Based on 100 points total, roughly 0.6 per check box unless noted as a %. Extra credit evaluated separately.

Lab 3 - Passing values Collecting Data.

The purpose of this exercise is to:

> Link two web pages together passing a value from one page to the other to be used in a query to retrieve one record and display it. We will increase our database security and get a form to display a sql insert statement upon submission. You will be inserting data into two different tables. You will also need to use version control software. If that is not working you cannot receive a grade higher than 50% even if you have completed more than half of the check boxes.

As a reminder all labs are due by noon. Many classes have them due by midnight but I have extended all the due dates to noon the next day to allow you a little more time.

NOTE: When you make a commit (there are LOTS of commits) it is assumed that the code is working at that point.

Set up version control for this lab.

□ Make a copy of your dev-lab2 folder naming it dev-lab3. i.e. just copy the whole folder and rename the copy as this will have all the files to start.

□ Be sure to have a README.md with just your First and Last Name.

□ Create a new private GitHub project in the class Repo named Lab-3-FirstName-LastName. For example mine would be Lab-3-Robert-Erickson.

□ Verify that you have used the case and hypens in your repo name as detailed above.

□ Keep the content of the first GitHub page saving them in gitcommands.txt overwriting the commands from lab 2. This file should contain only the commands for lab 3.

□ Erase all previous commits from your dev-lab3 folder so we can start new. Be sure you are in the correct folder dev-lab3

```
rm -rf .git
```

□ Set up your dev-lab3 folder to be a git repo (git init) on your computer.

```
git init
```

□ Check if a remote is connected and remove it. It should already be deleted but it does not hurt to check.

```
git remote -v (will show you if a remote is there)
git remote rm origin (if one is there delete it)
```

□ Now connect your local computer to your git repo:

```
git remote add origin command
```

□ Make a commit after you sFTP your files and they are working as before.

Set up the live site.

□ Using Terminal or git bash shell, connect to the server and get into your class folder creating the live folder (live-lab3)

□ Initialize live-lab3 as a git repo.

□ Connect your live folder with the GitHub repo.

□ Pull your files live.

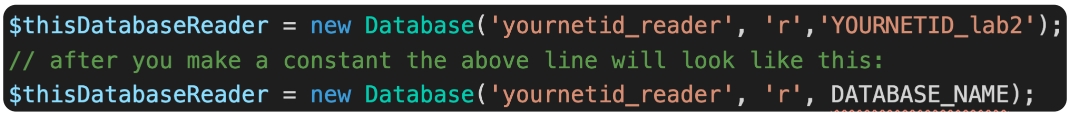
□ After you pull always verify the site is working.

Setup a new database.

□ Using the web site webdb.uvm.edu (at the botttom of the page) create a new database named cs148_lab3

□ Using phpMyAdmin click on cs148_lab2 database, tblWildlife then choose operations and copy the structure and data to LCARRERA_cs148_lab3

□ Since we are changing databases, we have to change the code. Let's move the database name to constants.php NOTE: top.php has database connection for a new pdo that has the database name.



```
$thisDatabaseReader = new Database('yournetid_reader', 'r','YOURNETID_lab2');
// after you make a constant the above line will look like this:
$thisDatabaseReader = new Database('yournetid_reader', 'r', DATABASE_NAME);
```

top.php using the constant.

□ Make a commit for setting up the database connection. NOTE: If you connection does not work your web site will break.

□ Make sure you copy lib/pass.php to the live site.

Page to display data based on a value.

□ Create a file named displayCritter.php using the same template as your other files. Be sure to change the heading.

□ Verify displayCritter.php template works.

□ Make a commit for displayCritter.php

□ displayCritter.php will have an id value passed in with the get format, create a variable to hold this sanitized integer. Be sure to use a good name for your variable.

□ Pick a name for the variable to be passed in. The variable name is in the URL for example: www.test.com?*variable*=value

□ Temporarily display this value and test the page to be sure it works. Use a simple print statement being sure to sanitize the variable. Then test to pass in nothing, 5, <script>alert('hi');</script>, 300.3 it should display 0, 5, 0, 300

□ Make a commit for testing the value passed in.

□ displayCritter.php needs to display the information for the animal clicked on which involves a query with a parameter, for each loop etc.

□ displayCritter.php should display all appropriate data. Just because a column is in the table should it be displayed?

□ Be sure to properly call the select function.

□ The layout and look are up to you but should be presentable.

□ Update the footer with citation of where the information comes from.

□ Make a commit for displaying the record.

□ Add a link to adoptCritter.php sending the id to a form you will create soon. The location and look is up to you but it should be prominent.

□ Make a commit for adding the link.

Connect index.php to displayCritter.php.

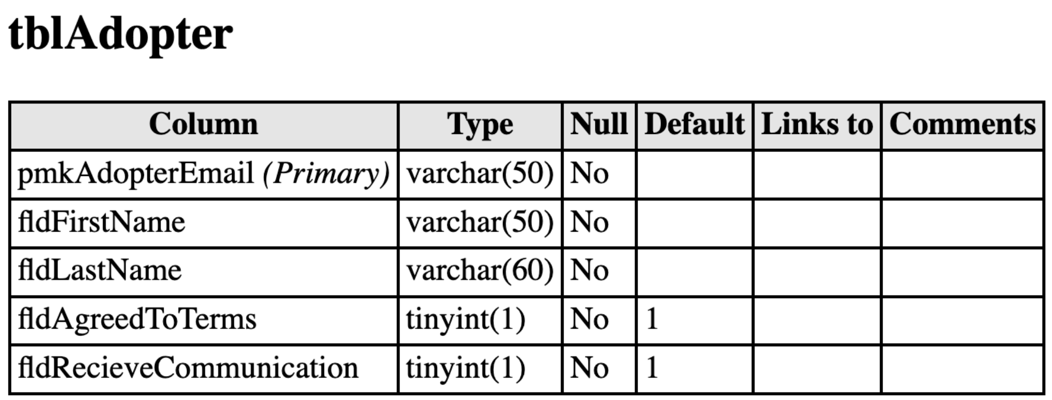
□ Make each figure element a hyper link passing the primary key to displayCritter.php

□ Verify your links works correctly.

□ Make a commit for index.php

Make the tables to hold Adopt a Critter Dontation information.

□ Using phpMyAdmin (sql or web form) create a table named tblAdopter based on this data dictionary.

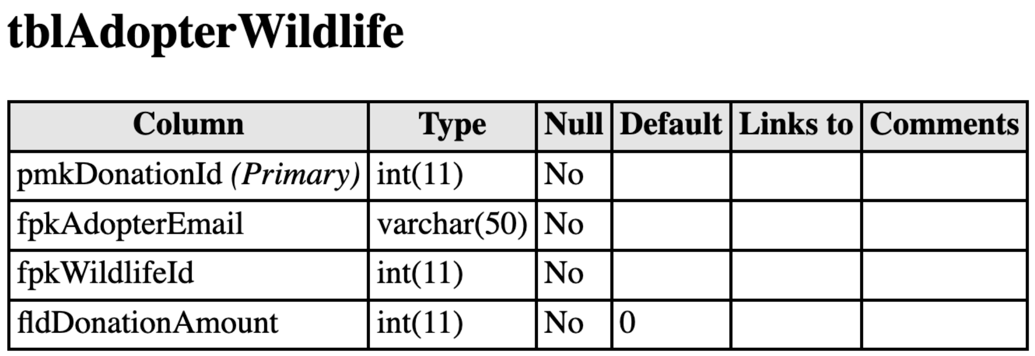


tblAdopter

Column	Type	Null	Default	Links to	Comments
pmkAdopterEmail *(Primary)*	varchar(50)	No			
fldFirstName	varchar(50)	No			
fldLastName	varchar(60)	No			
fldAgreedToTerms	tinyint(1)	No	1		
fldRecieveCommunication	tinyint(1)	No	1		

Data Dictionary.

- □ Save the create table sql commands in sql.php. If you used the form, export the table and view the sql to get the create table statement.
- □ Using phpMyAdmin (sql or web form) create a table named tblAdopterWildlife based on this data dictionary. NOTE pmkDonationId is auto increment.



tblAdopterWildlife

Column	Type	Null	Default	Links to	Comments
pmkDonationId *(Primary)*	int(11)	No			
fpkAdopterEmail	varchar(50)	No			
fpkWildlifeId	int(11)	No			
fldDonationAmount	int(11)	No	0		

Data Dictionary.

- □ Save the create table sql commands in sql.php. If you used the form, export the table and view the sql to get the create table statement.

Create a form

- □ Make a new file named adoptCritter.php based on your template.
- □ Create a variable to hold the id value passed in. NOTE: This is the same as you did in displayCritter.php
- □ Sanitize the variable passed in.
- □ Select the id and the common name from the database.
- □ Display the common name in a heading. For example: click on beaver the heading would be Adopt a Beaver
- □ Make a commit for this template.

- □ Make adoptCritter.php a form. (ie form element and submit)
- □ Add the function getData to sanitize the data as shown in the text Figure 1.10.2 (my book has two 1.10.2). (just add the function not how to use it)
- □ Add an if statement before the form code to print out the post array if the form was submitted and DEBUG is set to true. NOTE: use two if statments.
- □ Make a commit for the start of this form.

Adding a form element

- □ A. Create and initialize (default value) a php variable for the dollar amount to donate. Be sure to name your variable correctly.
- □ B. Add a range slider to your form to accept the dollar amount to donate in the range of $25 to $1000 in $25 increments. Be sure to name your element correctly.
- □ C. Have your range slider set to the default value.
- □ D. In your form processing if statement sanitize the data from your range slider. refer to Figure 1.10.2 for sanitize and you can cast that to int.
- □ E. In your form processing if statement validate the data from your range slider.
- □ Remember to create a flag if you should save the data.
- □ F. Make a commit for this stage of the form adding just the range slider.

Testing

- □ Create a new method in your database.php class to display your query statement with the data values. You can create your own or use my example: http://sandbox.onlinephpfunctions.com/code/33dbf58a063132647c8bcc65426c8a8c79243b38
- □ If there are no errors on the form submission, create a sql statement to insert the type of critter. We are not executing the query yet, only building it.
- □ Using your method you created display the query.
- □ Copy the query printed into PHPMyadmin to verify that it inserts the record. Delete the record afterwards as it is missing all the other data at this point and we just wanted to test it.
- □ Make a commit for this stage of the code.

Complete the form, testing each element before you go on.

Can a person adopt more than one animal?

- □ Repeat step A Create and initialize (default value) a php variable for email address.
- □ Repeat step B Add a text box for email address.
- □ Repeat step C Have your text box set to the default value for email address.
- □ Repeat step D Sanitize the data from your text box for email address.
- □ Repeat step E Validate the data from your text box for email address.
- □ Add email address to your sql insert statement.
- □ Add email address to a different sql insert statement. NOTE: There will be two sql statements inserting into two different tables.
- □ Verify the sql insert statements are working (deleting the records afterwards).
- □ Repeat step F make a commit for email address.

- □ Repeat step A Create and initialize (default value) a php variable for first name.
- □ Repeat step B Add a text box for first name.
- □ Repeat step C Have your text box set to the default value for first name.
- □ Repeat step D Sanitize the data from your text box for first name.
- □ Create a function to check for letter (alpha) and space data only to be used in your validation for first name.
- □ Repeat step E Validate the data from your text box for first name.
- □ Add first name to the correct sql insert statement.
- □ Verify the sql insert statement is working (deleting the record afterwards).
- □ Repeat step F make a commit for first name.

- □ Repeat step A Create and initialize (default value) a php variable for last name.
- □ Repeat step B Add a text box for last name.
- □ Repeat step C Have your text box set to the default value for last name.
- □ Repeat step D Sanitize the data from your text box for last name.
- □ Repeat step E Validate the data from your text box for last name.
- □ Add last name to your sql insert statement.

☐ Verify the sql insert statement is working (deleting the record afterwards).

☐ Repeat step F make a commit for last name.

☐ Repeat step A Create and initialize (default value) a php variable for agreeing to the terms and conditions. Set this to 1

☐ Repeat step B Add a checkbox for agreeing.

☐ Repeat step C Have your checkbox set to the default value for agreeing.

☐ Repeat step D Sanitize the data from your check box for agreeing.

☐ Repeat step E Validate the data from your check box for agreeing. You may not need to do this step.

☐ Add agreeing to your sql insert statement.

☐ Verify the sql insert statement is working (deleting the record afterwards).

☐ Repeat step F make a commit for agreeing.

☐ Repeat step A Create and initialize (default value) a php variable for receiving email. Set this to 1

☐ Repeat step B Add a checkbox for receiving email.

☐ Repeat step C Have your check box set to the default value for receiving email.

☐ Repeat step D Sanitize the data from your check box for receiving email.

☐ Repeat step E Validate the data from your check box for receiving email. You may not need to do this step.

☐ Add receiving email to your sql insert statement.

☐ Verify the sql insert statement is working (deleting the record afterwards).

☐ Repeat step F make a commit for receiving email.

We need one more piece of data, The wildlife id value that is the value passed into this page.

Step A is already complete.

☐ Repeat step B Add a hidden input for wildlife id.

☐ Repeat step C Have your hidden input set to the default value for wildlife id.

☐ Repeat step D Sanitize the data from your hidden input for wildlife id.

□ Repeat step E Validate the data from your hidden input for wildlife id.

□ Add wildlife id to your sql insert statement.

□ Verify the sql insert statement is working (deleting the record afterwards).

□ Repeat step F make a commit for wildlife id.

Database Insert method.

□ Create a new method to insert a record which will returns true or false. This will look almost like the select method.

□ Edit top.php to include a new Database variable that will allow us to save information named: $thisDatabaseWriter NOTE: Same as $thisDatabaseReader but be sure to use the wrtier username and password variables.

□ Set the code so displayQuery only prints if DEBUG is set to true;

□ Add the insert method call to your form page to insert the records into your datase. NOTE: you need to use $thisDatabaseWriter you just created.

□ Provide feedback to the user on success or failure of the insert.

□ Add a Donation with your form and verify that your records are inserted into the database tables.

□ Make a commit for the insert method.

Style your form.

□ Edit your css file as needed to style your form page to display appropriately.

□ Make a commit for your css.

□ Quality of your css. Are there major issues?

□ Pull to the live site.

□ Make sure you have sFTP pass.php to the live site (it is ignored by git).

Standard Requirements

□ Main goals of assignment met.

□ Be sure DEBUG is set to false.

□ Update your main index.php file with links to all the files in the correct place (public and supporting).

□ Do not use any banned elements (div, span, br).

□ Be sure your text is still legible with and without the Color blindness tool.

□ All files must use proper coding guidelines.

□ Make sure paragraph, lists, forms text is not centered. Centering boxes is ok.

□ (5% of your grade) Proper use of version control (commits on a regular basis, commits small with good message about what was accomplished with this commit) jigsaw.w3.org/css-validator/

□ (5% of your grade) Make sure your files all pass w3c HTML validation: validator.w3.org/

□ (5% of your grade) Make sure your files all pass w3c CSS validation: jigsaw.w3.org/css-validator/

□ sFTP your files to your silk account which is our web server

□ Submit your assignment only once in blackboard. See next direction first.

□ Copy this code as a text submission in blackboard, look for the Write Submission button. You will need to click on the html button once you are in the editor and paste this code there:

```
<a target="_blank"
href="//lcarrera.w3.uvm.edu/cs148/live-lab3">
http://lcarrera.w3.uvm.edu/cs148/live-lab3</a>
```