Assignment1

Content

- Familiar with MS Access
- Establish a university database
  - university.mdb
- Load data into the database
- Make a group of random tuples for student table and let student has more than 5000 tuples

Schema and Sample Data

- The schema of the database is provided below (keys are in bold, field types are omitted):

  student(**sid**, sname, sex, age, year, gpa)
  dept(**dname**, numphds)
  prof(**pname**, dname)
  course(**cno**, cname, **dname**)
  major(**dname**, **sid**)
  section(**dname**, **cno**, **sectno**, pname)
  enroll(**sid**, grade, **dname**, **cno**, **sectno**)

- Before you start writing SQL, it is a good idea to take a look at the database and familiarize yourself with its contents.

Data Files

- Download the following data files from http://cse.seu.edu.cn/people/lzxu/resource/courses/university.rar
  - course.txt
  - dept.txt
  - enroll.txt
  - major.txt
  - prof.txt
  - section.txt
  - student.txt

Submission

- File name format:
  Sno_A1.zip
  including:
  report_A1.doc
  university_A1.mdb

Assignment2

Content

- SQL queries on university database
- Write SQL queries that answer the questions below (one query per question) and run them on the Microsoft ACCESS Database System using its SQL interpreter. The query answers must not contain duplicates, but you should use the SQL keyword **distinct** only when necessary.
- The SQL interpreter in ACCESS is not quite the same as the one described in the textbook. If the query you write is not accepted by ACCESS (usually it gives you some strange errors), try different ways until you get one that works with ACCESS. For this assignment, creation of temporary tables is not allowed, i.e., for each question you have to write exactly one SQL statement.

Questions

1. Print the names of professors who work in departments that have fewer than 50 PhD students.
2. Print the name(s) of student(s) with the lowest gpa.
3. For each Computer Sciences class, print the cno, sectno, and the average gpa of the students enrolled in the class.
4. Print the course names, course numbers and section numbers of all classes with less than six students enrolled in them.
5. Print the name(s) and sid(s) of the student(s) enrolled in the most classes.
6. Print the names of departments that have one or more majors who are under 18 years old.
7. Print the names and majors of students who are taking one of the College Geometry courses. (Hint: You'll need to use the "like" predicate and the string matching character in your query.)
8. For those departments that have no majors taking a College Geometry course, print the department name and the number of PhD students in the department.
9. Print the names of students who are taking both a Computer Sciences course and a Mathematics course.
10. Print the age difference between the oldest and youngest Computer Sciences major(s).
11. For each department that has one or more majors with a GPA under 1.0, print the name of the department and the average GPA of its majors.
12. Print the ids, names, and GPAs of the students who are currently taking all of the Civil Engineering courses.

Demand

- This is an individual assignment – no group submissions are allowed. Hand in an ACCESS database that contains the answers to the twelve questions. The database should contain twelve queries, named as follows:

  Query1

  Query2

  …

  Query12
- Test the function of index with query related with student table
- Hand in a report which indicates your answers

Submission

- File name format:

  Sno_A2.zip

  including:

  report_A2.doc

  university_A2.mdb

Assignment3

Content

- Very simple application based on university database (without programming)
  - For example, a student grade management
  - Involving at least 3~4 tables in university database
  - Can design an application background by yourselves.
- Use the tools offered by Access directly to implement the interface, queries, report, etc.

Demands

- Show your completed application to supervisor
- Hand in a report about the assignment
  - Main contents
  - Database design
  - Function design
  - Interface design
  - Database query
- Hand in your .mdb database with the application
- Two students a group

Submission

- File name format:
  Sno1_Sno2_A3.zip
  including:
  report_A3.doc
  university_A3.mdb

Assignment4

Content

- Please refer to our course Web site directly.

Submission

- File name format:
  Sno_A4.zip
  including:
  report_A4.doc

Assignment5

Content

- Improved the former application (programming)
- Use ODBC or JDBC to access database
- Use Java/VC/VB/Delphi to program
- Improve functions which is hard to implement in the former exercise
  - Login / logout (general customer / administrator)
  - Associating between interface components
  - Mutual access to database
  - …

Demand

- Hand in a report about the assignment
  - Main contents
  - Database design
  - Function design
  - Software design & implementation
  - Database query
- Package your database, source code, executable file into a zip file SnoName_A5.zip.
- Two students a group

Submission

- File name format:
  Sno1_Sno2_A5.zip
  including:
  report_A5.doc
  university_A5.mdb
  source code
  executable file

Assignment6

Content

- Internet bookstore
  - Web based
  - Three tiered architecture
  - Client is IE
- Improved database
- Improved function

Client Functionality

This section describes how our user interface will work and how customers will view and be able to interact with our web site. A more detailed description of how different pages work internally will be discussed in following sections.

We start by going over the types of users on our system. There are two different classes:

1. Anonymous Users
   All users who initially connect to our website are in this class and maintain in it until they log in with an account. They can search for books online, view book information for specific books, add books to their shopping carts, modify books in their shopping cart, view help menus, create new user accounts, and login to the system.
2. Logged in Users
   Once users have logged in from a previously created account, they have all the rights and functionality of an anonymous user as well as the ability to proceed to the checkout. Once there, they can confirm and then place their order.

Client Functionality

We will now examine the experience that each class of user will have during a visit to our website.

The anonymous user can view 10 types of pages:

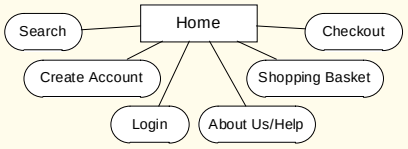
1. Home Page
2. Search Page
3. Book Information Page
4. Account Creation Page
5. Shopping Basket
6. Login Page
7. Help/Information Page
8. F.A.Q.
9. System Rules Page
10. Contact Page

The logged in user can view also view:

11. Checkout Page

Common to All Page

- Every page will have a menu bar at the top. This will allow users to move fluidly around our system from any page. All users, whether logged in or not, will have access to the same menu bar although some of the web pages may not be available unless a user has logged in. The menu bar will offer the following options:



Page 1: Home Page

- The home page will be the first page users see when they connect to the website. Users will be greeted by a welcome message for our web site. They will also be presented with the following list of options:

Action	*Directed To*
Create new Account	**Page 4**
Login	**Page 6**

- The application layer will display the same HTML home page for all users who connect to the system.

Page 2: Search Page

This page will consist of a series of questions that will assist a user in searching our database. A User selects whether they would like to search for a book by author, title, or ISBN number. They then enter a non-case sensitive text string used to search for their book. Some examples of possible queries are:

Author = Charles Dickens
Title = American Pastoral
ISBN = 1829853333

Users will be presented with a list of books and have the ability to view individual books book information pages, **Page 3**, for each book in the database.

The application layer will load a generic web form as described above. When the user submits the query, the application layer will call a JSP to retrieve the requested information from the database as well as present this information to the user as HTML.

Page 3: Book Information Page

- For each book on the website, the server will generate a unique book information page based on the ISBN number of the book. Users will be presented with information such as the books title, author, price, and availability.
- Based on the quantity of the book currently in stock, the user will receive one of the following messages about the book's availability:

Quantity in Stock	Availability Message Displayed
< 0	Usually ships within 4 weeks
0-4	Usually ships within 1 week
5-19	Usually ships within 2 to 3 days
≥ 20	Usually ships within 1 to 2 days

- The application layer will generate a unique book information page by querying the database with the provided ISBN number in the link. For example, a call for a book with the ISBN number of '123222111' would be of the following form:
- http://server.com/bookhome.jsp?ISBN=1232221111
- The application layer will also check for invalid, missing, or unknown ISBN numbers and report to the user if the book they requested cannot be found on the system.

Page 4: Account Creation Page

- Users without accounts or those wishing to create new ones can do so by choosing to create a new account from the common menu bar or from the system home page, **Page 1**. They will be asked a series of personal questions including choosing a new username for the system as well as a password. If any of the information they provide is incorrect, missing, or invalid, they will be asked to enter it again. For example, if the user enters a password that is too short or choosing a username that is all ready in the system, they will be informed that these fields are invalid and that they must re-enter their information before proceeding.
- After the account information has been entered correctly, the account will be created by the system and the user will be automatically logged in and directed to their shopping basket, **Page 5**.
- The application layer will check the information the user enters thoroughly each time they submit it before attempting to create the account. If all the data passes, the server will attempt to make the account and check for account creation errors.
- In addition, the Account Creation Page will invoke the Login Page's JSP file from **Page 6** in order to log the user into their account directly after it has been created.

Page 5: Shopping Basket

- The shopping basket is a temporary state page that shows the user which items they have added to their order during their current visit to the website. In addition, users can change their shopping cart by both modifying the quantities of books in their baskets, as well as deleting books from their basket. To make changes to their shopping cart, users will need to click an **Update** button that will reload their shopping cart with their new options present.
- Note that no information contained in the shopping basket is maintained by the system permanently; so that a user's shopping basket is erased each time they connect to our server.
- The application layer will present a form to the user listing each of their books and the current quantities of each book they wish to order. The form will allow users to modify these quantities taking care to allow only valid integers from a valid numerical range. The form will submit to itself by saving the shopping cart information to a cookie on the user's computer and then reloading the shopping cart based on the user's new cookie information.

Page 6: Login Page

- The login page will ask a user for their username and password to use to log into the system. In addition, they will have the following options based on the information they provided:

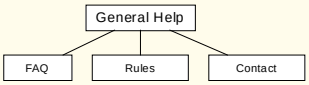
Action	*Directed To*
Forgot Password	**E-mail Message to Administrator**
Login: Incorrect Information	**Page 5**
Login: Correct Information	**Back to Page 6**

- The application layer will query the database with the information the user has provided and if it is valid for a user in the system, it will log a user in by creating a cookie on the user's machine.

Page 7: Help/Information Page

- The help/information Page is a sub-section of the main website, which attempts to help users who may have difficulty, find their way. They are greeted by a welcome message telling them about our site as well as given the choice to view the following pages:

Action	*Directed To*
View F.A.Q.	**Page 8**
View Rules	**Page 9**
Display Contact Information	**Page 10**



- On all help pages, the application layer will simply load HTML webpages for any user requesting the information.

Page 8: F.A.Q.

- The F.A.Q. Page, or Frequently Asked Questions Page, will inform the user of common questions many users have about the system or how to use the system as well as providing a solution to the problem.

Page 9: System Rules Page

- Our website will have a page displaying system rules that users, by using our system, are required to follow.

Page 10: Contact Page

- This page will make available to the user information on how to contact a staff member via e-mail to help them with any problem they might have.

Page 11: Checkout Page

The Logged In User

- Only when a user has at least one item in their shopping basket and has logged into our system will they be able to proceed to the checkout page. First, they will be asked to confirm the items in their order. Next, they will be asked to provide a valid credit card for the order as well as shown their shipping address for the order. Finally, they will be shown the information for their entire order and asked to confirm it. Once they confirm the order, it is sent to the database for processing.
- Even if quantities of a specific book are not available at the time of purchase, the transaction will still be completed. If a book's quantity in stock falls to below zero, that will represent to the store owners that X number of holds have been placed on the book, to be filled as soon as new copies of the book arrives.
- The application layer will present multiple forms to the user each asking them a small set of questions including asking them for their credit card information and whether they would like to confirm their order or not. Each form will submit to a JSP, which will then generate a new form based on the results of the previous form. When all the information has been processed and confirmed, the order will be added to the database and the user will be informed.
- In addition, the quantities of the books in stock will be decremented by the quantities of books the user purchased for each book.

A Sample Result

Demands

- Hand in a report about the assignment
  - Main contents
  - Database design
  - Function design
  - Software design & implementation
  - Database query
- Package your database, source code, executable file into a zip file Sno_A6.zip .
- Three students a group
- If use Access database, should consider concurrency control by yourself. Should discuss these kinds of problems in your report

Submission

- Deadline1:
  Requirement analysis report & system plan
  Sno1_Sno2_Sno3_A6.1.doc
- Deadline2:
  System design & database design
  Sno1_Sno2_Sno3_A6.2.doc
- Deadline3:
  Final report & result
  Sno1_Sno2_Sno3_A6.3.zip