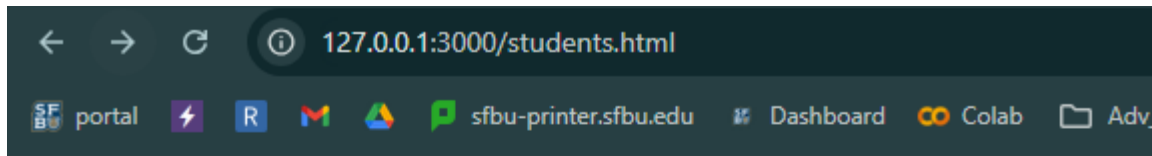


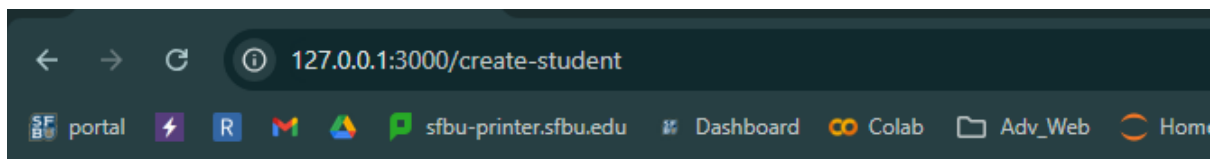
Q1.

Output:



Name ?

GPA ?



~ inserted student with ID: 793

**Code:**

```
<> students.html X JS server.js Q1.sql
Exam > Q1 > <> students.html > ...
1  <!DOCTYPE html>
2  <html lang="en">
3
4  <head>
5      <meta charset="UTF-8">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7      <title>Document</title>
8  </head>
9
10 <body>
11     <form method="post" action="/create-student">
12         <p>
13             <label>Name ?</label>
14             <input type="text" name="name">
15         </p>
16         <p>
17             <label>GPA ?</label>
18             <input type="text" name="gpa">
19         </p>
20         <input type="submit" value="save">
21     </form>
22 </body>
23
24 </html>
25
```

<> students.html JS server.js X Q1.sql

Exam > Q1 > JS server.js > app.get("") callback

```
1  const mysql = require('mysql2')
2  const express = require('express');
3  const app = express()
4
5  const conn = mysql.createConnection({
6    host: "localhost",
7    user: 'root',
8    password: 'root',
9    database: 'FinalExamDB'
10 })
11
12 // Send file
13 app.get('', (req, res) => {
14   // res.send('This is the home page')
15   res.sendFile(path.join(__dirname, 'students.html'))
16 })
17
18 app.use(express.urlencoded({ extended: true }))
19 app.use(express.static(__dirname))
20
21 // get-students call
22 app.get('/get-students', (req, resp) => {
23   conn.query('select * from Students',
24     function (err, rows) {
25       if (err) {
26         console.log("app.get ~ err:", err)
27       } else {
28         rows.forEach(element => {
29           console.log("app.get ~ element:", element)
30         });
31       }
32     }
33   })
34 })
--
```

students.html
JS server.js
Q1.sql

Exam > Q1 > JS server.js > app.get("") callback

```

36 // get-students-by-gpa call
37 app.get('/get-students-by-gpa', (req, resp) => {
38   const minGPA = req.query.min;
39   const maxGPA = req.query.max;
40
41   conn.query('select * from Students where GPA>? and GPA<? order by GPA DESC', [minGPA, maxGPA],
42     function (err, rows) {
43       if (err) {
44         console.log("app.get ~ err:", err)
45       } else {
46         console.log("app.get ~ rows:", rows)
47         let html = '<ol>'
48         for (let i = 0; i < rows.length; i++) {
49           const element = rows[i];
50           html += '<li> ${element.StudentName} = ${element.GPA} </li>';
51         }
52         html += '</ol>'
53         resp.send(html)
54       }
55     })
56 })
57
58 // create-student call
59 app.post('/create-student', function (req, res) {
60   const student = req.body;
61   console.log('from data', student);
62
63   conn.query('insert into Students(StudentName,GPA) values(?,?)',
64     [student.name, student.gpa], function (err, result) {
65       console.log("~ result:", result)
66       if (err) {
67         console.log("~ err:", err)
68       } else {
69         res.send('~ inserted student with ID: ${result.insertId}')
70       }
71     })
72 })

```

students.html
JS server.js
Q1.sql

Exam > Q1 > JS server.js > app.get("") callback

```

59 app.post('/create-student', function (req, res) {
60   [student.name, student.gpa], function (err, result) {
71   })
72 })
73
74 // app.get('/', (req, res) =>{
75 //   res.sendFile(path.join(__dirname, 'expl.html'))
76 // })
77
78 app.listen(3000, () => { console.log('server is up'); })
79

```

## Terminal:

```
PS C:\Users\abhi\OneDrive\Desktop\Masters\Sem_2\3_CS526 - Advanced Web Programming\Final\Exam\Exam\Q1> node .\server.js
server is up
from data { name: 'abhi', gpa: '4.0' }
~ result: ResultSetHeader {
  fieldCount: 0,
  affectedRows: 1,
  insertId: 792,
  info: '',
  insertId: 792,
  info: '',
  serverStatus: 2,
  warningStatus: 0,
  changedRows: 0
}
```

## Database:

students.html

JS server.js

Q1.sql

X

Exam > Q1 > Q1.sql

```
1  create database FinalExamDB;
2
3  CREATE TABLE `FinalExamDB`.`Students` (
4    `StudID` INT NOT NULL AUTO_INCREMENT,
5    `StudentName` VARCHAR(45) NULL,
6    `GPA` FLOAT NULL,
7    PRIMARY KEY (`StudID`));
8
9  INSERT INTO `FinalExamDB`.`Students` (`StudID`, `StudentName`, `GPA`) VALUES ('123', 'Susan', '3.1');
10 INSERT INTO `FinalExamDB`.`Students` (`StudID`, `StudentName`, `GPA`) VALUES ('456', 'Billy', '2.5');
11 INSERT INTO `FinalExamDB`.`Students` (`StudID`, `StudentName`, `GPA`) VALUES ('777', 'Maria', '3.3');
12 INSERT INTO `FinalExamDB`.`Students` (`StudID`, `StudentName`, `GPA`) VALUES ('789', 'Alice', '4.0');
13
```

15 • SELECT \* FROM `FinalExamDB`.`students`;

16

Result Grid

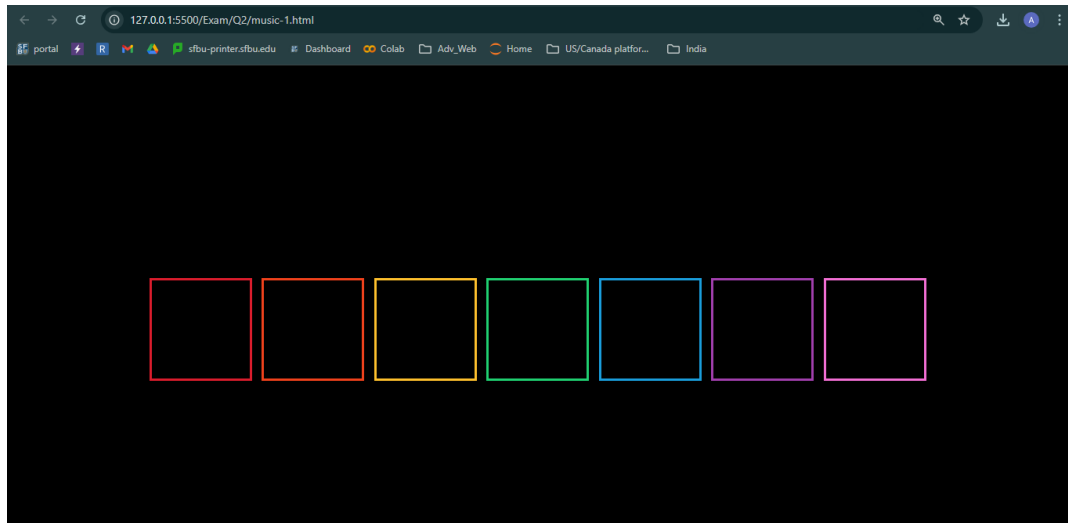
Filter Rows:

Edit: Export/Import: Wrap Cell Content:

	StudID	StudentName	GPA
	123	Susan	3.1
	456	Billy	2.5
	777	Maria	3.3
	789	Alice	4
▶	793	abhi	4
*	NULL	NULL	NULL

Q2.

Output:



Code:

```
music-1.html x library.html students.html
Exam > Q2 > music-1.html > html > body > div#instrument > div#c.box > audio
1 <!doctype html>
2 <html>
3
4 <head>
5   <meta charset="utf-8">
6   <title>Musical Boxes</title>
7   <!-- Write the code to link the css and js file -->
8   <link rel="stylesheet" href="music.css">
9   <script src="music.js"></script>
10 </head>
11
12 <body>
13   <!-- Create audio elements to play the audio
14   You will have to create as many audio elements as there are musical notes
15   using audio tag to play them . The source's src attribute can be used to refer to the
16   audio file -->
17   <div id="instrument">
18     <div id="c" class="box" onclick="fun3()">
19       <audio src="c_note.mp3"></audio>
20     </div>
21     <div id="d" class="box" onclick="fun4()">
22       <audio src="d_note.mp3"></audio>
23     </div>
24     <div id="e" class="box" onclick="fun5()">
25       <audio src="e_note.mp3"></audio>
26     </div>
27     <div id="f" class="box" onclick="fun6()">
28       <audio src="f_note.mp3"></audio>
29     </div>
30     <div id="g" class="box" onclick="fun7()">
31       <audio src="g_note.mp3"></audio>
32     </div>
33     <div id="a" class="box" onclick="fun1()">
34       <audio src="a_note.mp3"></audio>
35     </div>
36     <div id="b" class="box" onclick="fun2()">
37       <audio src="b_note.mp3"></audio>
38     </div>
39   </div>
40 </body>
41
42 </html>
```

# music.css x JS music.js

Exam > Q2 > # music.css > ...

```
1  @charset "utf-8";
2  /* CSS Document */
3
4  body{background-color:■black;}
5
6  #instrument{
7      height:116px;
8      width:812px;
9      padding-right:10px;
10     padding-top:50px;
11     margin:auto;
12     position:absolute;
13     top:0; left:0; bottom:0; right:0;
14 }
15 .box{
16     width:100px;
17     height:100px;
18     margin-left:10px;
19     margin-top:10px;
20     float:left;
21 }
22 #c{
23     /* code goes here to set the color*/
24     border:3px solid ■#db1d2d;
25 }
26 /* Code goes here for changing the color and pointer after hovering the mouse */
27 #c:hover{
28     background-color: ■#661920;
29 }
30 /* Code goes here for the color during active state */
31 #c:active{
32     background-color: ■#db1d2d;
33 }
34
35
36 #d{
37     /* code goes here to set the color*/
38     border:3px solid ■#f0421c;
39 }
40 /* Code goes here for changing the color and pointer after hovering the mouse */
41 #d:hover{
42     background-color: ■#6e2819;
43 }
44 /* Code goes here for the color during active state */
45 #d:active{
46     background-color: ■#f0421c;
47 }
```

# music.css x JS music.js

Exam > Q2 > # music.css > ...

```
47 }
48
49
50 #e{
51     /* code goes here to set the color */
52     border:3px solid #fec02d;
53 }
54 /* Code goes here for changing the color and pointer after hovering the mouse */
55 #e:hover{
56     background-color: #735b20;
57 }
58 /* Code goes here for the color during active state */
59 #e:active{
60     background-color: #fec02d;
61 }
62
63
64
65 #f{
66     /* code goes here to set the color*/
67     border:3px solid #20d071;
68 }
69 /* Code goes here for changing the color and pointer after hovering the mouse */
70 #f:hover{
71     background-color: #1b613b;
72 }
73 /* Code goes here for the color during active state */
74 #f:active{
75     background-color: #20d071;
76 }
77
78
79
80 #g{
81     /* code goes here to set the color */
82     border:3px solid #1a9ddb;
83 }
84 /* Code goes here for changing the color and pointer after hovering the mouse */
85 #g:hover{
86     background-color: #184d65;
87 }
88 /* Code goes here for the color during active state */
89 #g:active{
90     background-color: #184d65;
91 }
92
```



```
92
93
94
95 #a{
96     /* code goes here to set the color*/
97     border:3px solid #a13fad;
98 }
99 /* Code goes here for changing the color and pointer after hovering the mouse */
100 #a:hover{
101     background-color: #4e2753;
102 }
103 /* Code goes here for the color during active state */
104 #a:active{
105     background-color: #a13fad;
106 }
107
108
109
110 #b{
111     /* code goes here to set the color*/
112     border:3px solid #f26fd4;
113 }
114 /* Code goes here for changing the color and pointer after hovering the mouse */
115 #b:hover{
116     background-color: #6f3a62;
117 }
118 /* Code goes here for the color during active state */
119 #b:active{
120     background-color: #f26fd4;
121 }
122
```

Exam &gt; Q2 &gt; JS music.js &gt; fun6

```
1  /* This psuedocode is provided just to help you.
2  You can implement it using your way as well
3  Psuedocode for defining the function:
4  1. Access the audio element on the web page with its ID and store it in a variable .
5  2. Set the current playback time of the variable's audio element to 0 second using .currentTime
6  |   This sets the currentTime property of variable to 0, which rewinds the audio to the start
7  3. Play the audio element using play (.play() )function.
8  4. Test it and then implement other functions the same way.
9  */
10
11
12  function fun3(){
13  |   note_c = document.getElementById('c');
14  |   document.addEventListener().play()
15  | }
16
17  function fun4(){
18  |   note_d = document.getElementById('d')
19  | }
20
21  function fun5(){
22  |   note_e = document.getElementById('e')
23  | }
24
25  function fun6(){
26  |   note_f = document.getElementById('f')
27  | }
28
29  function fun7(){
30  |   note_g = document.getElementById('g')
31  | }
32
33  function fun1(){
34  |   note_a = document.getElementById('a')
35  | }
36
37  function fun2(){
38  |   note_b = document.getElementById('b')
39  | }
40
41
```