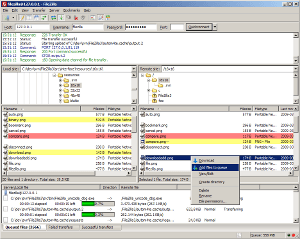
COMP3278 Introduction to Database Management Systems

Tutorial 2 - My Book Catalogue

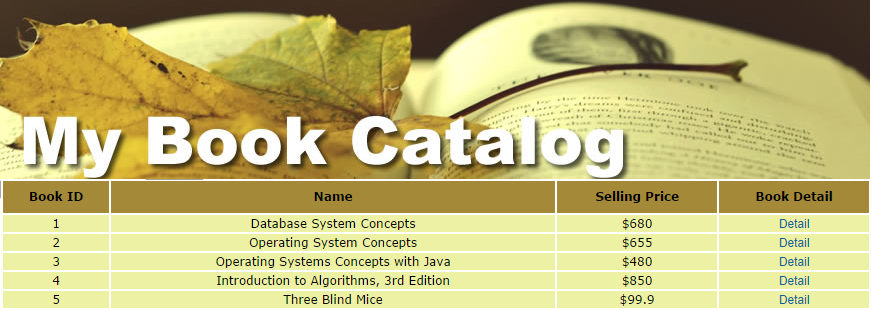
**Section 3. Building applications.**

**We are going to have 4 tasks in this section:**

* **Task 1.** Building catalog.php – display all books in the database.
* **Task 2.** Enhancing catalog.php by adding a drop down menu.
* **Task 3.** Building book.php – display the details of a book, and a list of authors of the book.
* **Task 4.** (Exercise) Building author.php – display the details of an author, and a list of books written by the author.

**Preparation: Dowload Filezilla use this link:** **https://filezilla-project.org/  
And open it like this:**

**Task 1.** Building catalog.php – display all books in the database.



* **Step 1.1.** Open the catalog.php file in 3278\_tutorial2/. (Right click the file > Edit with sublime text)
* **Step 1.2.** Add the following code to connect to the database system.

|  |
| --- |
| <?php  $host = "localhost"; // In CS department, the database is located in a machine called sophia. If you using your own laptop, you can use localhost  $username=""; // Your CSID, or use root if you use your own laptop  $password=""; // Your MySQL password.  $database="book"; // In CS department, we create a database for you with name equal to your CSID.  // Connect to database server  $con = mysqli\_connect($host,$username,$password);  // Select the database we work on  mysqli\_select\_db($con,$database) or die( "Unable to select database");  // Step 1.3: More codes here to display the interface in browser.  // Last step. Close the MySQL database connection  mysqli\_close($con);  ?> |

* Open a browser and visit catalog.php :

<https://i7.cs.hku.hk/~yourCSID/catalog.php>

* Update the $username , $password, $database so that catalog.php can connect to the database. (Reload the page and you should see no error message if the MySQL login info is correct).
* **Step 1.3.** Add the following HTML code in catalog.php to display the interface in browser.

|  |
| --- |
| echo "<!DOCTYPE html>"; echo "<html>";  echo "<head>";  echo "<title>My Catalog</title>";  echo "<link rel='stylesheet' type='text/css' href='style.css'>";  echo "</head>";  echo "<body>";  echo "<div id='header' ></div>";  // Step 1.4-1.6 More codes here to select and display book info from database  echo "</body>";  echo "</html>"; |

* + <!DOCTYPE html> means the generated document is an HTML file (Browsers should interpret the markup with HTML standards).
  + The tags <head>, <title>, <body>, <div> are HTML tags, they are responsible for marking up the document for browser to render a web page.
  + <link ref='stylesheet' type='text/css' href='style.css'> is HTML markup to tell the browser to look for the styling information stored in the file style.css.
  + <div id='header' ></div> specifies an empty division in the HTML page, the logic in style.css will tell the browser this division should display a banner “My Book Catalog” (For your interest, you can open style.css and see the CSS rules, CSS are covered in the course **COMP3322 Modern technologies in WWW**).
* **Step 1.4.** Create an SQL to retrieve all books from database

|  |
| --- |
| // Prepare the database query  $query = "SELECT \* FROM t3\_books;"; |

* **Step 1.5.** Execute the query and store the returned tuples in $result

|  |
| --- |
| // Execute the query  $result = mysqli\_query($con,$query) or die( "Unable to execute query:".mysqli\_error($con)); |

* + mysqli\_query($con,$query)will call MySQL database to execute the query stored in $query.
  + or die(...) is the part that specifies what to output if the query failed (e.g., the query doesn’t follow the syntax of SQL).
  + mysqli\_error()returns the error message if the query was failed to execute.
* **Step 1.6.** Add the following HTML code to build a table for displaying the selected tuples.

|  |
| --- |
| echo "<table>";  echo "<tr><th>Book ID</th><th>Name</th><th>Selling Price</th><th>Book Detail</th></tr>";  // Step 1.7. Codes to retrieve each tuple selected.  echo "</table>"; |

* + <table> is an HTML tag to tell the browser to render a table.
  + <tr>...</tr> marks up a table row.
  + <th>...</th> marks up a table cell (header cell)
  + <td>...</td> marks up a table cell (normal cell)
* **Step 1.7.** Add in the following code to retrieve the tuples that are stored in $result.

|  |
| --- |
| while($row = mysqli\_fetch\_array($result, MYSQLI\_ASSOC))  {  // Step 1.8. Codes to handle display of each selected tuple  } |

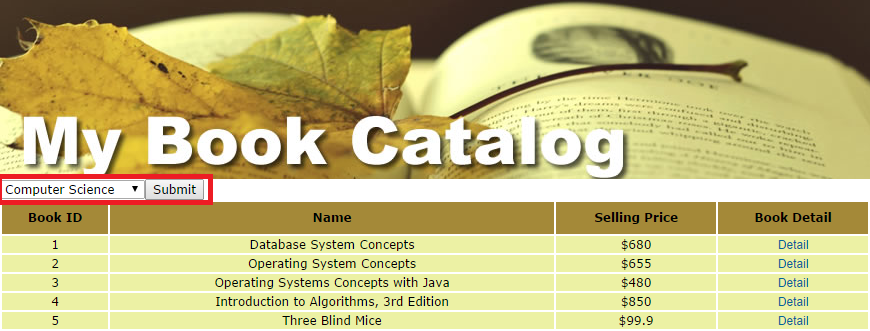
* **Step 1.8.** Handle the display of each selected tuple.

|  |
| --- |
| echo "<tr>";  echo "<td>".$row['bookID']."</td>";  echo "<td>".$row['name']."</td>";  echo "<td>$".$row['price']."</td>";  echo "<td><a href='book.php?bookID=".$row['bookID']."'>Detail</a></td>";  echo "</tr>"; |

* Reload catalog.php in browser to see the result.

**Task 2.** Enhancing catalog.php by adding a drop-down menu.

We would like to enhance by adding a drop-down menu. The drop-down menu consists of the **categories** of the books. After we have selected the category and click the submit button, catalog.php should be reloaded and display the list of books that belongs to the selected category.



* **Step 2.1.** Create an HTML form. Let’s add the following code after the line echo "<div id='header' ></div>"; in catalog.php, so that the form will display right after the banner.

|  |
| --- |
| echo "<form action='catalog.php' method='GET'>";  // Step 2.2 Create a dropdown menu here.  echo "</form>"; |

* + <form> is an HTML code to create a web form.
  + action='catalog.php' means that the data in the web form will forward to catalog.php (i.e., the same page).
  + method='GET' means that the data will forward using HTTP GET method. i.e., the data will be appended at the end of the URL, e.g.,

[http://i7.cs.hku.hk/~yourCSID /catalog.php?categoryID=1](http://i7.cs.hku.hk/~yourCSID%20/catalog.php?categoryID=1)

* + We can use PHP variable $\_GET['categoryID'] to access to the value of the HTTP GET variable.
* **Step 2.2.** Create a drop-down menu inside the <form>.

|  |
| --- |
| echo "<select name='categoryID'>";  // Step 2.3. Add in options in the drop down menu here  echo "</select>"; |

* + name='categoryID' means that the selected option value can be accessed in the HTTP GET variable named categoryID. (i.e., We can use PHP variable $\_GET['categoryID'] to access to the value of the HTTP GET variable.)
* **Step 2.3.** Add in the option(s) to the drop-down menu, since the categories information are stored in the “**t3\_categories**” table in the database, we need to access the database to retrieve the categoryID and name of all categories.

|  |
| --- |
| // Prepare the database query  $queryB = "SELECT \* FROM t3\_categories;";  // Execute the query  $resultB = mysqli\_query($con,$queryB) or die( "Unable to execute query:".mysqli\_error()); |

* **Step 2.4.** Each selected tuple becomes an option in the dropdown menu.

|  |
| --- |
| // Each selected tuple becomes an option in the dropdown menu.  while($rowB = mysqli\_fetch\_array($resultB, MYSQLI\_ASSOC))  {  echo "<option value='".$rowB['categoryID']."'>";  echo $rowB['name'];  echo "</option>";  } |

* **Step 2.5.** Create a submit button in the HTML <form>. Please put the following code **after** the line echo "</select>"; and before the code echo "</form>";

|  |
| --- |
| echo "<input type='submit'>"; |

* **Step 2.6.** Update the $query to select the book from database. Let’s replace the line $query = "SELECT \* FROM t3\_books;"; with the following code.

|  |
| --- |
| if (isset($\_GET['categoryID'])){  $query = "SELECT \* FROM t3\_books B1, t3\_belongs B2 WHERE B1.bookID=B2.bookID AND B2.categoryID=".$\_GET['categoryID']." ;";  }else{  $query = "SELECT \* FROM t3\_books;";  } |

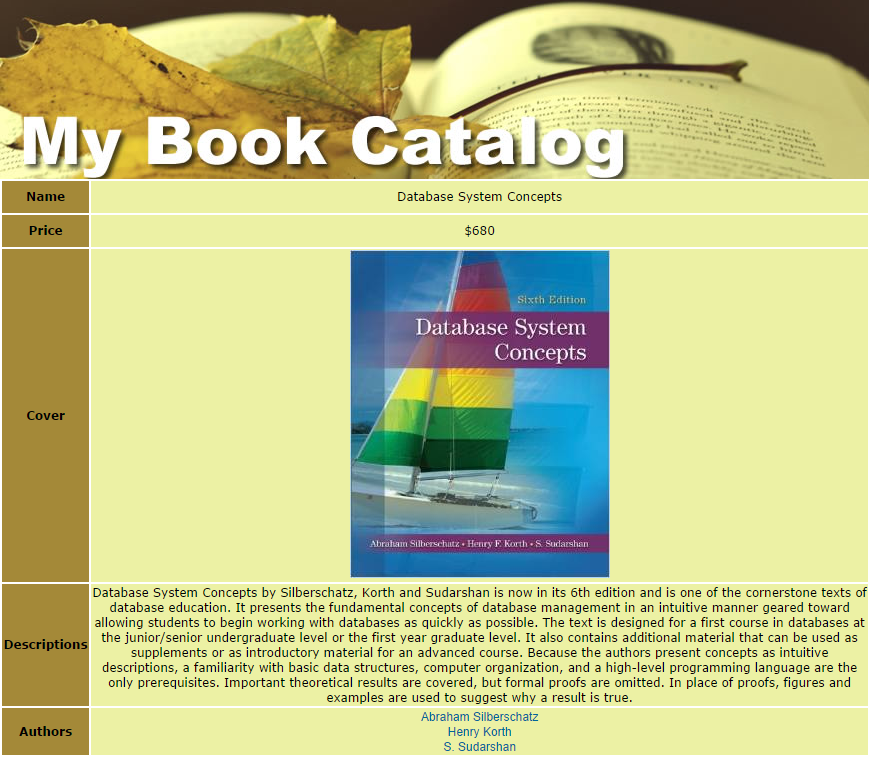
* + $\_GET['categoryID']is the variable that hold the value of the selected **catagoryID** after users submit the <form>.
  + isset() is a PHP built in function that checks if a variable exists or not. isset($\_GET['categoryID'])) returns true if ($\_GET['categoryID']) exists (i.e., users visit the catalog.php by clicking the submit button in the <form>)
  + isset($\_GET['categoryID'])) returns false if ($\_GET['categoryID']) does not exist (i.e., users visit the catalog.php by other mean, so our $query simply retrieves all books)
* **Step 2.7.** **Checkpoint** – reload the page and see if the drop-down list works.
  + Note that there are a lot to improve in this part. For example, the drop-down list should have the default selected value marking the category we submitted. In addition, we can use HTML checkboxes to replace the dropdown list.
  + Various web interface related technologies such as HTML5, CSS3, AJAX, XML …etc will be covered in the course **COMP3322 Modern Technologies on World Wide Web.**

**Task 3. Building book.php**

In catalog.php, each book entry has a “**Book detail**” URL that links to the following page:

[http://i7.cs.hku.hk/~yourCSID /book.php?bookID=X](http://i7.cs.hku.hk/~yourCSID%20/book.php?bookID=X)

* **X** is the **bookID** of the book in the database.
* Note that bookID is also a HTTP GET variable, we can access its value in book.php using the variable $\_GET['bookID'].
* book.php is blank now, we would like to build book.php in this task so that it will display the following detail information of a book.



* **Step 3.1.** Open book.php in 3278\_tutorial2/. (Right click > Edit with sublime text)
* **Step 3.2.** Add the following code to connect to the database system.

|  |
| --- |
| <?php  $host = "localhost"; // In CS department, the database is located in a machine called sophia. If you using your own laptop, you can use localhost  $username=""; // Your CSID, or use root if you use your own laptop  $password=""; // Your MySQL password.  $database="book"; // In CS department, we create a database for you with name equal to your CSID.  // Connect to database server  $con = mysqli\_connect($host,$username,$password);  // Select the database we work on  mysqli\_select\_db($con,$database) or die( "Unable to select database");  // Step 3.3: More codes here to display the interface in browser.  // Last step. Close the MySQL database connection  mysqli\_close($con);  ?> |

* Update the $username , $password, $database so that book.php can connect to the database. (Reload the page and you should see no error message if the MySQL login info is correct).
* **Step 3.3.** Add in the HTML code in book.php to display the interface in browser

|  |
| --- |
| echo "<!DOCTYPE html>"; echo "<html>";  echo "<head>";  echo "<title>Book detail information</title>";  echo "<link rel='stylesheet' type='text/css' href='style.css'>";  echo "</head>";  echo "<body>";  echo "<div id='header' ></div>";  // Step 3.4-3.6 More codes here to select and display book info from database  echo "<a href='catalog.php'>Back to Catalog</a>";  echo "</body>";  echo "</html>"; |

* **Step 3.4.** Create an SQL to retrieve books from database

|  |
| --- |
| // Prepare the database query  $query = "SELECT \* FROM t3\_books where bookID='".$\_GET['bookID']."';"; |

* **Step 3.5.** Execute the query and store the returned tuples in $result

|  |
| --- |
| // Execute the query  $result = mysqli\_query($con,$query) or die("Unable to execute query:".mysqli\_error()); |

* **Step 3.6.** Add in the HTML code to build a table for displaying each selected tuples.

|  |
| --- |
| // Build a table for displaying each selected tuples  echo "<table id='catalog'>";  while($row = mysqli\_fetch\_array($result, MYSQLI\_ASSOC))  {  echo "<tr>";  echo "<th>Name</th>";  echo "<td>".$row['name']."</td>";  echo "</tr>";  echo "<tr>";  echo "<th>Price</th>";  echo "<td>$".$row['price']."</td>";  echo "</tr>";  echo "<tr>";  echo "<th>Cover</th>";  echo "<td><img src='".$row['picture']."'></td>";  echo "</tr>";  echo "<tr>";  echo "<th>Descriptions</th>";  echo "<td>".$row['description']."</td>";  echo "</tr>";  echo "<tr>";  echo "<th>Authors</th>";  echo "<td>";  // Step 3.7 More codes here to display a list of authors of the book here.  echo "</td>";  echo "</tr>";  }  echo "</table>"; |

* **Checkpoint:** Please reload book.php?bookID=1, you should see the details of the book are display except the blank cell for the Authors part.
* **Step 3.7.** Display the list of authors of the book.

|  |
| --- |
| // Prepare another database query  $queryB = "SELECT \* FROM t3\_authors A, t3\_writes W where A.authorID = W.authorID AND W.bookID='".$\_GET['bookID']."';"; |

* + This SQL selects the list of authors of the book with **bookID** equals to $\_GET['bookID']. We will explain how this SQL works in Chapter 3 ☺.
* **Step 3.8.** Execute the query and store the tuples to$resultB.

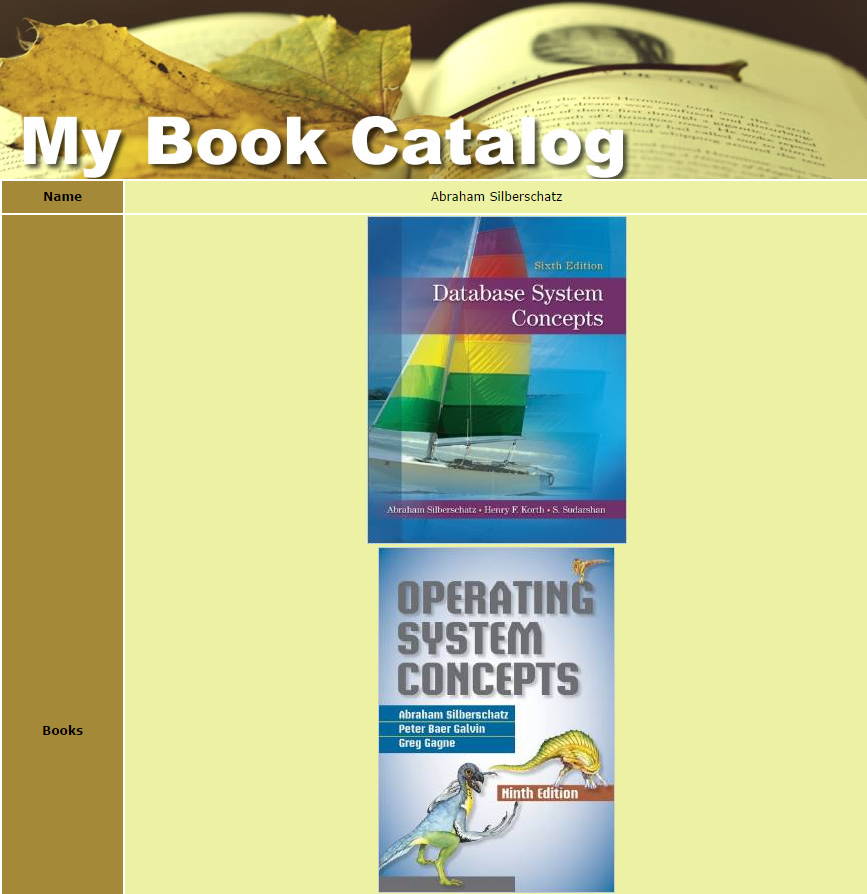
|  |
| --- |
| // Execute the query  $resultB = mysqli\_query($con,$queryB) or die( "Unable to execute queryB :".mysqli\_error()); |

* **Step 3.9.** Display each selected author as a hyperlink within the cell.

|  |
| --- |
| // Display each selected author as a hyperlink within the cell.  while($rowB = mysqli\_fetch\_array($resultB, MYSQLI\_ASSOC))  {  echo "<a href='author.php?authorID=".$rowB['authorID']."'>".$rowB['name']."</a><br>";  } |

* + <a> is HTML tag to mark up hyperlink.
  + href='[URL]' tells the browser to link the hyperlink to [URL].
  + We are linking to the page 'author.php?authorID=**X**' where X is the selected authorID, therefore we append 'author.php?authorID=' with $rowB['authorID'] to form the URL of the hyperlink.
* **Step 3.10. Checkpoint:** Let’s reload book.php?bookID=1 to see if the book details and list of authors are display in the page.

**Exercise.** Let’s click on the name of the author “[Abraham Silberschatz](http://i2.cs.hku.hk/~ckchui/3278_tutorial3/author.php?authorID=1)”, we will browse to author.php?authorID=1. But author.php is not implemented yet. Let’s implement author.php so that it will display the following information of the author Abraham Silberschatz (with authorID=1).



End of tutorial 2

Please feel free to contact our TA if you have any problem when working on this tutorial☺. We are very happy to help you.