

Carl Crivell Ganareal

Salvador Phillip Uriel

John Vincent Luciano

“CRUD AJAX JQUERY ACTIVITY – Documentation”

Student Records

+ Add New Student

Show

10

 entries

Search:

Id	Name	Email	Username	Action
8	Phillip	1111	BSIT	<div><div></div><div></div></div>

Showing 1 to 1 of 1 entries

Previous

1

Next

Warning!

This area is for use by developers only. Scammers have been known to encourage people to copy/paste information here to hack accounts. Do not proceed if you are unsure.

DATA Deleted

index.php:284

Uncaught TypeError: Cannot read properties of undefined (reading 'toLowerCase')

at HTMLDocument.eval (eval at <anonymous> (eval at ExBtSrcCd (eval at <anonymous> (name1.js:96:4016))), <anonymous>:1:58487)

DATA Inserted

index.php:140

Console

Issues

Include third-party cookie issues

Audit usage of navigator.userAgent, navigator.appVersion, and navigator.platform

```
<script type="text/javascript">
1  $(document).ready(function(){
2    showAllName();
3    //View Record
4    function showAllName(){
5      $.ajax({
6        url : "config/dbsend.php",
7        type : "POST",
8        data : {action:"view"},
9        success:function(response){
10          $("#tableData").html(response); //shows all Student in table at the foodadd php file and passes it to a div
11          class named table
12          $(".table").DataTable({
13            order:[0, 'DESC']
14          });
15        }
16      });
17    }
18    //insert ajax request data
19    $("#submit").click(function(e){
20      if ($("#formData")[0].checkValidity()) { //if submit is clicked the form is checked and is then redirected to a url
21        in form data that allows it to pass the
22        e.preventDefault(); //values to a sql query and then hides the modal automatically
23        $.ajax({
24          url : "config/dbsend.php",
25          type : "POST",
26          data : $("#formData").serialize()+"&action=insert",
27          success:function(response){
28            Swal.fire({
29              icon: 'success',
30              title: 'Student added successfully',
31            });
32            console.log("DATA Inserted");
33            $("#addModal").modal('hide');
34            $("#formData")[0].reset();
35            showAllName();//reloads the table
36          }
37        });
38      }
39    });
40  });
41  });
```

```

<!-- Modal Header -->
<div class="modal-header">
  <h4 class="modal-title">Add New Student</h4>
  <button type="button" class="close" data-dismiss="modal">&times;</button>
</div>
<!-- Modal body -->
<div class="modal-body">
  <form id="formData">
    <div class="form-group">
      <label for="name">Name:</label>
      <input type="text" class="form-control" name="name" placeholder="Enter Name" required="">
    </div>
    <div class="form-group">
      <label for="email">Age:</label>
      <input type="number" class="form-control" name="Age" placeholder="Enter Age" required="">
    </div>
    <div class="form-group">
      <label for="username">Course:</label>
      <input type="text" class="form-control" name="course" placeholder="Enter Course" required="">
    </div>

    <hr>
    <div class="form-group float-right">
      <button type="submit" class="btn btn-success" id="submit">Submit</button>
      <button type="button" class="btn btn-danger" data-dismiss="modal">Close</button>
    </div>
  </form>
</div>
</div>
</div>
div>

```

```

$dbObj = new Database();
// Insert Record
if (isset($_POST['action']) && $_POST['action'] == "insert") {
  $name = $_POST['name'];
  $age = $_POST['Age'];
  $course = $_POST['course'];

  $dbObj->insertRecord($name, $age, $course);
}
// View record
if (isset($_POST['action']) && $_POST['action'] == "view") {
  $output = "";
  $students = $dbObj->displayRecord();
  if ($dbObj->totalRowCount() > 0) {
    $output .= "<table class='table table-striped table-hover'>
      <thead>
        <tr>
          <th>Id</th>
          <th>Name</th>
          <th>Email</th>
          <th>Username</th>

          <th>Action</th>
        </tr>
      </thead>
      <tbody>";

    foreach ($students as $student) {
      $output .= "<tr>
        <td>".$student['id']. "</td>
        <td>".$student['studname']. "</td>
        <td>".$student['studage']. "</td>
        <td>".$student['studcourse']. "</td>

        <td>
          <a href='#editModal' style='color:green' data-toggle='modal'
            class='editBtn' id='".$student['id']."'><i class='fa fa-pencil'></i></a>&nbsp;
          <a href=''" style='color:red' class='deleteBtn' id='".$student['id']."'>
            <i class='fa fa-trash' ></i></a>
        </td>
      </tr>";
    }
  }
}

```

```
// Insert customer data into customer table
public function insertRecord($name, $age, $course)
{
    $sql = "INSERT INTO $this->studentstbl (studname, studage, studcourse) VALUES('$name', '$age', '$course')";
    $query = $this->con->query($sql);
    if ($query) {
        return true;
    }else{
        return false;
    }
}

public function displayRecord()
{
    $sql = "SELECT * FROM $this->studentstbl"; //takes the number of rows in the database and calls upon the associated
    data for each row through using while                                     giving initializing it
    $query = $this->con->query($sql); //as data
    $data = array();
    if ($query->num_rows > 0) {
        while ($row = $query->fetch_assoc()) {
            $data[] = $row;
        }
        return $data;
    }else{
        return false;
    }
}
}
```

Creates a table that takes all the value in the database and through ajax and jquery displays the table in order to show all of the students.

Inserting students passes the data from the form to DBSend which is then sent to DBconfig as a query then at the end of the ajax runs the table again to refresh the data. This way of inserting data is OOP and ajax at the same time we are using serialize to run through the data note if you are planning to add a picture to the form avoid using serialize.

The screenshot displays a web application interface for managing student records. On the left, a table titled 'Student Records' shows a list of students with columns for 'Id' and 'Name'. The first entry is '8' and 'Phillip'. Below the table, it says 'Showing 1 to 1 of 1 entries'. A modal window titled 'Edit Student' is open in the center, allowing users to update a student's information. The modal contains three input fields: 'Name' (with placeholder 'Enter Name'), 'Age' (with placeholder 'Enter Age'), and 'Course' (with placeholder 'Enter Course'). At the bottom of the modal are 'Update' and 'Close' buttons. On the right side of the screenshot, a browser's developer console is visible, showing a warning message and two log entries. The first log entry is 'DATA Deleted' with a red error icon, indicating an 'Uncaught TypeError: Cannot read properties of undefined (reading 'toLowerCase')'. The second log entry is 'DATA Inserted' showing a successful database update for the student with ID 8, name 'Phillip', age 1111, and course 'BSIT'.

Student Records

+ Add New Student

Show 10 entries

Search:

Id	Name	Email	Username	Action
8	Carl Crivell Ganareal	21	BSIT	<div></div>

Showing 1 to 1 of 1 entries

Previous1Next

Warning!

This area is for use by developers only. Scammers have been known to encourage people to copy/paste information here to hack accounts. Do not proceed if you are unsure.

DATA Deleted

index.php:284

Uncaught TypeError: Cannot read properties of undefined (reading 'toLowerCase') at HTMLDocument.eval (eval at ExRtSvrCd (eval at <anonymous> (panel.js:96:4916))), <anonymous>:1:58487) VM137:1

DATA Inserted

index.php:140

({"id":"8","studname":"Phillip","studage":"1111 index.php:162","studcourse":"BSIT"}) VM137:1

Uncaught TypeError: Cannot read properties of undefined (reading 'toLowerCase') at HTMLDocument.eval (eval at ExRtSvrCd (eval at <anonymous> (panel.js:96:4916))), <anonymous>:1:58487) VM137:1

DATA UPDATED

index.php:174

Console

Issues

Include third-party cookie issues

Audit usage of navigator.userAgent, navigator.appVersion, and navigator.platform

```
<!-- Edit Record Modal -->
<div class="modal" id="editModal">
  <div class="modal-dialog">
    <div class="modal-content">
      <!-- Modal Header -->
      <div class="modal-header">
        <h4 class="modal-title">Edit Student</h4>
        <button type="button" class="close" data-dismiss="modal">&times;</button>
      </div>
      <!-- Modal body -->
      <div class="modal-body">
        <form id="EditformData">
          <input type="hidden" name="id" id="edit-form-id">
          <div class="form-group">
            <label for="name">Name:</label>
            <input type="text" class="form-control" name="name" id="name" placeholder="Enter Name" required="">
          </div>
          <div class="form-group">
            <label for="email">Age:</label>
            <input type="number" class="form-control" name="Age" id="Age" placeholder="Enter Age" required="">
          </div>
          <div class="form-group">
            <label for="username">Course:</label>
            <input type="text" class="form-control" name="course" id="course" placeholder="Enter Course" required="">
          </div>
          <hr>
          <div class="form-group float-right">
            <button type="submit" class="btn btn-primary" id="update">Update</button>
            <button type="button" class="btn btn-danger" data-dismiss="modal">Close</button>
          </div>
        </form>
      </div>
    </div>
  </div>
</div>
</div>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
```

```

3 //Edit Record
4 $("body").on("click", ".editBtn", function(e){
5     e.preventDefault();
6     var editId = $(this).attr('id');
7     $.ajax({
8         url : "config/dbsend.php",
9         type : "POST",
10        data : {editId:editId},
11        success:function(response){
12            var data = JSON.parse(response);
13            $("#edit-form-id").val(data.id); //edit record takes the json to autofill the update modal
14            $("#name").val(data.NameName);
15            $("#Age").val(data.NameAge);
16            $("#course").val(data.Course);
17            console.log(response);
18        }
19    });
20    $("#update").click(function(e){
21        if ($("#EditformData")[0].checkValidity()) {
22            e.preventDefault();
23            $.ajax({
24                url : "config/dbsend.php",
25                type : "POST",
26                data : $("#EditformData").serialize()+"&action=update", //uses the record to get the id to call the update route in
27                //foodadd to get the sql function
28                success:function(response){
29                    console.log("DATA UPDATED");
30                    Swal.fire({
31                        icon: 'success',
32                        title: 'Student updated successfully',
33                    });
34                    $("#editModal").modal('hide');
35                    $("#EditformData")[0].reset();
36                    showAllName();
37                }
38            });
39        }
40    });
41 }

```

```

}
// Edit Record
if (isset($_POST['editId'])) {
    $editId = $_POST['editId'];
    $row = $dbObj->getRecordById($editId);
    echo json_encode($row);
}

if (isset($_POST['action']) && $_POST['action'] == "update") {
    $id = $_POST['id'];
    $name = $_POST['name'];
    $age = $_POST['Age'];
    $course = $_POST['course'];

    $dbObj->updateRecord($id, $name, $age, $course,);
}

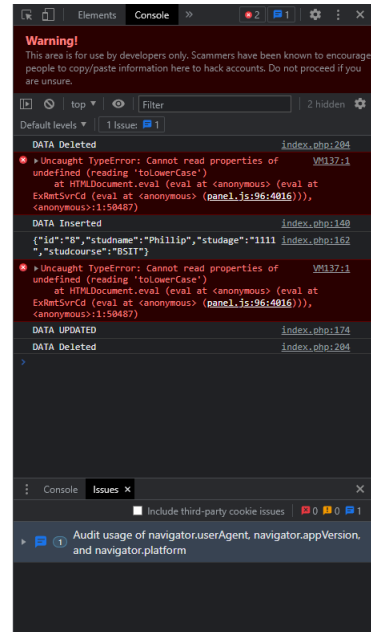
```

Using the ID of the row the data from the database is sent to the form and automatically calls the id this is done so that when we enter the data only the said row would be affected.

Student Records

+ Add New Student

No records found



```
if (isset($_POST['deleteBtn'])) {  
    $deleteBtn = $_POST['deleteBtn'];  
    $dbObj->deleteRecord($deleteBtn);  
}
```

```
$( "body" ).on( "click", ".deleteBtn", function( e ) {  
    e.preventDefault();  
    var tr = $( this ).closest( 'tr' );  
    var deleteBtn = $( this ).attr( 'id' );  
    if ( confirm( 'Are you sure want to delete this Record' ) ) { //gets the id to place the delete action to the delete function  
        $.ajax({  
            url : "config/dbsend.php",  
            type : "POST",  
            data : {deleteBtn:deleteBtn},  
            success: function( response ) {  
                tr.css( 'background-color', '#ff6565' );  
                console.log( "DATA Deleted" );  
                Swal.fire({  
                    icon: 'success',  
                    title: 'Student delete successfully',  
                });  
                showAllName();  
            }  
        });  
    }  
});
```

In the same principle delete is called through the use of ID to run it to a query to delete the row. And then we reset the table to show changes.