Lesson Number 3

Name:

Selecting Data

Description:

Selecting all data, a single piece of data, and joined data.

GROUP ACTIVITY - PHP Quick Review

5 minutes with a partner to jot down (without looking at notes) what you remember from last week.

CREATE Review & READ Overview

Review of CREATE

- What is the application flow?
  - User fills out form and submits it
  - HTTP Request is posted to the processing script
  - The processing script processes the data
  - The processing script connects to the database
  - The user's data is written to the database
  - **HTML/PHP page -> POST -> PHP processing page -> Connect to DB -> Store Data**
- What two HTML form element attributes are required to submit data to a PHP processing page?
- What is the name of the super global associative array created by PHP that stores the HTTP Request POST body?
- What HTTP Request method should be used to **CREATE** new data in the application?

Why Use POST over GET to Submit User Data

- POST is slightly more secure than GET as the data is obsfucated (hidden)
  - POST doesn't store historical data in the browser
  - POST requests are not cached, so a fresh request is used each time
- POST is the semantically correct way to create a new resource on the server. This ensures clarity of its purpose
- POST allows you to process file uploads
- POST allows for a greater amount of data to be passed to the server

---

- GET is the semantically correct way for retrieving and reading resources on the server
- GET parameters are viewable within the URL of the browser
- GET requests can be bookmarked and can be stored in the history of the browser
- GET requests can be cached increasing their performance for repeat calls

Idempotent VS Non-Idempotent (IF SEMANTICS ARE OBEYED)

- GET, PUT, and DELETE are Idempotent methods

        - if the same request is repeated, it has no side-effect on the server, as the same result will occur
            - GET request will result in the same data being retrieved
            - PUT will result in the same resource being updated or replaced
            - DELETE will result in the same resource being destroyed
- POST is a Non-Itempotent method
    - if the same request is repeated, it will result in a new resource being created, duplicating data
- POST, PUT, and DELETE are destructive methods, as they will result in a change in the database, however POST is the only one that can repeat that change
- GET, PUT, and DELETE are considered safe methods as their actions can't be repeated
- GET is considered the safest as it can only retrieve/read data, and therefore is non-destructive
- Most browsers don't support the HTTP methods PUT and DELETE, so POST is used instead. The idempotent and non-idempotent idea remains the same, however.

Overview of READ

- What is the application flow?
    - User requests a page or resource
    - Request is sent to a processing page on the server
    - The server connects to the database
    - The server requests the data from the database
    - The data is returned to the user
- WAIT! What about GET???
    - GET has a role in requesting data based on dynamic (changeable) criteria
        - searching a website for a specific resource may use GET to perform this action
            - http://mysite.com?search_term=boat&category=fishing
    - GET isn't necessary if we're retrieving predetermined data such as every record in a table
        - Predetermined Data would be data that the developer has defined as a viewable resource

RESTful URLs

- these URLs tend to follow this structure
    - http://mysite.com/posts/2016/party_of_the_century
- these are like GET requests as the required parameters are still within the URL
    - the processing script will break apart the URL and review each piece to define what resource the user is requesting
        - resource **posts**
        - year **2016**
        - blog **party_of_the_century**

ACTIVITY - Add More Data Into the Database

1. Navigate to this page: http://georgian.shaunmckinnon.ca/lesson-02/examples/new_artist.php
2. Add a new artist and their bio page
3. Get new data

```
SELECT DISTINCT( name ) FROM artists;
```

4. Run the generated SQL insert to populate our database

ACTIVITY - Creating artists.php

Selection Exercise

http://georgian.shaunmckinnon.ca/interactive-examples/game.php?name=select-exercise

Selecting Data

- **HTML/PHP Page Request -> PHP processing page -> Connect to DB -> Requests Data from Database -> Data is Returned to User**
- A template is built to display the data
    - ie: Wikipedia
    - ie: Facebook
    - ie: Amazon
- If more than one record is being returned
    - the data is looped through using the same template to display each element
    - sometimes it is displayed using rows of an HTML table
- the PDO function **fetchAll()** collects all the records into an associative array
- the **foreach()** language construct allows the data to be easily looped through well maintaining its identifiers

```
foreach ( $result as $row ) {
  echo $row['name'];
  echo $row['age'];
  echo $row['address'];
}

// or shorthand
foreach ( $result as $row ):
  echo $row['name'];
  echo $row['age'];
  echo $row['address'];
endforeach;
```

ACTIVITY - Creating new_song.php & add_song.php

Insert Exercise

http://georgian.shaunmckinnon.ca/interactive-examples/game.php?name=select-exercise

The SQL

```
CREATE TABLE songs (
                id int(11) NOT NULL AUTO_INCREMENT,
                title varchar(100) NOT NULL,
                length time DEFAULT NULL,
                artist_id int(11) NOT NULL,
                PRIMARY KEY (id),
                FOREIGN KEY (artist_id) REFERENCES artists (id) ON DELETE
CASCADE ON UPDATE CASCADE
              );
```

Understanding Dynamic Dropdowns

Example Dropdown

- dynamic dropdowns are a staple of the internet
- they force the user to select from a predefined list
- **Why is this a good thing?**
  - limits user error
  - ensures data integrity
- in a relational database we attempt to keep our data to first normal form
  - data that can be duplicated needs to be stored in a different table
  - we can then use a dynamic dropdown to allow users to select an option from that table
  - these tables are generally connect by using a foreign key
    - this enforces data integrity as you can't remove a parent item without first destroying all the children items
  - an example ERD model

    ERD Explaining Dynamic Dropdowns

ACTIVITY - Creating songs.php, artist_songs.php

- creating singular and plural views, and views with relational data

LAB

CREATE

1. create a new table in your database to record data. examples:
   - teams
   - people
   - games
   - movies

    - cars
    - shows
    - make sure your table obeys these laws:
        - we want to ensure rows in our table are not storing duplicate data (within reason)
        - we want to isolate data that can be divided, like full names, addresses, ingredients, instructions/processes, phone numbers (area code from base number)
        - we want to give our data a unique index so it is distinguishable from other data
        - we want to ensure we are not storing similar data in different fields, such as a table that has the columns artist, singer, actor (these could be condensed into one field, performer)
    - your table must contain at least 3 columns
2. create an HTML page and a form to collect user input for you table
3. create a processing script that will write the data to a database
4. add at least 5 entries to your database

READ

1. create an HTML page with an HTML table with a header row and all the datarows pulled from your database in a loop

The Rubric

Criteria	Possible Points	Bonus
SQL table contains at least 3 columns	3	more than 3 (1 bonus point)
SQL table obeys the 4 rules	4	
SQL table contains at least 5 mock entries	2	more than 10 (1 bonus point)
CREATE HTML page contains the proper HTML structure (doctype, head, title, body	2	
Form contains an action and method attribute	10	
Inputs contain a name attribute	3	
The most applicable form element was used for the datatype	3	
Processing Page connects to the mysql database successfully	5	
Processing Page creates a new record successfully	5	
READ HTML page contains the	2	

proper HTML structure (doctype, head, title, body)		
READ HTML table contains a header row	5	
READ HTML table contains all the rows available in the SQL table (there should be at least 5)	10	
HTML pages have Bootstrap attached and are utilizing its classes		5 bonus points (it has to work)