|  |  |  |  |
| --- | --- | --- | --- |
| Name |  | |  |
| Score | / 50 |  |  |
| Update Value | |  |  |
| Make all corrections and resubmit to earn update points | | | |

**Web Data Management**

152-155

**MVC & PDO Evaluation Form  
Student-Selected Project**

🗹 Update Recommended

**CSS**

* (½ point) Forms.css included
* (½ point) Most (if not all) formatting done using CSS

**Controller**

* (½ point) Model libraries correctly imported
* (½ point) Page determines action based on REQUEST value
* (½ point) Default action is appropriate (list)
* (½ point) Filter variable set (empty string if not defined)  
  (may be in list case)
* (½ point) Case statement used to process action
* (½ point) Details case calls function to generate list of combo box values
* (½ point) Details case calls function to get details for selected record
* (½ point) Details case displays details view
* (½ point) List case calls function to get **filtered** list of records
* (½ point) List case displays list view
* (½ point) New case calls function to generate list of combo box values
* (½ point) New case creates empty details array
* (½ point) New case displays details view
* (½ point) Update and save new cases combined
* (½ point) btnSave case first validates inputs
* (½ point) If no errors, IF statement used to call update or insert function
* (½ point) After insert/update, list view displayed
* (½ point) If errors, combo box list created, Request array transferred to details, details view redisplayed
* (½ point) btnCancel displays list view
* (½ point) btnDelete calls delete function then displays list view

**Table 2 Cases**

* (½ point) Details case calls function to generate list of combo box values
* (½ point) Details case calls function to get details for selected record
* (½ point) Details case displays details view
* (½ point) List case calls function to get list of records
* (½ point) List case displays list view
* (½ point) New case calls function to generate list of combo box values
* (½ point) New case creates empty details array
* (½ point) New case displays details view
* (½ point) Update and save new cases combined
* (½ point) btnSave case first validates inputs
* (½ point) If no errors, IF statement used to call update or insert function
* (½ point) After insert/update, list view displayed
* (½ point) If errors, combo box list created, Request array transferred to details, details view redisplayed
* (½ point) btnCancel displays list view
* (½ point) btnDelete calls delete function then displays list view

**JavaScript**

* (1 point) JavaScript file for first table correctly validates form inputs
* (1 point) JavaScript file for second table correctly validates form inputs

**Connection Model**

* (½ point) dsn variable linked to database
* (½ point) username is root
* (½ point) password is empty string
* (½ point) db variable defined as a new PDO object using above variables

**Table 1 Model**

* (½ point) Functions listed alphabetically
* (½ point) All functions appropriately named
* (½ point) All functions include appropriate description
* (½ point) Delete query is correct (uses ID)
* (½ point) Parameter binding correct
* (½ point) getDetails query is correct (uses ID)
* (½ point) Parameter binding is correct
* (½ point) fetch used to store one record in results
* (½ point) getList query retrieves appropriate fields (ID + )
* (½ point) Where clause includes filter parameter
* (½ point) List sorted
* (½ point) fetchAll used to define results
* (½ point) Insert query lists all fields except ID
* (½ point) Formatted fields unformatted
* (½ point) Checkbox values converted to true/false
* (½ point) Query includes parameters for all fields
* (½ point) Parameter bindings are correct
* (½ point) Optional fields set to null when appropriate
* (½ point) isDuplicate function includes logical key parameters
* (½ point) Query counts number of records that match logical keys (parameters)
* (½ point) Parameter bindings are correct
* (½ point) fetch used to retrieve number of matching records
* (½ point) Function returns Boolean value (count > 0)
* (½ point) Update query sets field types for all but ID fields
* (½ point) Formatted fields unformatted
* (½ point) Checkbox values converted to true/false
* (½ point) Where clause designates ID (parameter)
* (½ point) Query includes parameters for all fields
* (½ point) Parameter bindings are correct
* (½ point) Optional fields set to null when appropriate
* (1 point) validateInputs function correctly validates all inputs
* (½ point) Includes code to check for duplicate records
* (½ point) isDuplicate only called if no previous errors and logical key(s) has changed
* (½ point) Error message is appropriate

**Table 2 Model**

* (½ point) Functions listed alphabetically
* (½ point) All functions appropriately named
* (½ point) All functions include appropriate description
* (½ point) Delete query is correct (uses ID)
* (½ point) Parameter binding correct
* (½ point) getDetails query is correct (uses ID)
* (½ point) Parameter binding is correct
* (½ point) fetch used to store one record in results
* (½ point) getList query retrieves appropriate fields (ID + )
* (½ point) Where clause includes filter parameter
* (½ point) List sorted
* (½ point) fetchAll used to define results
* (½ point) Insert query lists all fields except ID
* (½ point) Formatted fields unformatted
* (½ point) Checkbox values converted to true/false
* (½ point) Query includes parameters for all fields
* (½ point) Parameter bindings are correct
* (½ point) Optional fields set to null when appropriate
* (½ point) isDuplicate function includes logical key parameters
* (½ point) Query counts number of records that match logical keys (parameters)
* (½ point) Parameter bindings are correct
* (½ point) fetch used to retrieve number of matching records
* (½ point) Function returns Boolean value (count > 0)
* (½ point) Update query sets field types for all but ID fields
* (½ point) Formatted fields unformatted
* (½ point) Checkbox values converted to true/false
* (½ point) Where clause designates ID (parameter)
* (½ point) Query includes parameters for all fields
* (½ point) Parameter bindings are correct
* (½ point) Optional fields set to null when appropriate
* (1 point) validateInputs function correctly validates all inputs
* (½ point) Includes code to check for duplicate records
* (½ point) isDuplicate only called if no previous errors and logical key(s) has changed
* (½ point) Error message is appropriate

**Table 1 Details View**

* (1 point) Form appearance is professional
* (1 point) Form includes header and footer
* (1 point) Form imports necessary JavaScript and CSS files
* (½ point) Form handles empty errors array
* (½ point) Form determines if this is a new record
* (½ point) Action variable set appropriately (new or update)
* (1 point) Each field determines if a previous value exists and displays it if it does
* (1 point) Each (appropriate) field includes an error marker that displays if the errors array contains the field name
* (1 point) Logical key fields remember their original values
* (½ point) Combo box correctly filled from data array
* (½ point) Combo box selected option set correctly
* (½ point) Delete button hidden for new records (flag defined)

**Table 1 List View**

* (1 point) Form appearance is professional
* (1 point) Form includes header and footer
* (½ point) Form includes a New button
* (½ point) New button includes appropriate action
* (½ point) Form includes a search text box and button
* (½ point) Search button includes appropriate action
* (½ point) Look up list links include appropriate action and ID number

**Table 2 Details View**

* (1 point) Form appearance is professional
* (1 point) Form includes header and footer
* (1 point) Form imports necessary JavaScript and CSS files
* (½ point) Form handles empty errors array
* (½ point) Form determines if this is a new record
* (½ point) Action variable set appropriately (new or update)
* (1 point) Each field determines if a previous value exists and displays it if it does
* (1 point) Each (appropriate) field includes an error marker that displays if the errors array contains the field name
* (1 point) Logical key fields remember their original values
* (½ point) Combo box correctly filled from data array
* (½ point) Combo box selected option set correctly
* (½ point) Delete button hidden for new records (flag defined)

**Table 2 List View**

* (1 point) Form appearance is professional
* (1 point) Form includes header and footer
* (½ point) Form includes a New button
* (½ point) New button includes appropriate action
* (½ point) Form includes a search text box and button
* (½ point) Search button includes appropriate action
* (½ point) Look up list links include appropriate action and ID number