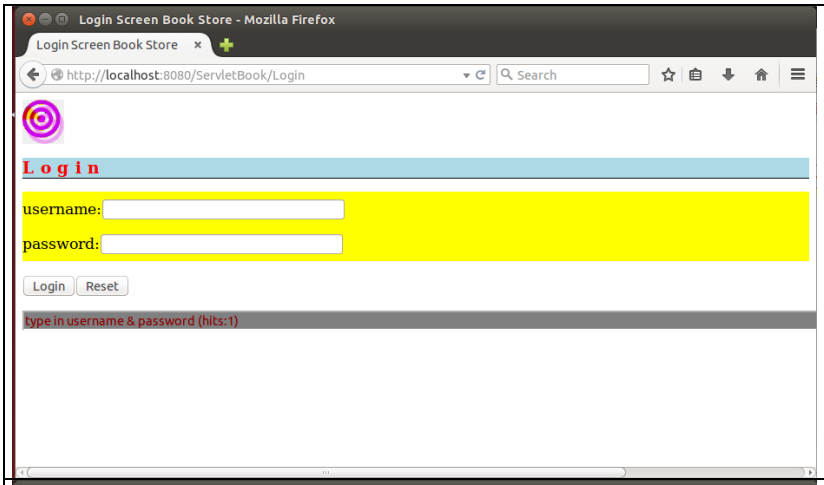
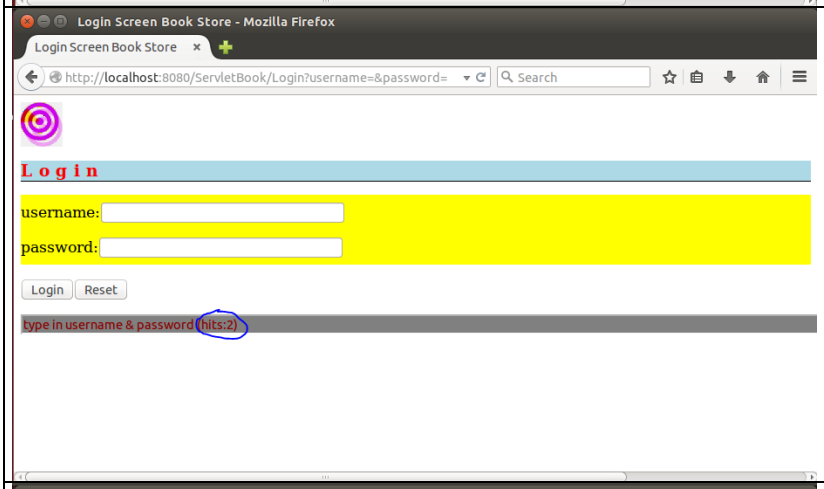
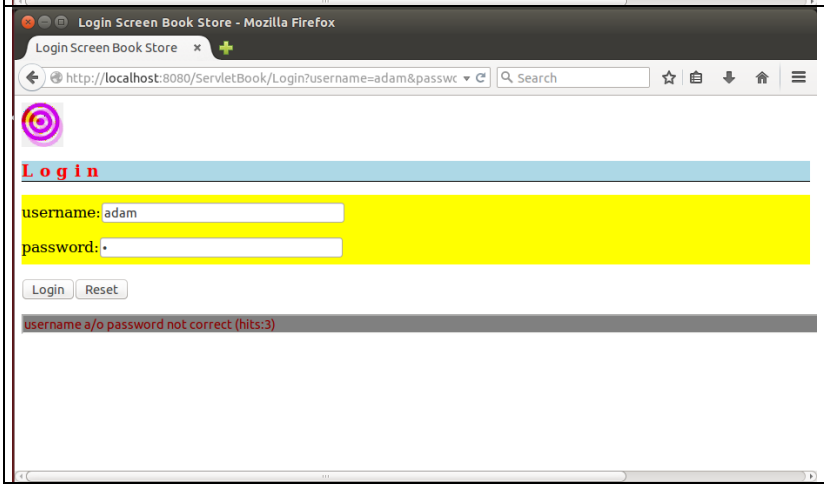
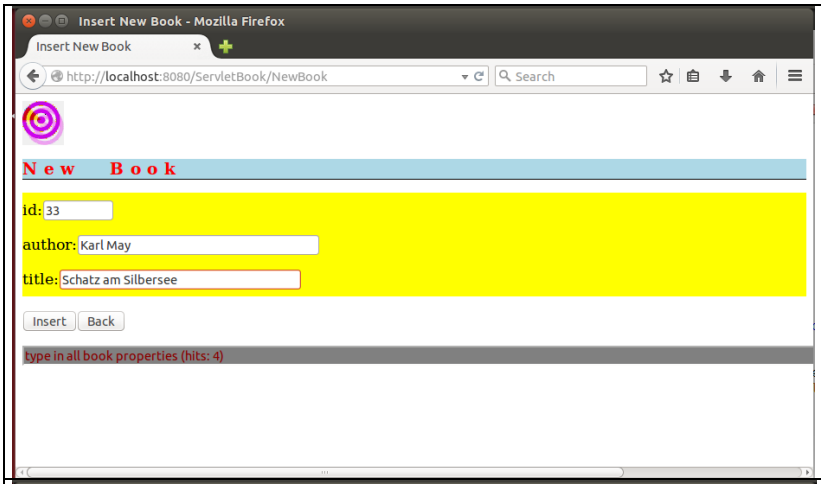
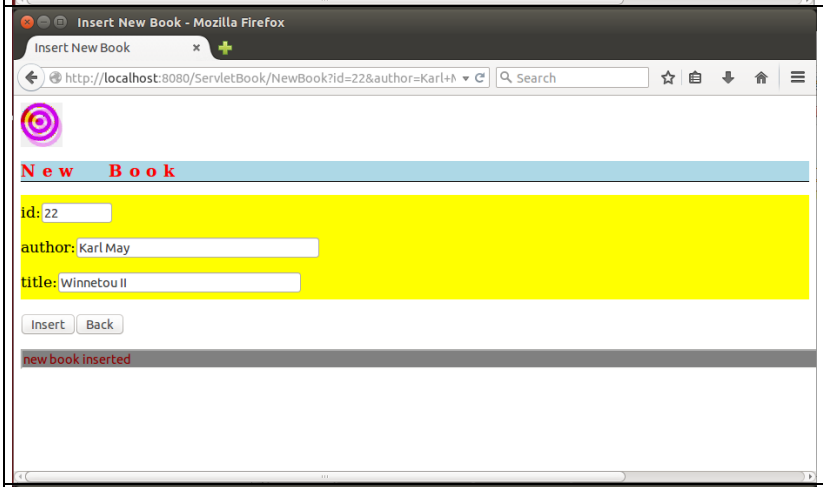
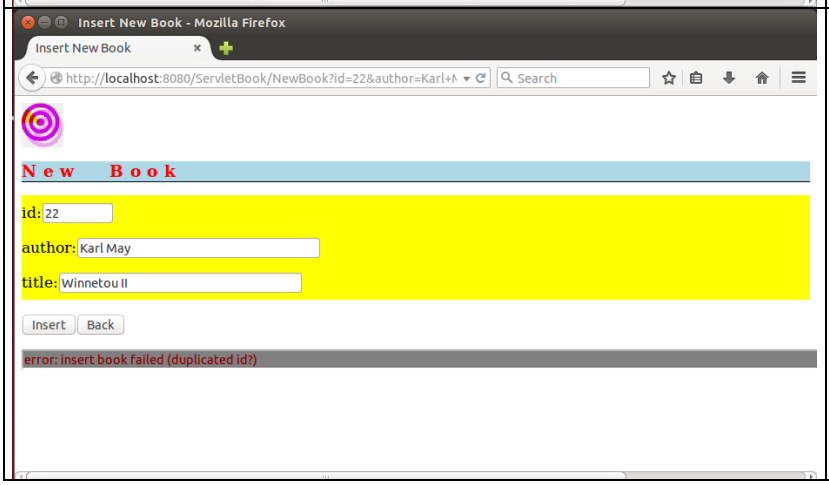
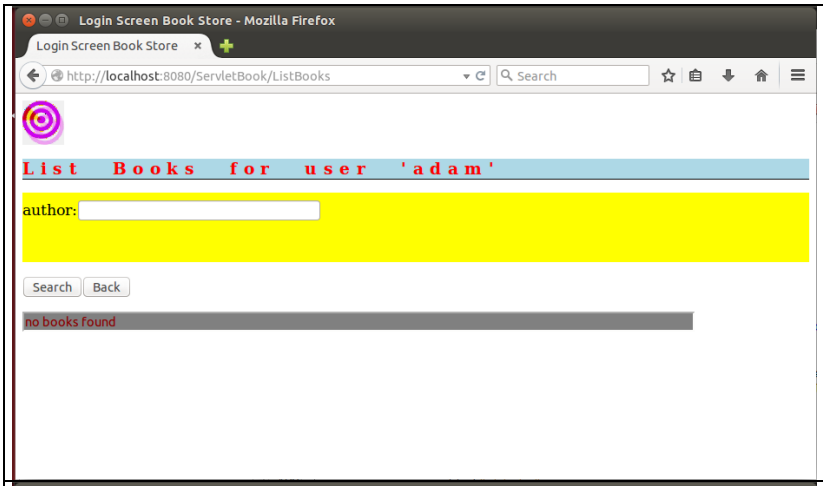
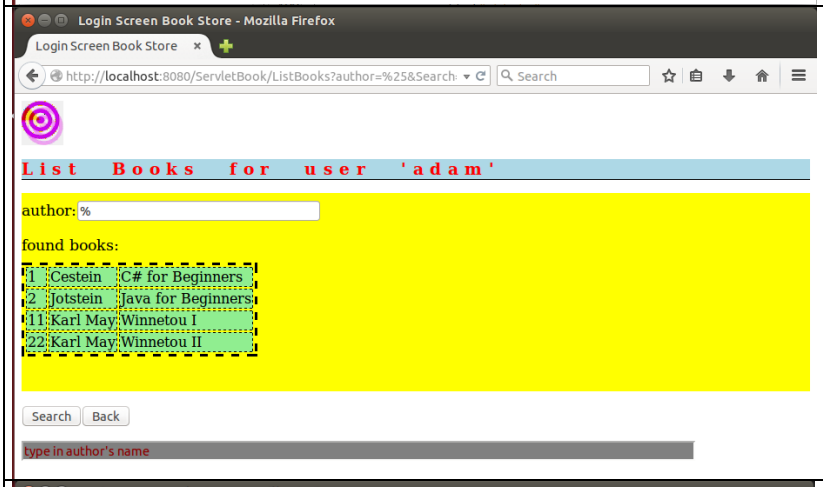
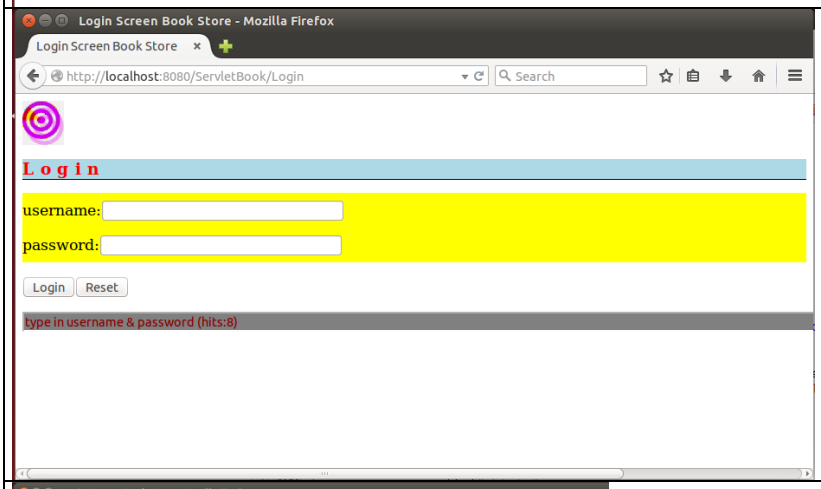
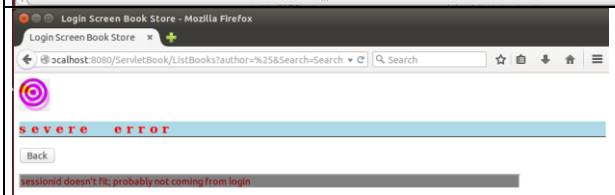
1. User's View

	servlet „Login" is started hints: • flashing design with the help of css • have a look to the url
	no input done, but login pressed hints: • have a look to the url and • message line
	incorrect login hints: o at the moment login-data in class „Database" may be stored as collection o watch URL and message line

	login as “admin”; entering new books
	if book ok
	if book not ok

	login as ordinary user
	searching for author (LIKE – wildcards)
	back again
	evil user: hinder evil users from accessing the bookshop without visiting the login-page (session)

Hints

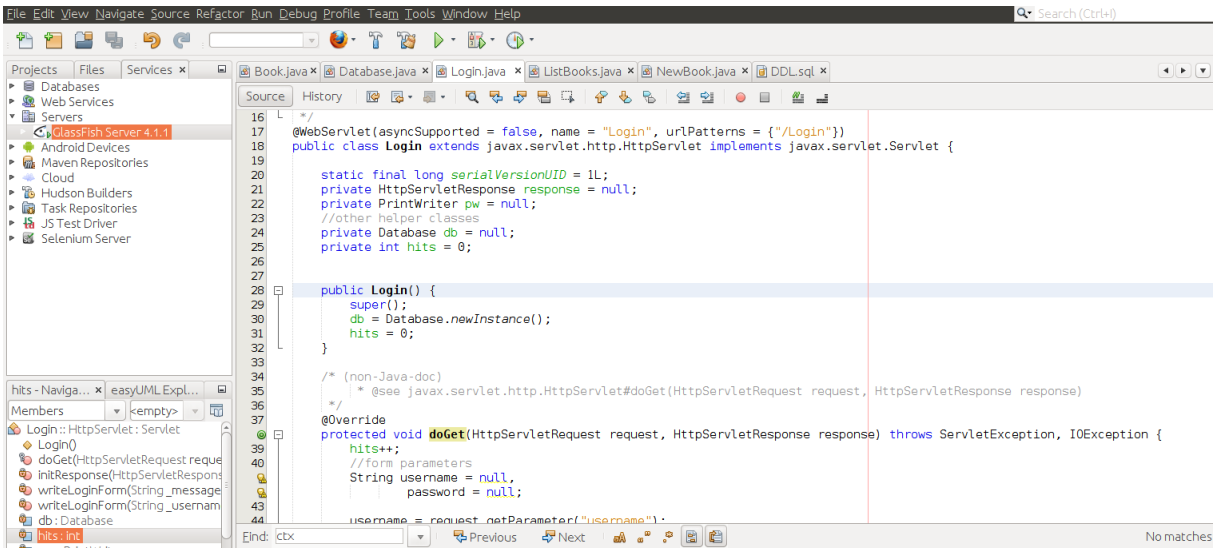
- oracle – tables: ***users, books***

```
DROP TABLE orders CASCADE CONSTRAINTS;
DROP TABLE users CASCADE CONSTRAINTS;
DROP TABLE books CASCADE CONSTRAINTS;
CREATE TABLE users (
      username VARCHAR2(20),
      password VARCHAR2(20),
      CONSTRAINTS pk_user PRIMARY KEY (username)
);
CREATE TABLE books (
      id INTEGER,
      title VARCHAR2(50),
      author VARCHAR2(20),
      CONSTRAINTS pk_books PRIMARY KEY (id)
);
INSERT INTO users VALUES('admin','secret');
INSERT INTO users VALUES('adam','adam');
INSERT INTO users VALUES('eva','eva');
```

- test multi-user ability and think about possible problems
- url-redirect:

```
String url = response.encodeRedirectURL(
request.getContextPath() + "/MenueAdmin");
response.sendRedirect(url);
```

Code-Snippet - 1



```
/**
 */
@WebServlet(asyncSupported = false, name = "Login", urlPatterns = {"/Login"})
public class Login extends javax.servlet.http.HttpServlet implements javax.servlet.Servlet {

    static final long serialVersionUID = 1L;
    private HttpServletResponse response = null;
    private PrintWriter pw = null;
    //other helper classes
    private Database db = null;
    private int hits = 0;


    public Login() {
        super();
        db = Database.newInstance();
        hits = 0;
    }

    /* (non-Java-doc)
     * @see javax.servlet.http.HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
     */
    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        hits++;
        //form parameters
        String username = null,
               password = null;

        username = request.getParameter("username");
```

Code-Snippet – 2

```
    public ListBooks() {
        super();
        db = Database.newInstance();
        hits = 0;
    }

    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        try {
            verifySession(request);
            hits++;
            //if page already shown and SEARCH pressed
            if (request.getParameter("Search") != null) {
                author = request.getParameter("author");
                initResponse(response);
                writeListBookForm(username, author, "type in author's name (hits: " + hits + ")");
            } //BACK was pressed
            else if (request.getParameter("Back") != null) {
                String url = response.encodeRedirectURL(request.getContextPath() + "/Login");
                response.sendRedirect(url);
            } //if coming from login
            else {
                username = request.getSession().getAttribute("username").toString();
```

```
private void verifySession(HttpServletRequest request) throws Exception {
    HttpSession session = request.getSession();
    if ((session.getAttribute("sessionId") == null)
            || !session.getId().equals(session.getAttribute("sessionId").toString())) {
        throw new Exception("sessionid doesn't fit; probably not coming from login");
    }
}
```