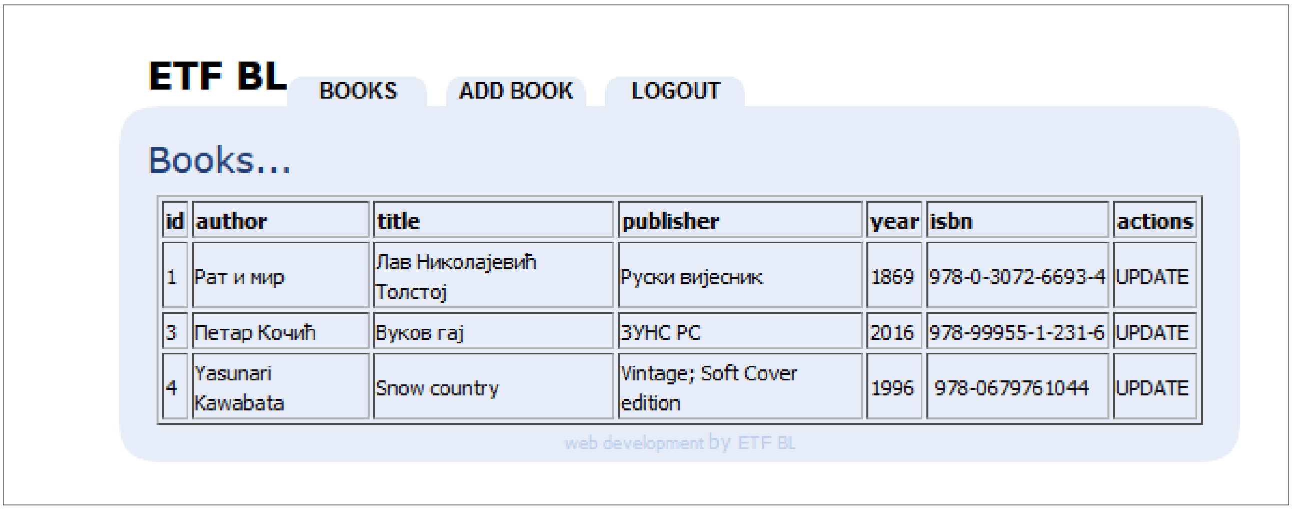


ETF BL

Repository...

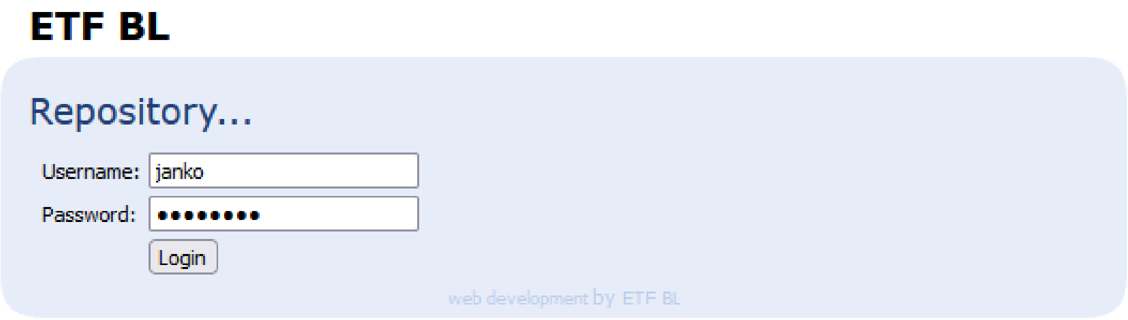
Username: marko

Password: ••••••••



Books...

| id | author | title | publisher | year | isbn | actions |
|---|---|---|---|---|---|---|
| 1 | Рат и мир | Лав Николајевић Толстој | Руски вијесник | 1869 | 978-0-3072-6693-4 | UPDATE |
| 3 | Петар Кочић | Вуков гај | ЗУНС РС | 2016 | 978-99955-1-231-6 | UPDATE |
| 4 | Yasunari Kawabata | Snow country | Vintage; Soft Cover edition | 1996 | 978-0679761044 | UPDATE |



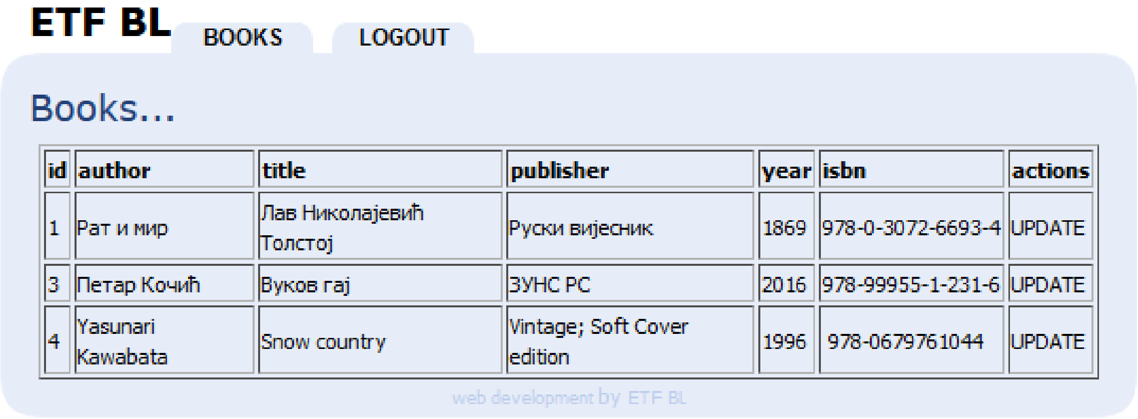
ETF BL

Repository...

Username: janko

Password: ●●●●●●●●●

web development by ETF BL

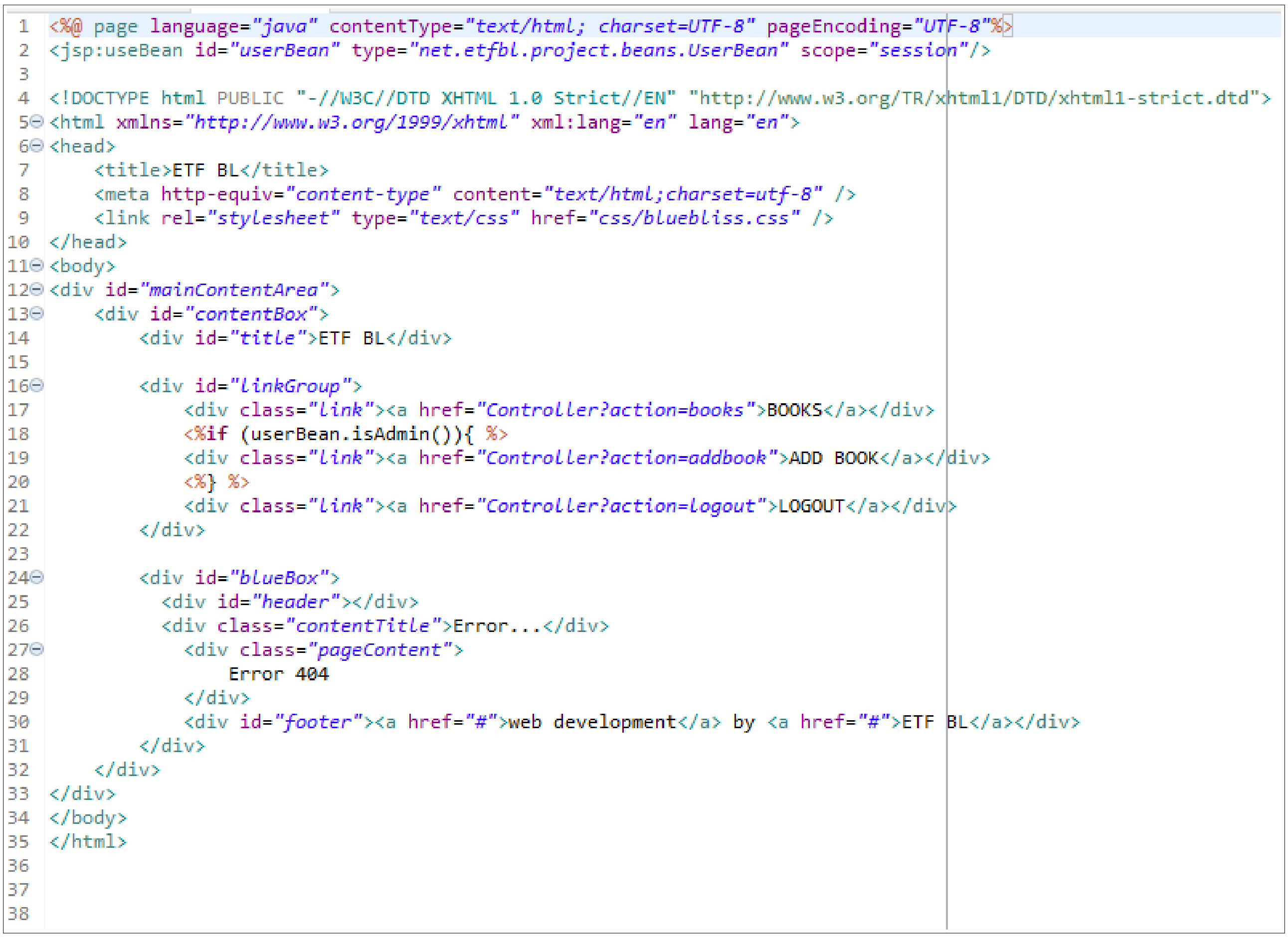


Books...

| id | author | title | publisher | year | isbn | actions |
|---|---|---|---|---|---|---|
| 1 | Рат и мир | Лав Николајевић Толстој | Руски вијесник | 1869 | 978-0-3072-6693-4 | UPDATE |
| 3 | Петар Кочић | Вуков гај | ЗУНС РС | 2016 | 978-99955-1-231-6 | UPDATE |
| 4 | Yasunari Kawabata | Snow country | Vintage; Soft Cover edition | 1996 | 978-0679761044 | UPDATE |

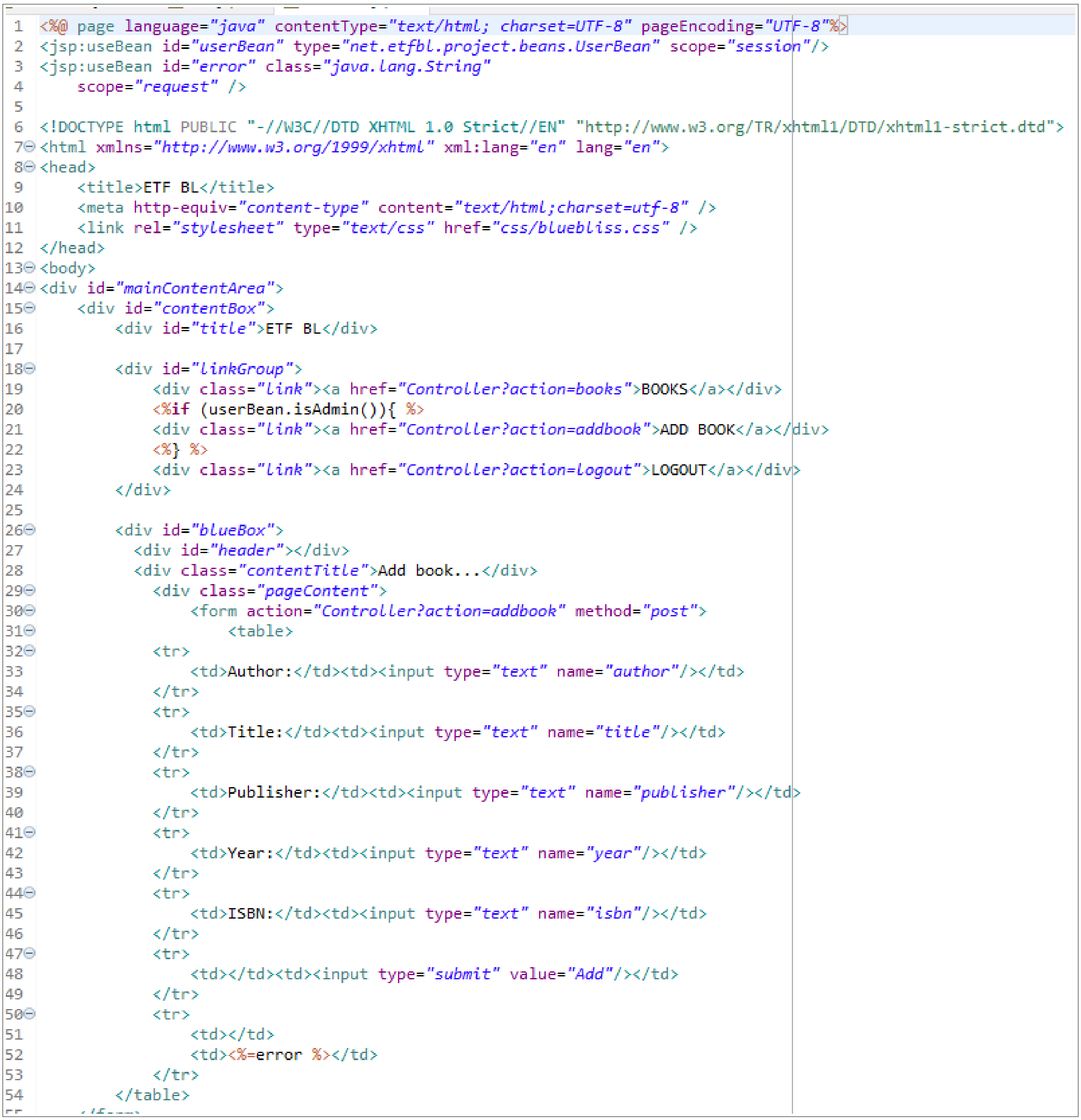
web development by ETF BL

404.JSP

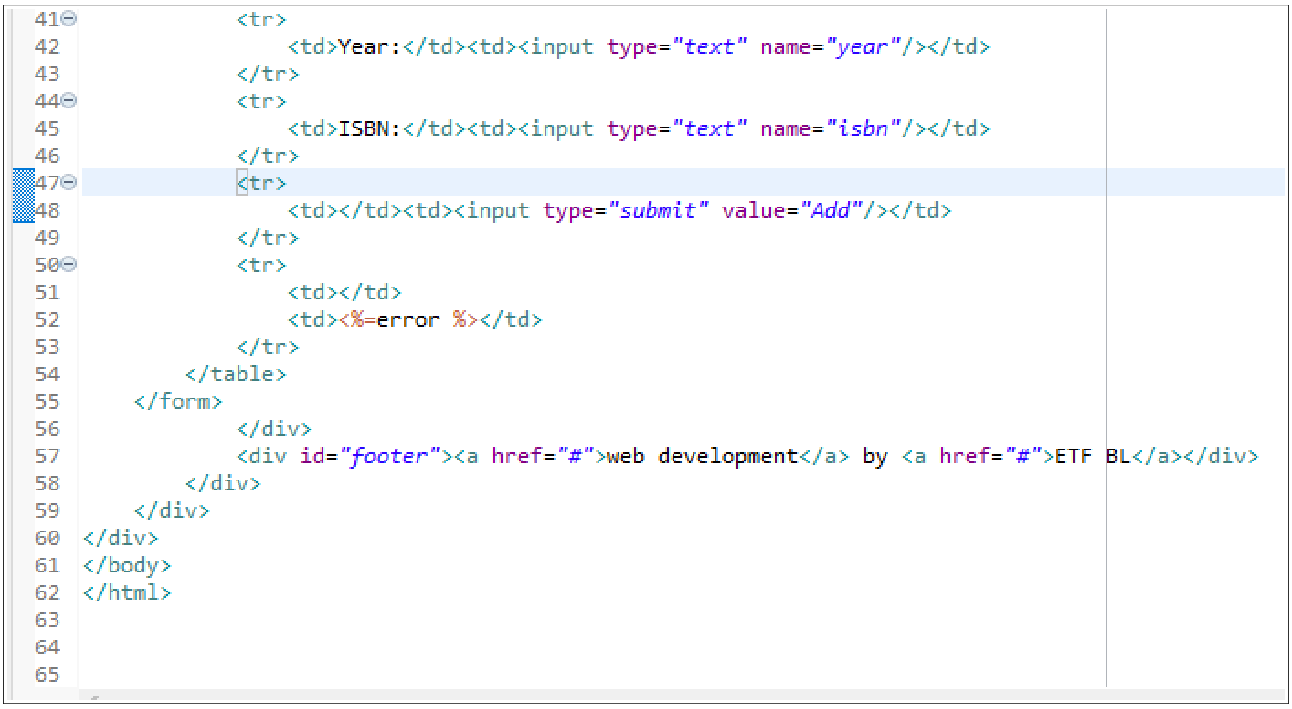


```
<%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<jsp:useBean id="userBean" type="net.etfbl.project.beans.UserBean" scope="session"/>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
    <title>ETF BL</title>
    <meta http-equiv="content-type" content="text/html;charset=utf-8" />
    <link rel="stylesheet" type="text/css" href="css/bluebliss.css" />
</head>
<body>
<div id="mainContentArea">
    <div id="contentBox">
        <div id="title">ETF BL</div>

        <div id="linkGroup">
            <div class="link"><a href="Controller?action=books">BOOKS</a></div>
            <%if (userBean.isAdmin()){ %>
            <div class="link"><a href="Controller?action=addbook">ADD BOOK</a></div>
            <%} %>
            <div class="link"><a href="Controller?action=logout">LOGOUT</a></div>
        </div>

        <div id="blueBox">
          <div id="header"></div>
          <div class="contentTitle">Error...</div>
            <div class="pageContent">
                Error 404
            </div>
            <div id="footer"><a href="#">web development</a> by <a href="#">ETF BL</a></div>
        </div>
    </div>
</div>
</body>
</html>
```

# ADDBOOK.JSP

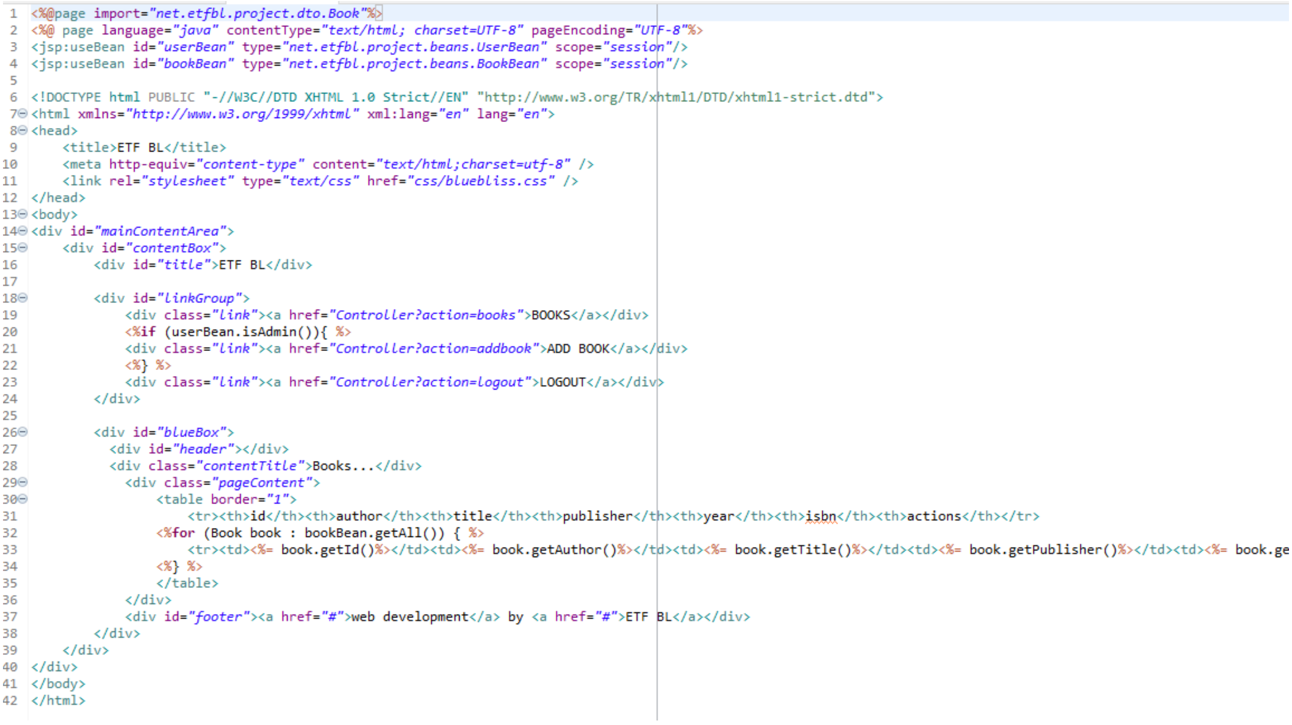


```
<%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<jsp:useBean id="userBean" type="net.etfbl.project.beans.UserBean" scope="session"/>
<jsp:useBean id="error" class="java.lang.String"
    scope="request" />

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
    <title>ETF BL</title>
    <meta http-equiv="content-type" content="text/html;charset=utf-8" />
    <link rel="stylesheet" type="text/css" href="css/bluebliss.css" />
</head>
<body>
<div id="mainContentArea">
    <div id="contentBox">
        <div id="title">ETF BL</div>

        <div id="linkGroup">
            <div class="link"><a href="Controller?action=books">BOOKS</a></div>
            <%if (userBean.isAdmin()){ %>
            <div class="link"><a href="Controller?action=addbook">ADD BOOK</a></div>
            <%} %>
            <div class="link"><a href="Controller?action=logout">LOGOUT</a></div>
        </div>

        <div id="blueBox">
          <div id="header"></div>
          <div class="contentTitle">Add book...</div>
            <div class="pageContent">
                <form action="Controller?action=addbook" method="post">
                    <table>
            <tr>
                <td>Author:</td><td><input type="text" name="author"/></td>
            </tr>
            <tr>
                <td>Title:</td><td><input type="text" name="title"/></td>
            </tr>
            <tr>
                <td>Publisher:</td><td><input type="text" name="publisher"/></td>
            </tr>
            <tr>
                <td>Year:</td><td><input type="text" name="year"/></td>
            </tr>
            <tr>
                <td>ISBN:</td><td><input type="text" name="isbn"/></td>
            </tr>
            <tr>
                <td></td><td><input type="submit" value="Add"/></td>
            </tr>
            <tr>
                <td></td>
                <td><%=error %></td>
            </tr>
        </table>
    </form>
```

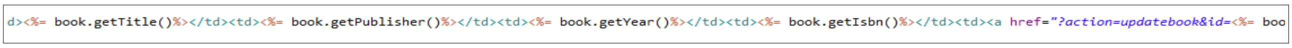


```
            <tr>
                <td>Year:</td><td><input type="text" name="year"/></td>
            </tr>
            <tr>
                <td>ISBN:</td><td><input type="text" name="isbn"/></td>
            </tr>
            <tr>
                <td></td><td><input type="submit" value="Add"/></td>
            </tr>
            <tr>
                <td></td>
                <td><%=error %></td>
            </tr>
        </table>
    </form>
            </div>
            <div id="footer"><a href="#">web development</a> by <a href="#">ETF BL</a></div>
        </div>
    </div>
</div>
</body>
</html>
```

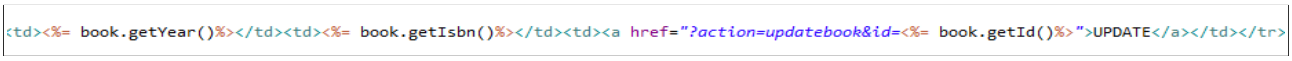
BOOKS.JSP



```
<%@page import="net.etfbl.project.dto.Book"%>
<%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<jsp:useBean id="userBean" type="net.etfbl.project.beans.UserBean" scope="session"/>
<jsp:useBean id="bookBean" type="net.etfbl.project.beans.BookBean" scope="session"/>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
    <title>ETF BL</title>
    <meta http-equiv="content-type" content="text/html;charset=utf-8" />
    <link rel="stylesheet" type="text/css" href="css/bluebliss.css" />
</head>
<body>
<div id="mainContentArea">
    <div id="contentBox">
        <div id="title">ETF BL</div>

        <div id="linkGroup">
            <div class="link"><a href="Controller?action=books">BOOKS</a></div>
            <%if (userBean.isAdmin()){ %>
            <div class="link"><a href="Controller?action=addbook">ADD BOOK</a></div>
            <%} %>
            <div class="link"><a href="Controller?action=logout">LOGOUT</a></div>
        </div>

        <div id="blueBox">
          <div id="header"></div>
          <div class="contentTitle">Books...</div>
            <div class="pageContent">
                <table border="1">
                    <tr><th>id</th><th>author</th><th>title</th><th>publisher</th><th>year</th><th>isbn</th><th>actions</th></tr>
                <%for (Book book : bookBean.getAll()) { %>
                    <tr><td><%= book.getId()%></td><td><%= book.getAuthor()%></td><td><%= book.getTitle()%></td><td><%= book.getPublisher()%></td><td><%= book.ge
                <%} %>
                </table>
            </div>
            <div id="footer"><a href="#">web development</a> by <a href="#">ETF BL</a></div>
        </div>
    </div>
</div>
</body>
</html>
```

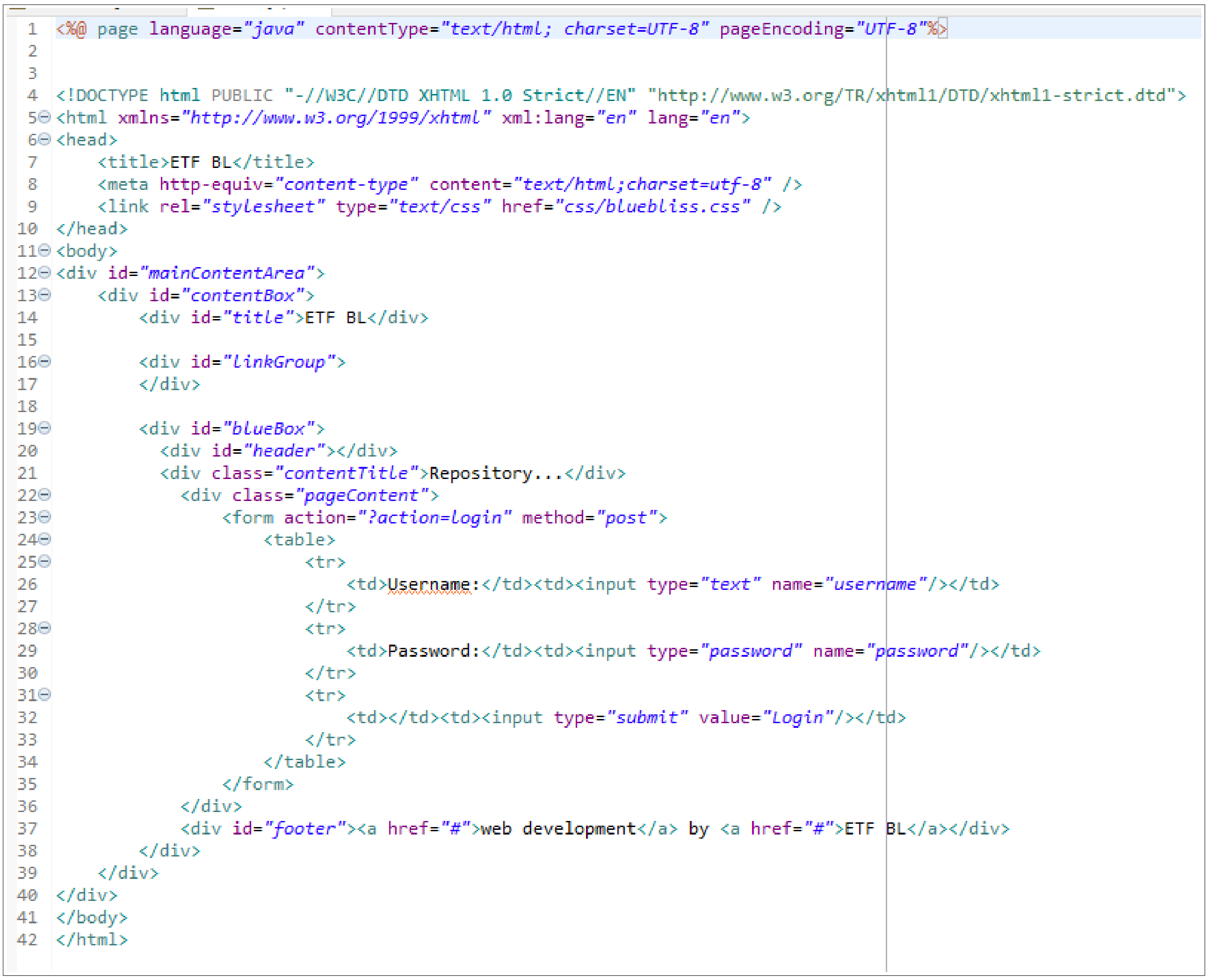


```
d><%= book.getTitle()%></td><td><%= book.getPublisher()%></td><td><%= book.getYear()%></td><td><%= book.getIsbn()%></td><td><a href="?action=updatebook&id=<%= boo
```



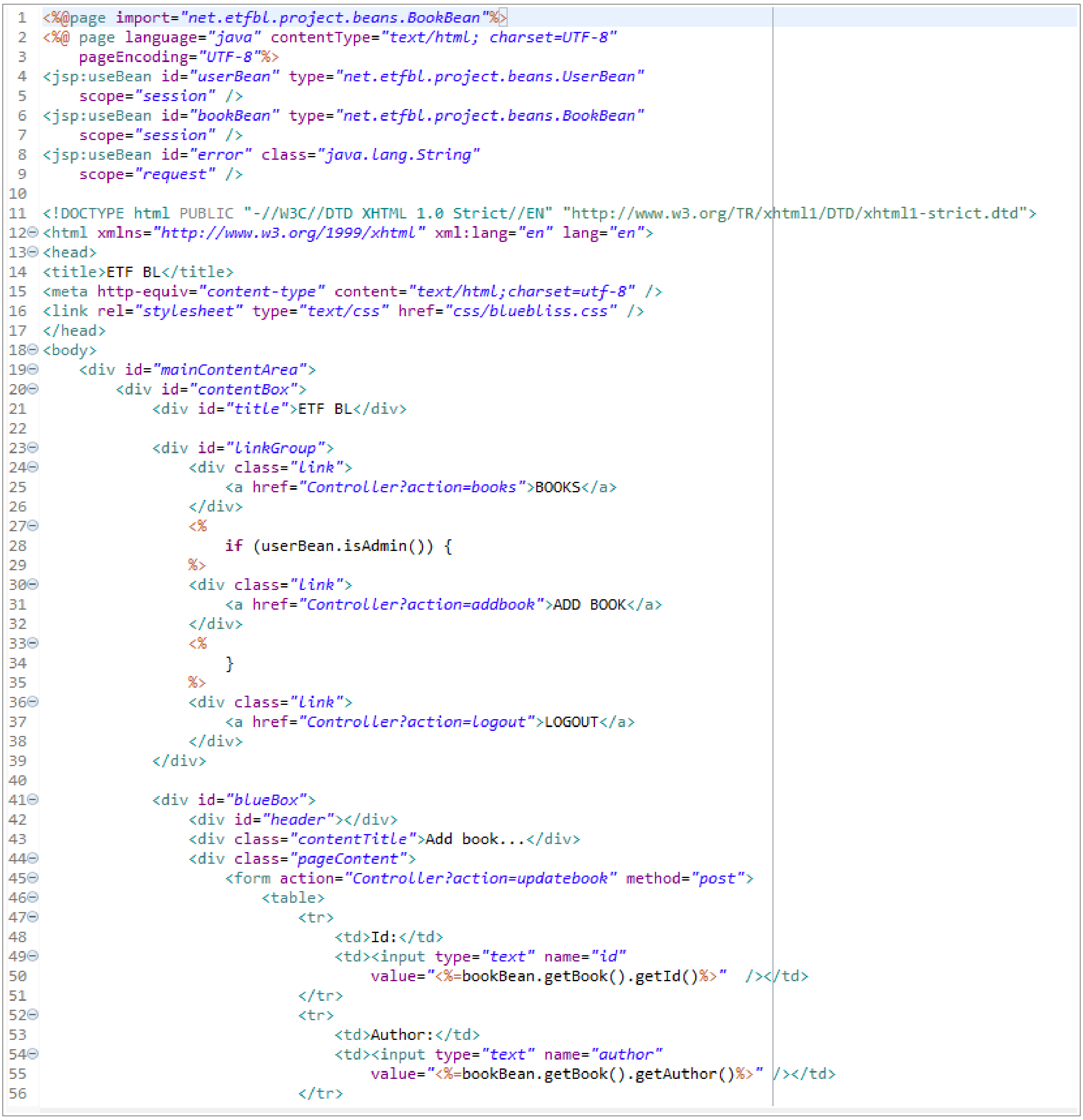
```
<td><%= book.getYear()%></td><td><%= book.getIsbn()%></td><td><a href="?action=updatebook&id=<%= book.getId()%>">UPDATE</a></td></tr>
```

# INDEX.JSP

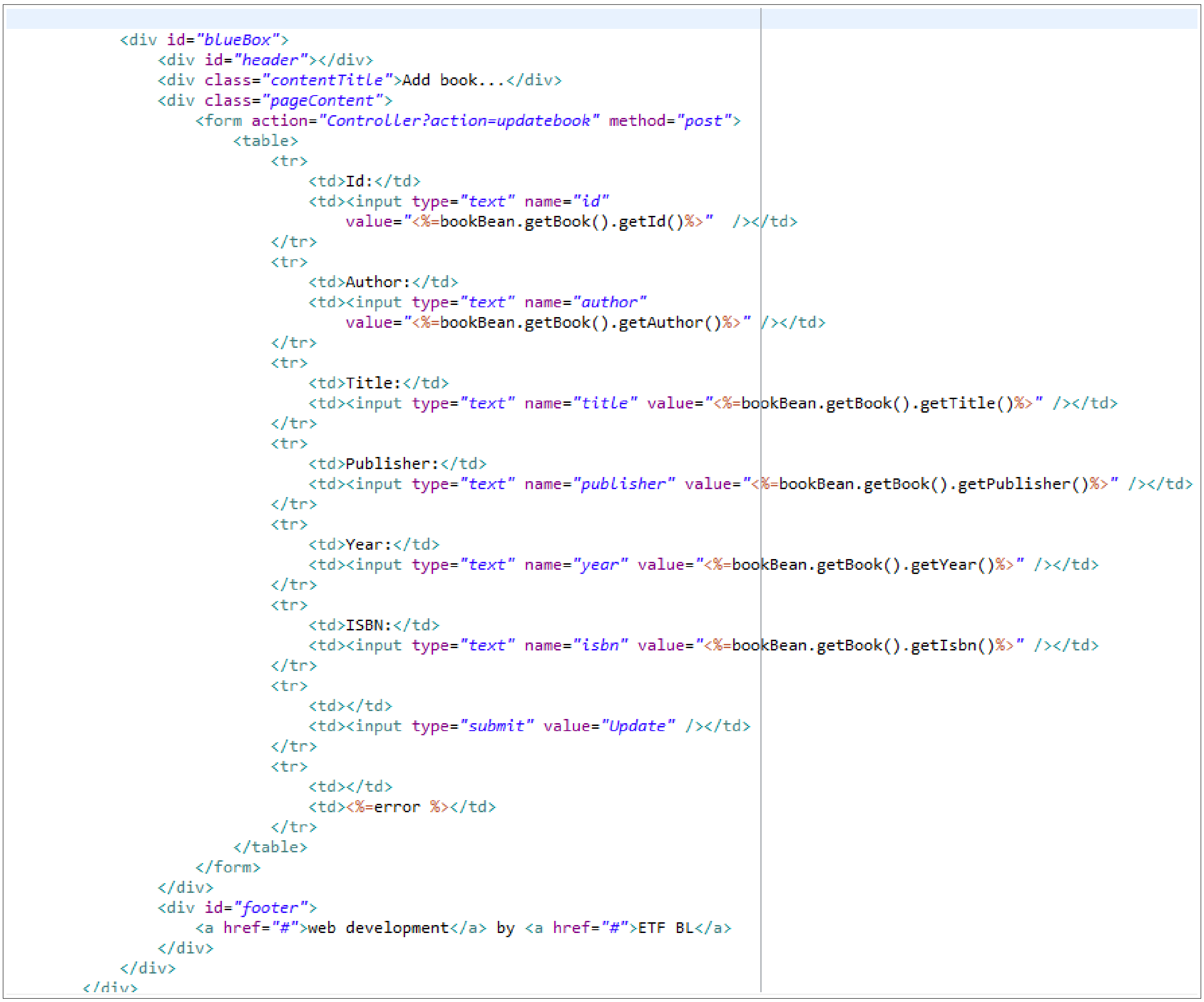


```
<%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>


<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
    <title>ETF BL</title>
    <meta http-equiv="content-type" content="text/html;charset=utf-8" />
    <link rel="stylesheet" type="text/css" href="css/bluebliss.css" />
</head>
<body>
<div id="mainContentArea">
    <div id="contentBox">
        <div id="title">ETF BL</div>

        <div id="linkGroup">
        </div>

        <div id="blueBox">
          <div id="header"></div>
          <div class="contentTitle">Repository...</div>
            <div class="pageContent">
                <form action="?action=login" method="post">
                    <table>
                        <tr>
                            <td>Username:</td><td><input type="text" name="username"/></td>
                        </tr>
                        <tr>
                            <td>Password:</td><td><input type="password" name="password"/></td>
                        </tr>
                        <tr>
                            <td></td><td><input type="submit" value="Login"/></td>
                        </tr>
                    </table>
                </form>
            </div>
            <div id="footer"><a href="#">web development</a> by <a href="#">ETF BL</a></div>
        </div>
    </div>
</div>
</body>
</html>
```

UPDATE_BOOK.JSP



```
<%@page import="net.etfbl.project.beans.BookBean"%>
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<jsp:useBean id="userBean" type="net.etfbl.project.beans.UserBean"
    scope="session" />
<jsp:useBean id="bookBean" type="net.etfbl.project.beans.BookBean"
    scope="session" />
<jsp:useBean id="error" class="java.lang.String"
    scope="request" />

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
<title>ETF BL</title>
<meta http-equiv="content-type" content="text/html;charset=utf-8" />
<link rel="stylesheet" type="text/css" href="css/bluebliss.css" />
</head>
<body>
    <div id="mainContentArea">
        <div id="contentBox">
            <div id="title">ETF BL</div>

            <div id="linkGroup">
                <div class="link">
                    <a href="Controller?action=books">BOOKS</a>
                </div>
                <%
                    if (userBean.isAdmin()) {
                %>
                <div class="link">
                    <a href="Controller?action=addbook">ADD BOOK</a>
                </div>
                <%
                    }
                %>
                <div class="link">
                    <a href="Controller?action=logout">LOGOUT</a>
                </div>
            </div>

            <div id="blueBox">
                <div id="header"></div>
                <div class="contentTitle">Add book...</div>
                <div class="pageContent">
                    <form action="Controller?action=updatebook" method="post">
                        <table>
                            <tr>
                                <td>Id:</td>
                                <td><input type="text" name="id"
                                    value="<%=bookBean.getBook().getId()%>"  /></td>
                            </tr>
                            <tr>
                                <td>Author:</td>
                                <td><input type="text" name="author"
                                    value="<%=bookBean.getBook().getAuthor()%>" /></td>
                            </tr>
```



```
    <div id="blueBox">
        <div id="header"></div>
        <div class="contentTitle">Add book...</div>
        <div class="pageContent">
            <form action="Controller?action=updatebook" method="post">
                <table>
                    <tr>
                        <td>Id:</td>
                        <td><input type="text" name="id"
                            value="<%=bookBean.getBook().getId()%>"  /></td>
                    </tr>
                    <tr>
                        <td>Author:</td>
                        <td><input type="text" name="author"
                            value="<%=bookBean.getBook().getAuthor()%>" /></td>
                    </tr>
                    <tr>
                        <td>Title:</td>
                        <td><input type="text" name="title" value="<%=bookBean.getBook().getTitle()%>" /></td>
                    </tr>
                    <tr>
                        <td>Publisher:</td>
                        <td><input type="text" name="publisher" value="<%=bookBean.getBook().getPublisher()%>" /></td>
                    </tr>
                    <tr>
                        <td>Year:</td>
                        <td><input type="text" name="year" value="<%=bookBean.getBook().getYear()%>" /></td>
                    </tr>
                    <tr>
                        <td>ISBN:</td>
                        <td><input type="text" name="isbn" value="<%=bookBean.getBook().getIsbn()%>" /></td>
                    </tr>
                    <tr>
                        <td></td>
                        <td><input type="submit" value="Update" /></td>
                    </tr>
                    <tr>
                        <td></td>
                        <td><%=error %></td>
                    </tr>
                </table>
            </form>
        </div>
        <div id="footer">
            <a href="#">web development</a> by <a href="#">ETF BL</a>
        </div>
    </div>
</div>
```