

CMPBD 497C Assignment 6 Due November 8

In this assignment we will write the functionality of a generic shopping basket. For this assignment we will use mybooks database that we created in the previous assignment (#5). To track clients on our web site the best way is to use either Cookies or HttpSession objects. But for this assignment we will use a table in mybooks database for tracking and also for the shopping cart.

The extra tables required for this assignment are given below:

Table clients:

```
CREATE TABLE clients (userName CHAR(30),
                      street CHAR(20),
                      state CHAR(2),
                      zip CHAR(5),
                      PRIMARY KEY(userName));
```

In the clients table the column named userName contains the client's logging username.

Table cart:

```
CREATE TABLE cart (userName CHAR(30),
                  book_isbn CHAR(20) NOT NULL,
                  quantity int,
                  total real,
                  FOREIGN KEY(book_isbn) REFERENCES inventory,
                  FOREIGN KEY(userName) REFERENCES clients,
                  PRIMARY KEY(userName, book_isbn));
```

The cart table references two tables: the inventory and the clients. The book_isbn and id columns together form the primary key for this table. An entry is inserted in this table when a client selects a book from the list. The quantity column saves the number of books the client decides to buy. Most of the times this will be 1, but a client may want to buy more than one copy of the same book. The total column contains the total amount the client will pay.

Table creditCards:

```
CREATE TABLE creditCards (cardNumber CHAR(7) PRIMARY KEY);
```

```
insert into creditCards values('123456');
insert into creditCards values('223456');
insert into creditCards values('323456');
insert into creditCards values('423456');
insert into creditCards values('523458');
insert into creditCards values('623459');
insert into creditCards values('723453');
insert into creditCards values('823457');
```

This table contains credit card numbers. To simplify, we assume that the credit card numbers 6 digits. This table will be consulted before the shopping is finalized. If credit card number is found in the table, the transaction will be committed, otherwise a rollback must be done and all information about the client for this transaction must be deleted from the tables.

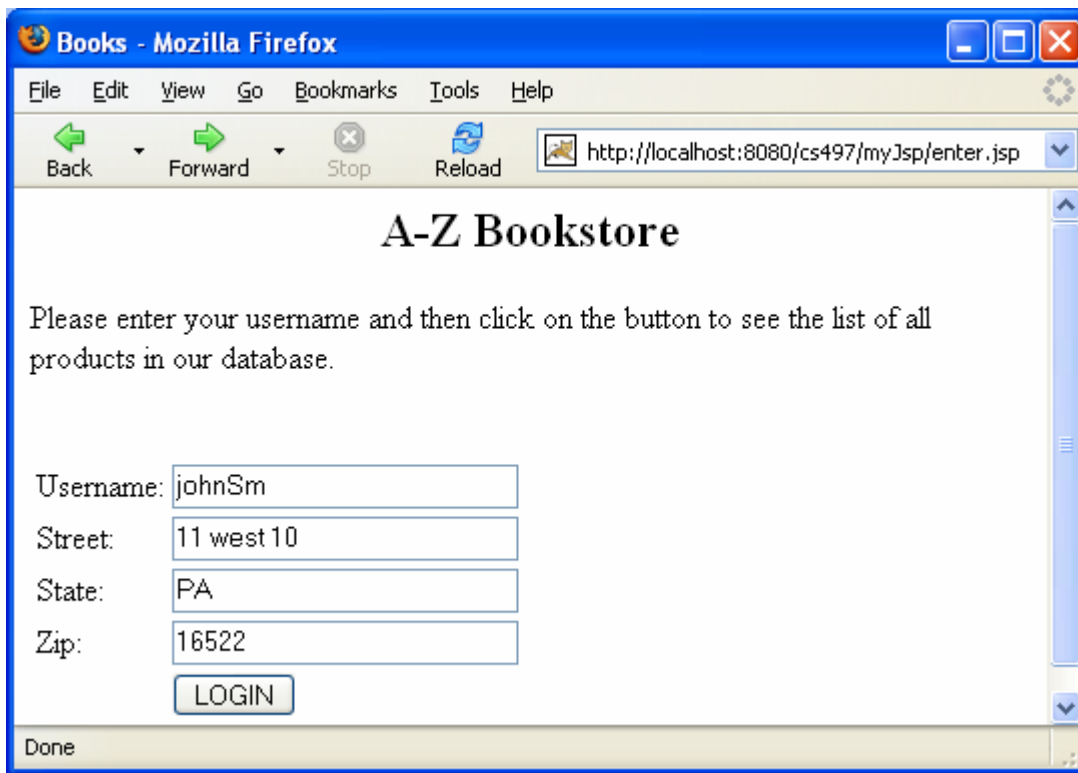
The JSP pages required for this assignment are listed below:

1. enter.jsp : This page is the main entry point to our web site. To simplify, we will assume that every client will enter a different username. But if you want to make it like a real application, you should not allow clients to enter the same username more than one time. Since userName column is the primary key column, MySQL will not allow duplicates. But the client will not see

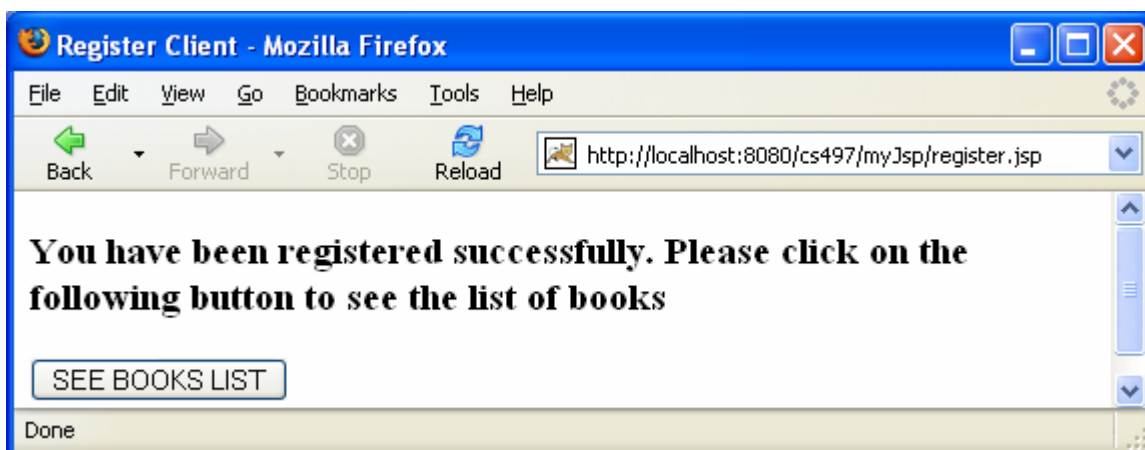
this rejection unless you send some information to the web browser. The simplest solution would be checking the clients table and see whether the username already exists. If yes, then request a new username.

In addition to username, we ask for street, state, and zip from client.

After providing a username, the client will be forwarded to another JSP page named register.jsp.



2. register.jsp: This page reads the input data (username, street, state, zip) and inserts a new row into the clients table:



If you check the clients table you should see an entry for this client:

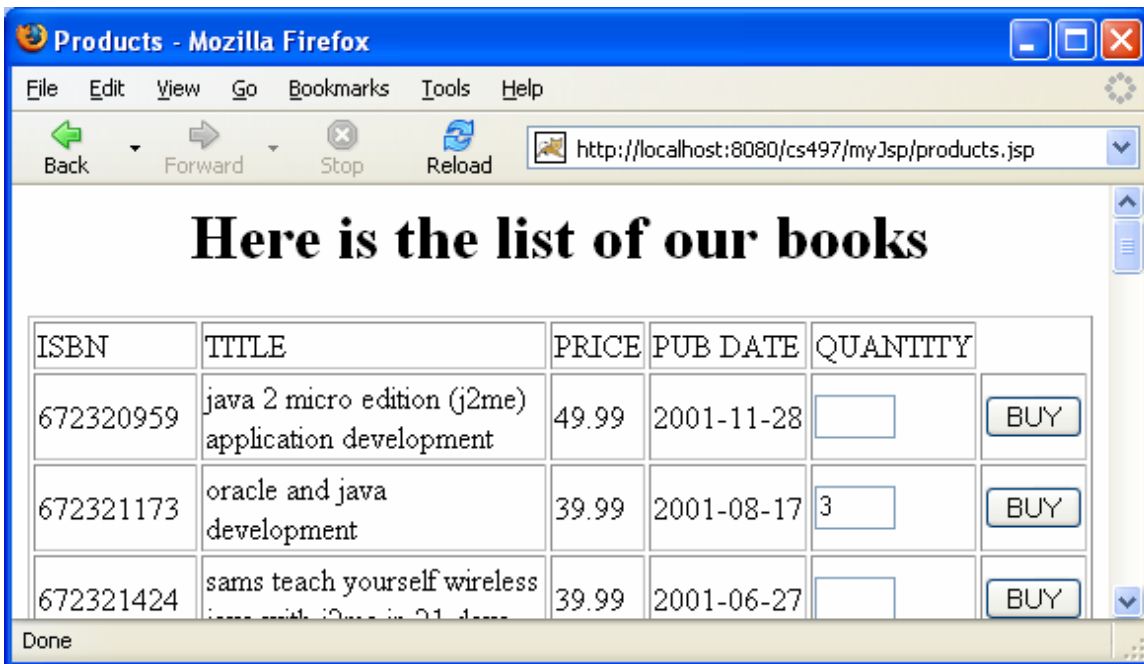
```

mysql> select * from clients;
+-----+-----+-----+-----+
| userName | street      | state | zip   |
+-----+-----+-----+-----+
| johnSm   | 11 west 10 | PA    | 16522 |
+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql>

```

3. products.jsp: This page displays the list of all books with their information except the description (since this column contains many lines of text and we don't want to display it). Each listed product should have textfield and a button next to it. The textfield will be used to enter the number of the selected book to buy. The button is used to activate another JSP named cart.jsp.



4. cart.jsp: This page will read the input data sent by the products.jsp. Then, it inserts an entry into the cart table. Finally it sends client to finalize.jsp.

5. finalize.jsp: this JSP will ask the credit card number to be charged. It then searches the table creditCard to verify it. If not found, it gives an error message and rollbacks the transaction.

NOTE: When calling stored procedures or stored functions, if you are getting the error

Data truncation: Data truncated for column ...

you should add jdbcCompliantTruncation=false to your database URL. For example,

```

Connection con = DriverManager.getConnection
("jdbc:mysql://localhost:3306/mybooks?user=root
&password=secret&jdbcCompliantTruncation=false");

```