Cloud based enterprise systems lab (15B28CI581)

Assignment week 8

Patil Amit Gurusidhappa
19104004
B11

1. **Create the following student table in SQLite/MySQL**

Student(Enroll Number,
Fname varchar (20),
Middle varchar (20),
Lname varchar (20),
Course varchar (20) default value B.Tech.,
Branch varchar (20) default value IT,
Mobile number(10),
DoB date,
Address varchar(20),
City varchar (20),
State varchar (20),
Country varchar (20) default value India)

Connecting to MYSQL Server through XAMPP

```
C:\Users\Amit>mysql.exe -u root --password
Enter password: ****
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 13
Server version: 10.4.22-MariaDB mariadb.org binary distribution
```

Create Table request to server

List of Available Databases

```
+--------------------+
6 rows in set (0.084 sec)

MariaDB [(none)]> create database studentdb
    -> ;
Query OK, 1 row affected (0.015 sec)

MariaDB [(none)]> show databases;
+--------------------+
| Database           |
+--------------------+
| gibbondatabase     |
| information_schema |
| mysql              |
| performance_schema |
| phpmyadmin         |
| studentdb          |
| test               |
+--------------------+
7 rows in set (0.002 sec)

MariaDB [(none)]>
```

Running server on port **3000**

```
Example app listening on port 3000
table Creat Success [object Object]
```

After the successful request execution table is created

```
ERROR 1146 (42S02): Table 'studentdb.student' doesn't exist
MariaDB [studentdb]> describe student;
+---------+-------------+------+-----+---------+-------+
| Field   | Type        | Null | Key | Default | Extra |
+---------+-------------+------+-----+---------+-------+
| title   | varchar(255)| NO   |     | NULL    |       |
| Fname   | varchar(20) | YES  |     | NULL    |       |
| Middle  | varchar(20) | YES  |     | NULL    |       |
| Lname   | varchar(20) | YES  |     | NULL    |       |
| Course  | varchar(20) | YES  |     | BTech   |       |
| Branch  | varchar(20) | YES  |     | IT      |       |
| Mobile  | int(10)     | YES  |     | NULL    |       |
| DoB     | date        | YES  |     | NULL    |       |
| Address | varchar(20) | YES  |     | NULL    |       |
| City    | varchar(20) | YES  |     | NULL    |       |
| State   | varchar(20) | YES  |     | NULL    |       |
| Country | varchar(20) | YES  |     | India   |       |
+---------+-------------+------+-----+---------+-------+
12 rows in set (0.064 sec)
```

Code

```
app.get('/createTB', (req, res) => {
    let mysql = require('mysql');
    let connection = mysql.createConnection({
        host: 'localhost',
        user: 'root',
        password: 'root',
        database: 'studentdb'
    });

    // connect to the MySQL server
    connection.connect(function (err) {
        if (err) {
            return console.error('error: ' + err.message);
        }

        let createStudent = `create table if not exists student(
```

```
                            title varchar(255) not null,
                            Fname varchar(20),
                            Middle varchar (20),
                            Lname varchar (20),
                            Course varchar (20) default "BTech",
                            Branch varchar (20) default "IT",
                            Mobile int(10),
                            DoB DATE,
                            Address varchar(20),
                            City varchar (20),
                            State varchar (20),
                            Country varchar (20) default "India"
                        )`;

        connection.query(createStudent, function (err, results, fields) {
            if (err) {
                console.log(err.message);
            }
            if (results) {
                console.log("table Creat Success " + results)
            }
            if (fields) {
                console.log(fields);
            }
        });

        connection.end(function (err) {
            if (err) {
                return console.log(err.message);
            }
        });
    });

    res.send('Table Created succesfully')
})
```

2. By using Node.js do the following:
a. Insert 10 records in the above created student table.

Insert Values request

Console logs

```
Connected!
Number of records inserted: 10
```

Database query for verification

```
MariaDB [studentdb]> select * from student;
+----------+---------+--------+-------+--------+--------+------------+------------+------------+------+-------+---------+
| title    | Fname   | Middle | Lname | Course | Branch | Mobile     | DoB        | Address    | City | State | Country |
+----------+---------+--------+-------+--------+--------+------------+------------+------------+------+-------+---------+
| 19104001 | John    | kumar  | singh | B.Tech | IT     | 2147483647 | 2000-12-12 | Highway 71 | ny   | dsds  | usa     |
| 19104002 | amit    | kumar  | singh | B.Tech | IT     | 2147483647 | 2000-12-12 | Highway 71 | ny   | dsds  | INDIA   |
| 19104003 | sanjoli | kumar  | singh | B.Tech | IT     | 2147483647 | 2000-12-12 | Highway 71 | ny   | dsds  | usa     |
| 19104004 | saransh | kumar  | singh | B.Tech | IT     | 2147483647 | 2000-12-12 | Highway 71 | ny   | dsds  | usa     |
| 19104005 | Jkd     | kumar  | singh | B.Tech | IT     | 2147483647 | 2000-12-12 | Highway 71 | ny   | dsds  | INDIA   |
| 19104006 | muskan  | kumar  | singh | B.Tech | IT     | 2147483647 | 2000-12-12 | Highway 71 | ny   | dsds  | usa     |
| 19104007 | kritika | kumar  | singh | B.Tech | IT     | 2147483647 | 2000-12-12 | Highway 71 | ny   | dsds  | usa     |
| 19104008 | jdnfjsf | kumar  | singh | B.Tech | IT     | 2147483647 | 2000-12-12 | Highway 71 | ny   | dsds  | usa     |
| 19104009 | mamam   | kumar  | singh | B.Tech | IT     | 2147483647 | 2000-12-12 | Highway 71 | ny   | dsds  | usa     |
| 19104010 | jdh     | kumar  | singh | B.Tech | IT     | 2147483647 | 2000-12-12 | Highway 71 | ny   | dsds  | usa     |
+----------+---------+--------+-------+--------+--------+------------+------------+------------+------+-------+---------+
10 rows in set (0.000 sec)
```

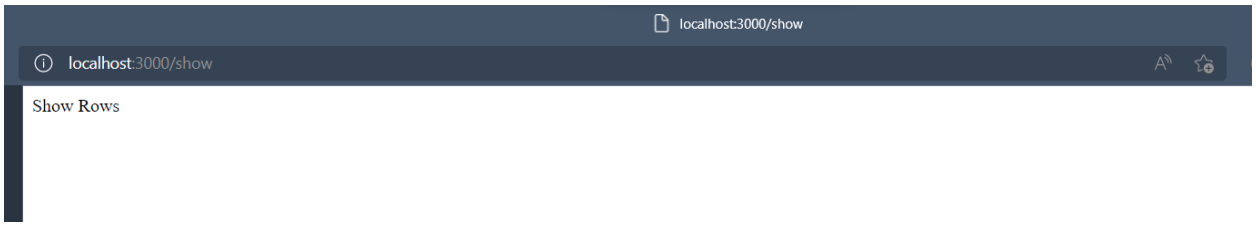
Code

```
app.get('/insertValues', (req, res) => {
    let mysql = require('mysql');
    let connection = mysql.createConnection({
        host: 'localhost',
        user: 'root',
        password: 'root',
        database: 'studentdb'
    });

    connection.connect(function (err) {
        if (err) {
```

```
            return console.error('error: ' + err.message);
        }
        console.log("Connected!");
        var sql = "INSERT INTO student VALUES ?";
        var values = [
            [19104001, 'John', 'kumar', 'singh', 'B.Tech', 'IT',
9857578442, '2000-12-12', 'Highway 71', 'ny', 'dsds', 'usa'],
            [19104002, 'amit', 'kumar', 'singh', 'B.Tech', 'IT',
9857578442, '2000-12-12', 'Highway 71', 'ny', 'dsds', 'INDIA'],
            [19104003, 'sanjoli', 'kumar', 'singh', 'B.Tech', 'IT',
9857578442, '2000-12-12', 'Highway 71', 'ny', 'dsds', 'usa'],
            [19104004, 'saransh', 'kumar', 'singh', 'B.Tech', 'IT',
9857578442, '2000-12-12', 'Highway 71', 'ny', 'dsds', 'usa'],
            [19104005, 'Jkd', 'kumar', 'singh', 'B.Tech', 'IT',
9857578442, '2000-12-12', 'Highway 71', 'ny', 'dsds', 'INDIA'],
            [19104006, 'muskan', 'kumar', 'singh', 'B.Tech', 'IT',
9857578442, '2000-12-12', 'Highway 71', 'ny', 'dsds', 'usa'],
            [19104007, 'kritika', 'kumar', 'singh', 'B.Tech', 'IT',
9857578442, '2000-12-12', 'Highway 71', 'ny', 'dsds', 'usa'],
            [19104008, 'jdnfjsf', 'kumar', 'singh', 'B.Tech', 'IT',
9857578442, '2000-12-12', 'Highway 71', 'ny', 'dsds', 'usa'],
            [19104009, 'mamam', 'kumar', 'singh', 'B.Tech', 'IT',
9857578442, '2000-12-12', 'Highway 71', 'ny', 'dsds', 'usa'],
            [19104010, 'jdh', 'kumar', 'singh', 'B.Tech', 'IT',
9857555552, '2000-12-12', 'Highway 71', 'ny', 'dsds', 'usa']
        ];
        connection.query(sql, [values], function (err, result) {
            if (err) throw (err);
            console.log("Number of records inserted: " +
result.affectedRows);
        });
    });
    res.send("Records Inserted Successfully")
})
```

b. Display all student records.

Show student request



Console Logs

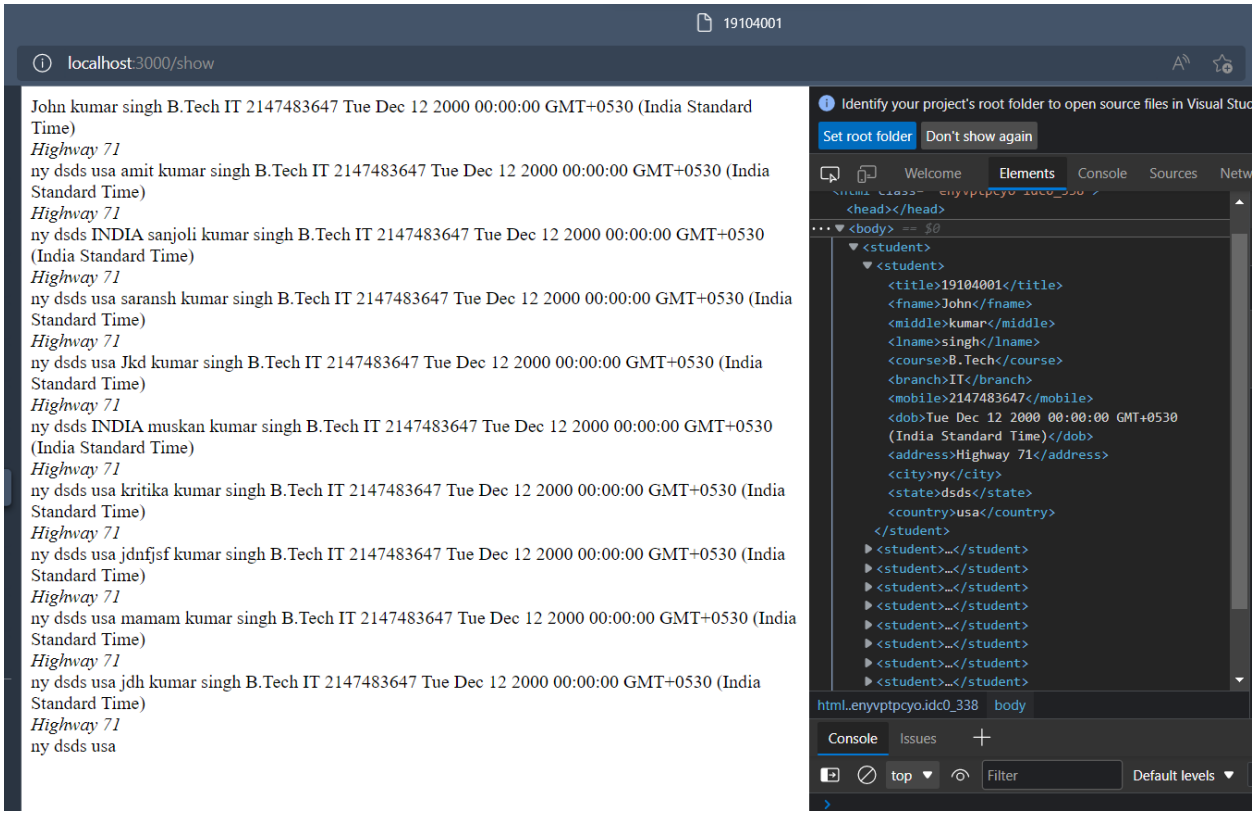
```
Example app listening on port 3000
Fetched Rows Success [object Object],[object Object],[object Object],[object Object],[object Object],[object Object],[object Object],
[object Object],[object Object],[object Object]
RowDataPacket {
  title: '19104001',
  Fname: 'John',
  Middle: 'kumar',
  Lname: 'singh',
  Course: 'B.Tech',
  Branch: 'IT',
  Mobile: 2147483647,
  DoB: 2000-12-11T18:30:00.000Z,
  Address: 'Highway 71',
  City: 'ny',
  State: 'dsds',
  Country: 'usa'
}
RowDataPacket {
  title: '19104002',
  Fname: 'amit',
  Middle: 'kumar',
```

Code

```
app.get('/show, async (req, res) => {
    // let isXml = req.query.isXml; //other wise bydefault JSON
    let isDonwload = req.query.isDonwload;
    let mysql = require('mysql');
    let connection = mysql.createConnection({
        host: 'localhost',
        user: 'root',
        password: 'root',
        database: 'studentdb'
    });

    var rows = []

    // connect to the MySQL server
    connection.connect(function (err) {
        if (err) {
```

```
            return console.error('error: ' + err.message);
        }

        let getAllRows = `SELECT * FROM student`;

        connection.query(getAllRows, function (err, results, fields) {
            if (err) {
                console.log(err.message);
            }
            if (results) {
                console.log("Fetched Rows Success " + results)
                Object.keys(results).forEach(function (key) {
                    var row = results[key];
                    rows.push(row);
                    // console.log(row)
                });
                res.set('Content-Type', 'application/txt);
                console.log(rows)
                return res.send(rows);

            }
            if (fields) {
                // console.log(fields);
            }

        });

        connection.end(function (err) {
            if (err) {
                console.log(err.message);
            }
        });
    });
})
```

c. Display all student records in XML format.



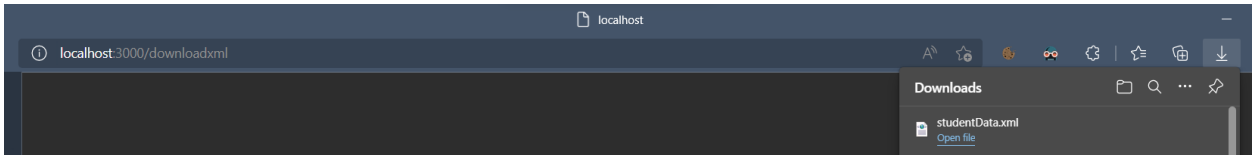
Code

```
app.get('/showxml', async (req, res) => {
    // let isXml = req.query.isXml; //other wise bydefault JSON
    let isDonwload = req.query.isDonwload;
    let mysql = require('mysql');
    let connection = mysql.createConnection({
        host: 'localhost',
        user: 'root',
        password: 'root',
        database: 'studentdb'
    });

    var rows = []

    // connect to the MySQL server
    connection.connect(function (err) {
        if (err) {
```

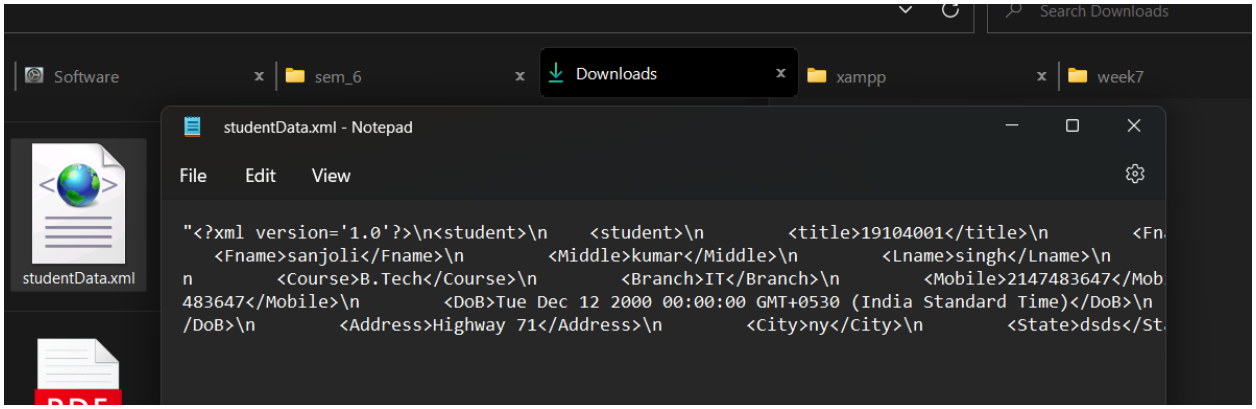
```
            return console.error('error: ' + err.message);
        }

        let getAllRows = `SELECT * FROM student`;

        connection.query(getAllRows, function (err, results, fields) {
            if (err) {
                console.log(err.message);
            }
            if (results) {
                console.log("Fetched Rows Success " + results)
                Object.keys(results).forEach(function (key) {
                    var row = results[key];
                    rows.push(row);
                    // console.log(row)
                });
                // if (isXml == true) {
                console.log("in XML")
                res.set('Content-Type', 'application/xml');
                xml_resultArray = js2xmlparser.parse("student", rows)
                console.log(xml_resultArray)
                return res.send(xml_resultArray);

            }
            if (fields) {
                // console.log(fields);
            }

        });

        connection.end(function (err) {
            if (err) {
                console.log(err.message);
            }
        });
    });
})
```

d. Download the XML data in a file.

Requesting to download xml file



Downloaded XML student file



Code

```
app.get('/downloadxml', async (req, res) => {
    // let isXml = req.query.isXml; //other wise bydefault JSON
    let isDonwload = req.query.isDonwload;
    let mysql = require('mysql');
    let connection = mysql.createConnection({
        host: 'localhost',
        user: 'root',
        password: 'root',
        database: 'studentdb'
    });

    var rows = []

    // connect to the MySQL server
    connection.connect(function (err) {
        if (err) {
            return console.error('error: ' + err.message);
        }

        let getAllRows = `SELECT * FROM student`;
```

```
        connection.query(getAllRows, function (err, results, fields) {
            if (err) {
                console.log(err.message);
            }
            if (results) {
                console.log("Fetched Rows Success " + results)
                Object.keys(results).forEach(function (key) {
                    var row = results[key];
                    rows.push(row);

                });

                res.set('Content-Type', 'application/xml');
                xml_resultArray = js2xmlparser.parse("student", rows)

                const fileName = 'studentData.xml';

                res.set('Content-disposition', `attachment;
filename=${fileName}`);
                return res.json(xml_resultArray);

            }
            if (fields) {
                // console.log(fields);
            }

        });

        connection.end(function (err) {
            if (err) {
                console.log(err.message);
            }
        });
    });
})
```

e. Display all student records in JSON format.

Show Json records

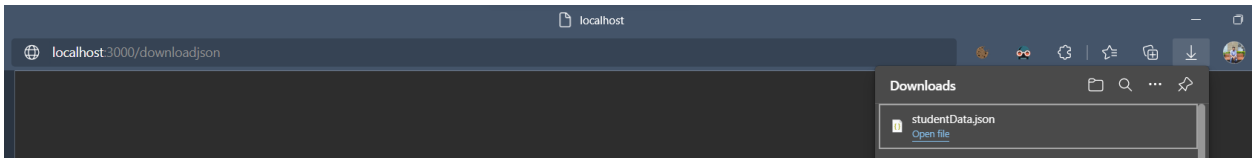
Code

```
app.get('/showjson', async (req, res) => {
    let isDonwload = req.query.isDonwload;
    let mysql = require('mysql');
    let connection = mysql.createConnection({
        host: 'localhost',
        user: 'root',
        password: 'root',
        database: 'studentdb'
    });

    var rows = []

    // connect to the MySQL server
    connection.connect(function (err) {
        if (err) {
            return console.error('error: ' + err.message);
        }

        let getAllRows = `SELECT * FROM student`;

        connection.query(getAllRows, function (err, results, fields) {
            if (err) {
                console.log(err.message);
            }
            if (results) {
```

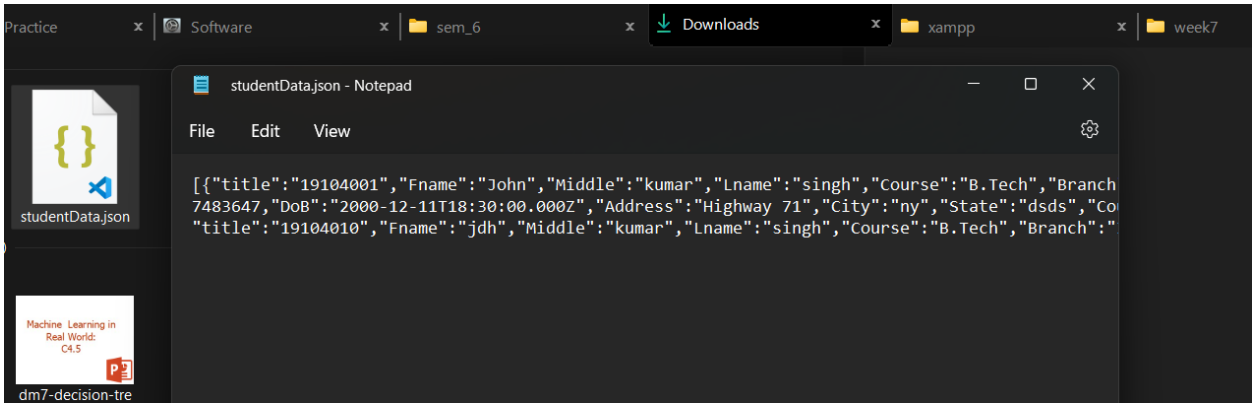
```
                console.log("Fetched Rows Success " + results)
                Object.keys(results).forEach(function (key) {
                    var row = results[key];
                    rows.push(row);
                    // console.log(row)
                });

                console.log("in JSON")
                res.set('Content-Type', 'application/json');
                resultArray =
Object.values(JSON.parse(JSON.stringify(rows)))
                return res.send(resultArray);
            }
            if (fields) {
                // console.log(fields);
            }
        });

        connection.end(function (err) {
            if (err) {
                console.log(err.message);
            }
        });
    });
})
```

f. Download the JSON data in a file.

Download json file request



Downloaded XML student file



Code

```
app.get('/downloadjson', async (req, res) => {
    let isDonwload = req.query.isDonwload;
    let mysql = require('mysql');
    let connection = mysql.createConnection({
        host: 'localhost',
        user: 'root',
        password: 'root',
        database: 'studentdb'
    });

    var rows = []

    // connect to the MySQL server
    connection.connect(function (err) {
        if (err) {
            return console.error('error: ' + err.message);
        }

        let getAllRows = `SELECT * FROM student`;
```

```
        connection.query(getAllRows, function (err, results, fields) {
            if (err) {
                console.log(err.message);
            }
            if (results) {
                console.log("Fetched Rows Success " + results)
                Object.keys(results).forEach(function (key) {
                    var row = results[key];
                    rows.push(row);
                    // console.log(row)
                });
                resultArray =
Object.values(JSON.parse(JSON.stringify(rows)))
                res.set('Content-Type', 'application/json');
                const fileName = 'studentData.json';

                res.set('Content-disposition', `attachment;
filename=${fileName}`);
                return res.json(resultArray);
            }
            if (fields) {
                // console.log(fields);
            }
        });

        connection.end(function (err) {
            if (err) {
                console.log(err.message);
            }
        });
    });
})
```