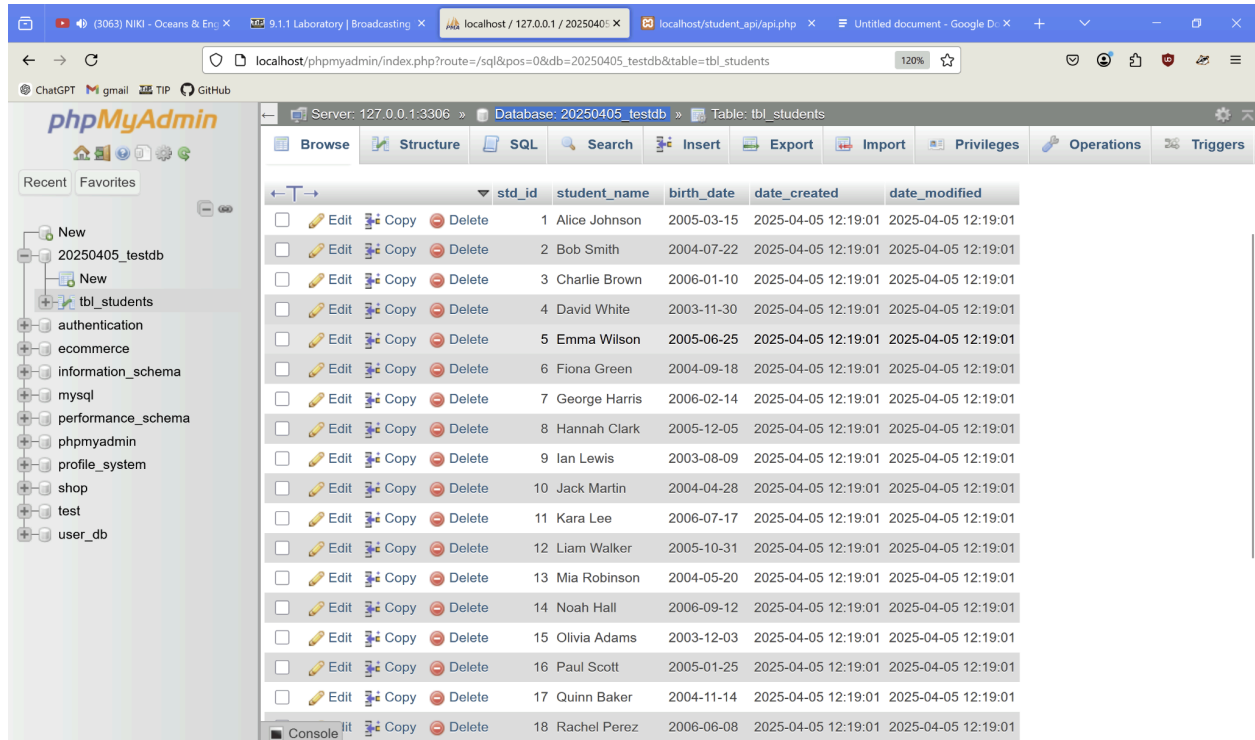


NAVAREZ, JOSHUA B.

CIT 403-IT42S2 - Systems Integration and Architecture 2

9.1.1 Laboratory | Broadcasting Web Services with Git Version Control

## TEST\_DB

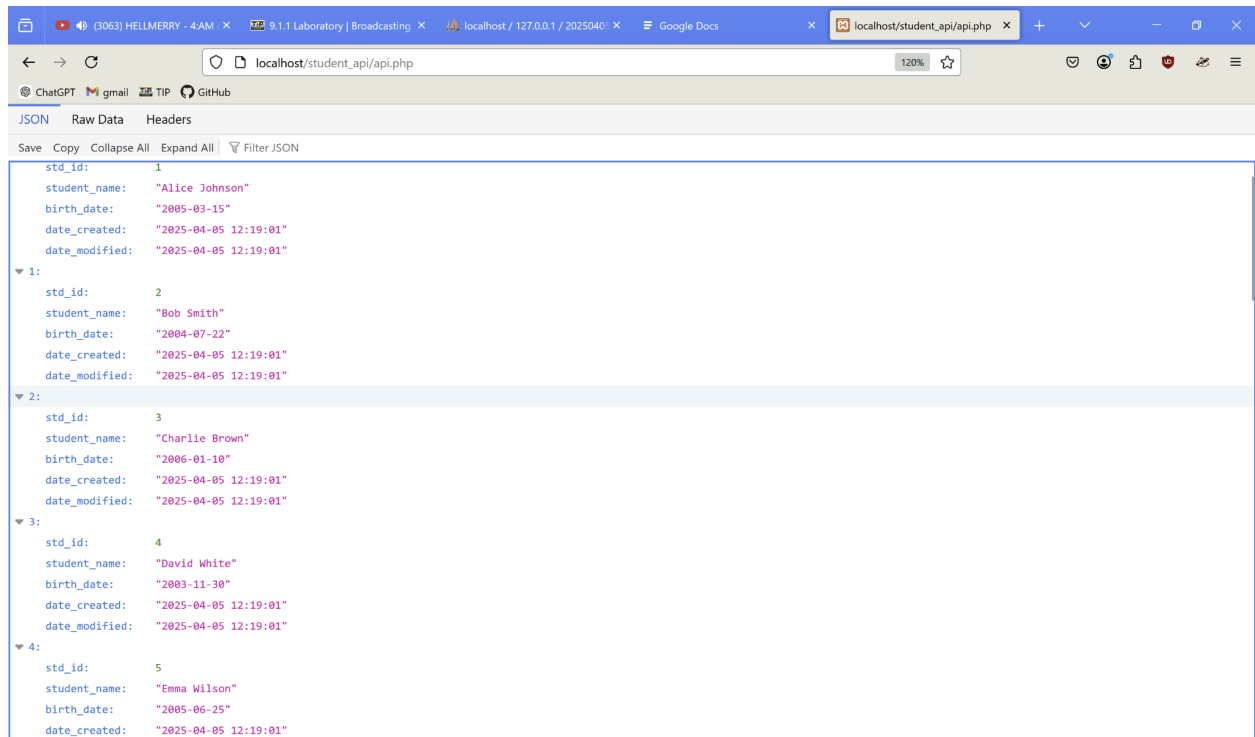


	std_id	student_name	birth_date	date_created	date_modified
<input type="checkbox"/>	1	Alice Johnson	2005-03-15	2025-04-05 12:19:01	2025-04-05 12:19:01
<input type="checkbox"/>	2	Bob Smith	2004-07-22	2025-04-05 12:19:01	2025-04-05 12:19:01
<input type="checkbox"/>	3	Charlie Brown	2006-01-10	2025-04-05 12:19:01	2025-04-05 12:19:01
<input type="checkbox"/>	4	David White	2003-11-30	2025-04-05 12:19:01	2025-04-05 12:19:01
<input type="checkbox"/>	5	Emma Wilson	2005-06-25	2025-04-05 12:19:01	2025-04-05 12:19:01
<input type="checkbox"/>	6	Fiona Green	2004-09-18	2025-04-05 12:19:01	2025-04-05 12:19:01
<input type="checkbox"/>	7	George Harris	2006-02-14	2025-04-05 12:19:01	2025-04-05 12:19:01
<input type="checkbox"/>	8	Hannah Clark	2005-12-05	2025-04-05 12:19:01	2025-04-05 12:19:01
<input type="checkbox"/>	9	Ian Lewis	2003-08-09	2025-04-05 12:19:01	2025-04-05 12:19:01
<input type="checkbox"/>	10	Jack Martin	2004-04-28	2025-04-05 12:19:01	2025-04-05 12:19:01
<input type="checkbox"/>	11	Kara Lee	2006-07-17	2025-04-05 12:19:01	2025-04-05 12:19:01
<input type="checkbox"/>	12	Liam Walker	2005-10-31	2025-04-05 12:19:01	2025-04-05 12:19:01
<input type="checkbox"/>	13	Mia Robinson	2004-05-20	2025-04-05 12:19:01	2025-04-05 12:19:01
<input type="checkbox"/>	14	Noah Hall	2006-09-12	2025-04-05 12:19:01	2025-04-05 12:19:01
<input type="checkbox"/>	15	Olivia Adams	2003-12-03	2025-04-05 12:19:01	2025-04-05 12:19:01
<input type="checkbox"/>	16	Paul Scott	2005-01-25	2025-04-05 12:19:01	2025-04-05 12:19:01
<input type="checkbox"/>	17	Quinn Baker	2004-11-14	2025-04-05 12:19:01	2025-04-05 12:19:01
<input type="checkbox"/>	18	Rachel Perez	2006-06-08	2025-04-05 12:19:01	2025-04-05 12:19:01

The Provided api code

```
1 <?php
2 header("Content-Type: application/json");
3 $servername = "localhost";
4 $username = "root";
5 $password = "";
6 $dbname = "20250405_testdb";
7 $conn = new mysqli($servername, $username, $password, $dbname);
8 if ($conn->connect_error) {
9     die(json_encode(["error" => "Database connection failed: " . $conn->connect_error]));
10 }
11 $std_id = isset($_GET['std_id']) ? intval($_GET['std_id']) : 0;
12 if ($std_id > 0) {
13     $stmt = $conn->prepare("SELECT * FROM tbl_students WHERE std_id = ?");
14     $stmt->bind_param("i", $std_id);
15     $stmt->execute();
16     $result = $stmt->get_result();
17     $student = $result->fetch_assoc();
18     $stmt->close();
19     $conn->close();
20     echo json_encode($student ? $student : ["error" => "Student not found"], JSON_PRETTY_PRINT);
21 } else {
22     $stmt = $conn->prepare("SELECT * FROM tbl_students");
23     $stmt->execute();
24     $result = $stmt->get_result();
25     $students = [];
26     while ($row = $result->fetch_assoc()) {
27         $students[] = $row;
28     }
29     $stmt->close();
30     $conn->close();
31     echo json_encode($students, JSON_PRETTY_PRINT);
32 }
33 ?>
```

http://localhost/student\_api/api.php

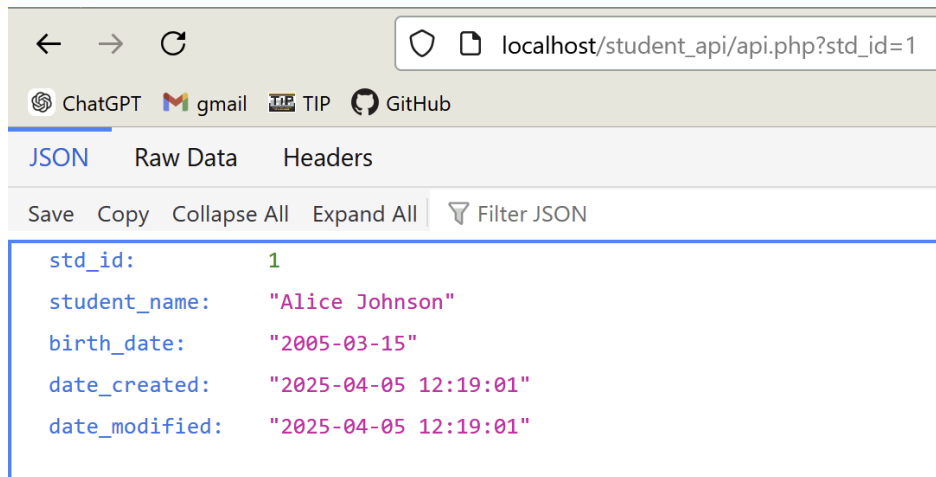


localhost/student\_api/api.php

JSON Raw Data Headers

Save Copy Collapse All Expand All Filter JSON

```
{
  "std_id": 1,
  "student_name": "Alice Johnson",
  "birth_date": "2005-03-15",
  "date_created": "2025-04-05 12:19:01",
  "date_modified": "2025-04-05 12:19:01",
  "1": {
    "std_id": 2,
    "student_name": "Bob Smith",
    "birth_date": "2004-07-22",
    "date_created": "2025-04-05 12:19:01",
    "date_modified": "2025-04-05 12:19:01",
    "2": {
      "std_id": 3,
      "student_name": "Charlie Brown",
      "birth_date": "2006-01-10",
      "date_created": "2025-04-05 12:19:01",
      "date_modified": "2025-04-05 12:19:01",
      "3": {
        "std_id": 4,
        "student_name": "David White",
        "birth_date": "2003-11-30",
        "date_created": "2025-04-05 12:19:01",
        "date_modified": "2025-04-05 12:19:01",
        "4": {
          "std_id": 5,
          "student_name": "Emma Wilson",
          "birth_date": "2005-06-25",
          "date_created": "2025-04-05 12:19:01",
          "date_modified": "2025-04-05 12:19:01"
        }
      }
    }
  }
}
```

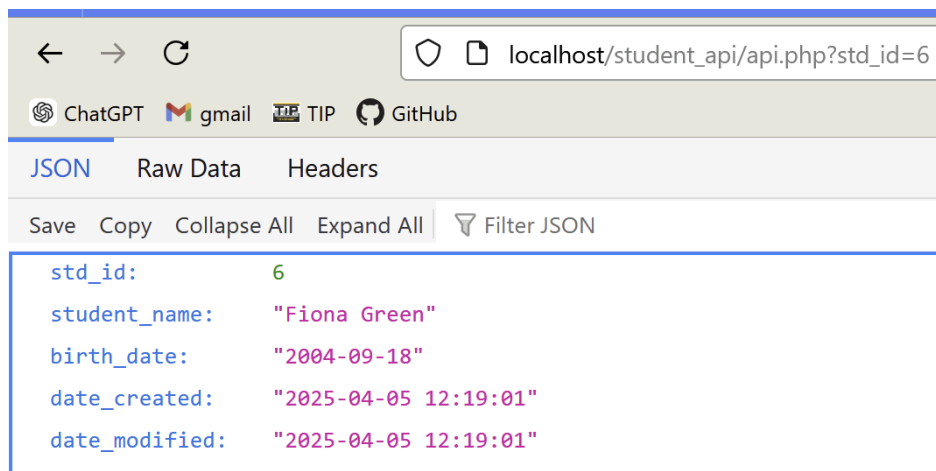


localhost/student\_api/api.php?std\_id=1

JSON Raw Data Headers

Save Copy Collapse All Expand All Filter JSON

```
{
  "std_id": 1,
  "student_name": "Alice Johnson",
  "birth_date": "2005-03-15",
  "date_created": "2025-04-05 12:19:01",
  "date_modified": "2025-04-05 12:19:01"
}
```



localhost/student\_api/api.php?std\_id=6

JSON Raw Data Headers

Save Copy Collapse All Expand All Filter JSON

```
{
  "std_id": 6,
  "student_name": "Fiona Green",
  "birth_date": "2004-09-18",
  "date_created": "2025-04-05 12:19:01",
  "date_modified": "2025-04-05 12:19:01"
}
```

Reflection Section (20 pts):

Write your insights based on the following questions:

1. What is the purpose of PHP code `header("Content-Type: application/json");`
  - sending a raw HTTP header
2. If Part 1, Step C was skipped or neglected to execute, what will be the output in Part 1, Step D using `"http://localhost/student_api/api.php?std_id=1"` URL

Nothing will appear