

Lab 7 (100 pts)**Objectives: Learn**

- Using HTML, PHP and Oracle SQL to build a Web application that reads the user input via an HTML form, processes the input on the server side using PHP and Oracle SQL.
-

Part 1 (60 pts)

Create the following tables for exercises.

```
Create table Books_read(title varchar(40) primary key,  
author varchar(30),category varchar(20),comments varchar(50));
```

```
Create table Book_Highlights(title varchar(40),  
highlight varchar(30), Foreign Key(title) references Books_read(title) );
```

Exercise 1 (15pts)

In this exercise, you will use an HTML web page that contains a form for user input and a PHP script on the server side that processes the input and inserts the input into the tables, **Books_read** and **Book_HighLights**. You will use the following files:

- [bookInputForm.html](#)
- [procBookInput.php](#)
- [showBooks.php](#)

- a) Copy all the files into your script directory (where your php scripts are)
- b) Open the **bookInputForm.html** with your text editor and give the correct path to your `procBookInput.php` script, in the **action attribute** of the **form**. Save your changes.
- c) Now, open the **procBookInput.php** file with your text editor. Edit the code to put your login and password. **Note:** At the end of the lab session, please feel free to change your password.

From your browser window, using its URL, open **bookInputForm.html**. You should see a form displayed.

This is the input form to insert data into the `Books_read` and `Book_highlights` tables.

- d) Enter data of a book you have read in the form. Click submit.
- e) Do you see any messages whether your submission succeeded?
- f) Now, from your SQLPlus prompt, do a select * on both tables and check if the data you submitted via the form is inserted into the tables. If the tables do not show any data, give it a second or so, and do the query again. Sometimes, there is a lag time before you see the data. You may log out of SQL and log back on and try the query.
- g) Enter a few more rows of data into the tables, using the HTML form.

Exercise 2 (15pts)

In this exercise, you will modify the HTML form in the **bookInputForm.html** file as follows:

- a) Add a few more radio buttons with values of your choice for category.
- b) Add a few more Check box options of your choice for Highlights.
- c) After you add your changes, open the form again through the browser window and add data for a couple of more books.
- d) From the SQL prompt, check if the data is inserted correctly.

Exercise 3 (10pts)

- a) Create a PLSQL function called bookCount as given below.

Create or Replace Function bookCount

Return Integer is

l_cnt Integer;

BEGIN

SELECT count(*) INTO l_cnt

FROM Books_Read ;

return l_cnt;

END;

/

Make sure the function compiles without any errors.

In this exercise, you will use showBooks.php script, which when run opens a web page with two clickable links.

- b) Open the file showBooks.php in your text editor. Edit the path of the file in the href attribute in the <a> tag. Save your changes.
- c) Now, open the file, showBooks.php in the browser window. Click on the links and check if they are showing the information of books and count of books in response to the clicks.

Exercise 4 (20pts)

- a) Now it is your turn to complete the PHP function called **bookDetails()** in file, **showBooks.php** that shows the **title**, **author**, **category** and **highlights** of the books. **Note:** You may need to join two tables in your query.
- b) Using styles for color, font-size (and any other style elements that you use) display the results in a readable and pleasing manner.
- c) Add a link (similar to the two links already existing) to call the function, bookDetails.
- d) Open the php script, **showBooks.php** in the browser window and test your function by clicking the new link you have added.

Part 2 (40 pts)

In this exercise, you will develop a web application with an HTML page, PHP script with SQL, to display the names of employees in AlphaCoEmp table (you have created in your earlier labs) in a specific salary range that is selected by the user.

Exercise 5 (10 pts)

Create an HTML page called **alphacoemp.html** where the page should contain a list of radio buttons to represent the salaries, namely, **130000,110000,90000,70000,50000**.

When the form is submitted, it should invoke the PHP script, **showEmployees.php** that you will write in the next exercise.

Exercise 6 (30 pts)

Write a PHP script called **showEmployees.php** that processes the input received when the form is submitted via **alphacoemp.html** page. The script should get the input from the form when a radio button is selected, to input a salary. It should connect to the database and retrieve the employee names from the AlphaCoEmp table whose salaries are \geq the number selected from the radio button (let us say x) and $< x+20000$. For example, if the user selects the radio button for 70000, then the employee names with salaries ≥ 70000 and below 90000 should be displayed.

Note: Refer to HTML page and PHP scripts that you have used in Part 1.