Directorate of Online Education

SRM Institute of Science and

TechnologySRM Nagar,

Kattankulathur-603203

LAB MANUAL

Course :MCA

SEMESTER- IV

SUBJECTTITLE
Advanced Web Development

SUBJECT CODE

MCAO20104

Prepared By

Dr.S.Umarani
Professor, Department of Computer Science and Applications(BCA)
SRM IST

OBJECTIVES

1. To introduce students to the basic knowledge of Programming for web development

2. To impart writing skill of programming to the students and solving problems.

3. To impart the concepts like nodejs,expressjs,bootstrap,Heroku,mongodb and angular

COURSE OUTCOME

1. Able to write relatively advanced, well structured, computer programs inPython
2. Familiar with principles and techniques for optimizing the performance ofpython numeric applications
3. Able to implement object oriented Programming
4. Able to implement database and GUI applications
5. Able to analysis the data
6. Able to understand visualization and data projections

INTRODUCTION ABOUT VIRTUAL LAB

Virtual Lab Content is prepared by Course Coordinator of concern subject to help the students with their practical understanding and development of programming skills, and may be used as a base reference during the Practical Assignments. The model lab programs and List of Exercise Assignment prepared by staff members will be upload in LMS .Students have to submit Lab Exercise through LMS as Assignment Sections as Separate Folder of concern subject . The course coordinator of concern subject can be evaluated after students submit all program assignments for end semester sectional examination.. The lab Program reporting style in the prescribed format (Appendix-I) and List of Lab Exercises as

Assignments prescribed format (Appendix-II)

APPENDIX-I

1. IMPLEMENT SAMPLE APPLICATION TO DISPLAY HOME,CONTACT AND ABOUT

Aim

Develop program to display home, contact and about page

Procedure

1. Create folder in any place
2. Goto cmd
3. Check the following
   Node -v
   Npm -v
4. Npm install express
5. Open the folder in vscode
6. Add json file
   Npm init -y

7.Create app.js file

Add the following

\

```
const express = require("express");
const app = express() ;
app.get('/', (req, res) =>
{ res.send("Home"); });
app.get('/about', (req, res) =>
{ res.send("About"); });
 app.get('/contact', (req, res) =>
{ res.send("Contact"); });
app.listen(3000, () =>
{ console.log("listening on http://localhost:3000"); })
```

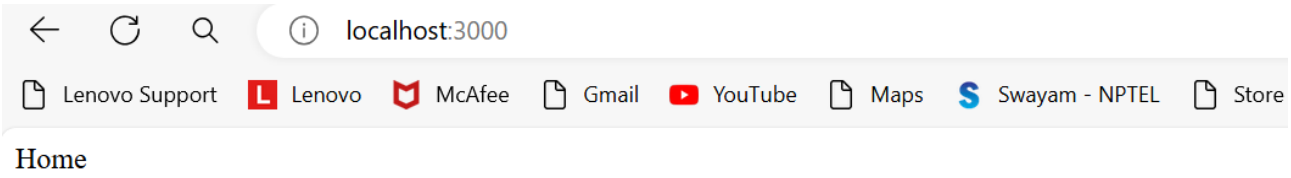
8. Save and run the application

   - Node app.js

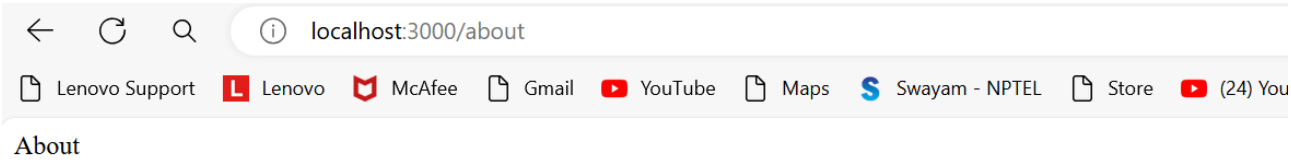
```
C:\Users\karth\OneDrive\Desktop\simple applications>node app.js
listening on http://localhost:3000
```

Output

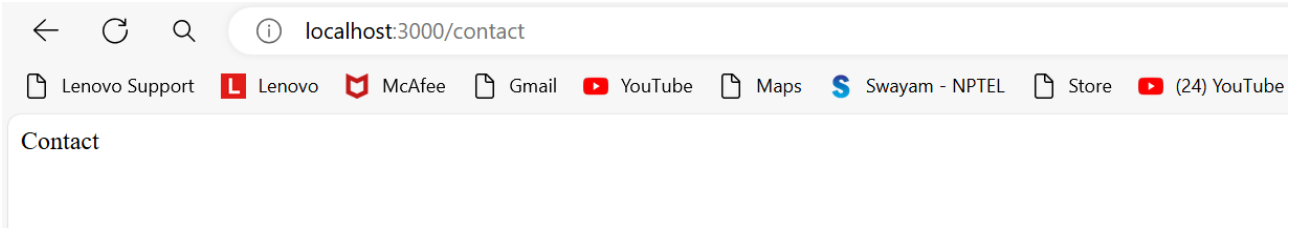
Home Screen



About screen



Contact Screen



Result

Thus the program has been successfully completed

2. IMPLEMENT LOGIN PAGE WITH HTML

Aim

To implement login page with html

Procedure

1. Create folder in any place
2. Goto cmd
3. Check the following
   - Node -v
   - Npm -v
4. Npm install express
5. Open the folder in vscode
6. Add json file
   - Npm init -y

7.Create app.js file

Add the following

```
const express = require("express") ;
const app = express() ;
app.get('/', function(req, res)
 {
   res.sendFile(__dirname +"/"+"index.html");
 }
);

app.listen(3000, () =>
{ console.log("listening on http://localhost:3000"); })
```

8.Create index.html file

Add the following

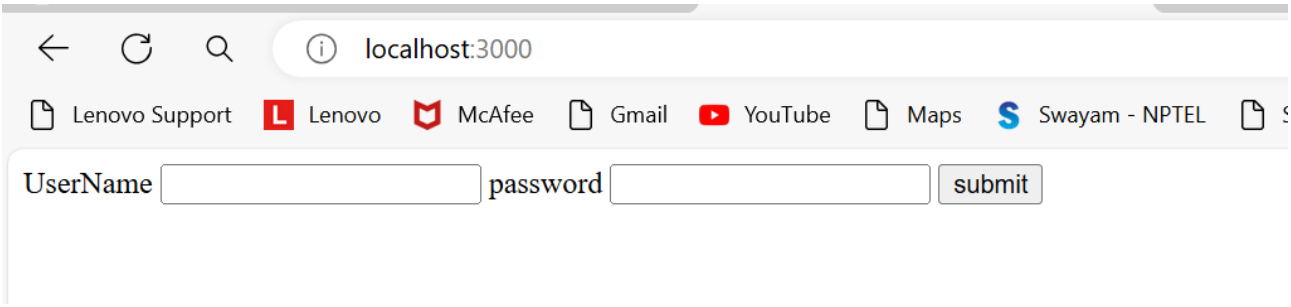
```
<html>
<body>
        <form action="localhost:8080/home" method ="GET">
UserName <input type="text" name="username"/>
password <input type="text" name="password"/>
<input type="submit" value="submit"/>
        </form>
</body>
</html>
```

9. Save and run the application

- Node app.js

```
C:\Users\karth\OneDrive\Desktop\simple applications>node app.js
listening on http://localhost:3000
```

Output



Result

Thus the program has been successfully completed

3. IMPLEMENT SAMPLE APPLICATION WITH HTML PAGE AND ROUTING

Aim

To implement sample application with html page and routing

Procedure

1. Create folder in any place
2. Goto cmd
3. Check the following
   Node -v
   Npm -v
4. Npm install express
5. Open the folder in vscode
6. Add json file
   Npm init -y

7.Create app.js file

```
const express = require("Express") ;
const app = express() ;
const routing = require('./routing');
app.use('/',routing)
app.listen(3000, () =>
{ console.log("listening on http://localhost:3000"); })
```

8.Create index.html

Add the following

```
<html>
   <body>
  <a href ="https://www.youtube.com/">youtube</a>
  <a href="https://www.tutorialpoint.com">TutorialPoint</a>
   </body>
</html>
```

9.Create rouring.js

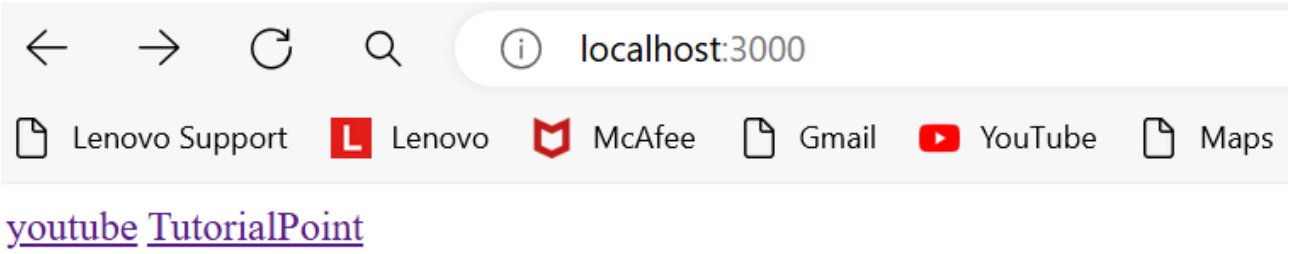
```
const express = require("express") ;
const router= express.Router() ;
router.get('/', function(req, res)
 {
   res.sendFile(__dirname +"/"+"index.html");
 }
);
module.exports = router;
```

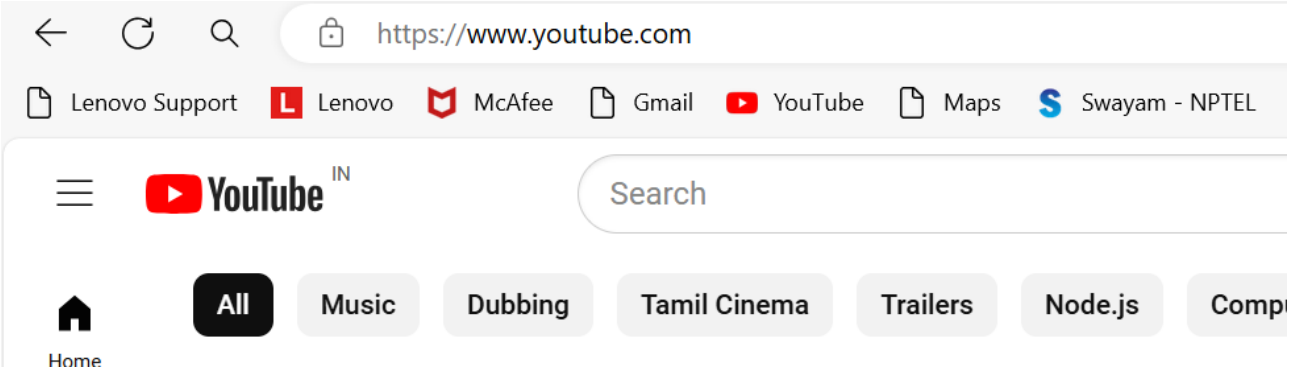
9.save and rin the program

>node app.js

```
C:\Users\karth\OneDrive\Desktop\simple applications>node app.js
listening on http://localhost:3000
```

Output





Result

Thus the program has been completed successfully

4.IMPLEMENT EJS TEMPLATE

Aim

To implement EJS Template.

Procedure

1. Create folder in any place
2. Goto cmd
3. Check the following
   Node -v
   Npm -v
4. Npm install express
5. Open the folder in vscode
6. Add json file
   Npm init -y
   npm install ejs express –save

7.Create app.js file

```
const express = require('express');
const app = express();

app.set('view engine', 'ejs');

app.get('/' , (req , res)=>{
   res.send("welcome");
})

var name = "<%= data %>"
      setTimeout(()=>{
      document.getElementById('hello').innerHTML = name;
      },1000);


app.listen(4000 , ()=>{
   console.log("server is running on port 4000");
})
```

8.Create index.ejs

```
<!--Ejs File–>
<!DOCTYPE html>
<html lang="en">
```

```
<head>
   <meta charset="UTF-8">
   <meta http-equiv="X-UA-Compatible" content="IE=edge">
   <meta name="viewport"
       content="width=device-width, initial-scale=1.0">
</head>
<body>
   <h1>welcome</h1>
</body>
</html>
```

9. Save and run the program

```
>Node app.js
```

Output

welcome

Result

Thus the program has been completed successfully

5.DEVELOP A PROGRAM FOR WEB APPLICATION WITH DATABASE

Aim

To develop a program for web application with database

Procedure

1. Create folder in any place
2. Goto cmd
3. Check the following
   Node -v
   Npm -v
4. Npm install express
5. Open the folder in vscode
6. Add json file
   Npm init -y
   npm install ejs express –save

7.Create index.js file

Index.js

```
const express=require("express");
const app = express();
const path=require("path");
const port =3000;
app.use(express.json());
app.post('/test/create',(req,res)=>
{
    res.send(req.body);
});
app.listen(3000,()=>
{
    console.log("port connected");
});
```

8. Install Postman
   https://web.postman.co/workspace/My-Workspace~5a23bb0b-e449-4d79-8b65-805c2feaede0/request/34988460-42340ceb-2ca5-466a-b7fe-1016629d1e02?tab=body

9. select collection
   - press plus symbol
   - create connection named as CRUD example
   - add method post example
   - change method as Post
   - type url http://localhost:3000/test/create

10 . click body -raw-json
   - add the following
     {
     “title”:”python”,
     “rating”:4.8
     }
   - Click send – result will be displayed

Output

{
“title”:”python”,
“rating”:4.8
}

Result

Thus the program has been completed successfully

6.DEVELOP A WEB APPLICATION USING PUG TEMPLATE

Aim

To develop a web application using PUG Template

Procedure

1.Install PUG
npm install pug
npm install pug –save

1. Create folder in any place
2. Goto cmd
3. Check the following
   Node -v
   Npm -v
4. Install Express
   Npm install express
5. Install PUG
   npm install pug
   npm install pug –save

6.Open the folder in vscode
Create index.pug

```
html
    head
        title= name
    body
        //comment

        h1 Greetings from #{name}
        br
        a(href= url) URL
        br
        br
        div abcd
        br
        div
            | To insert multiline text,
            | You can use the pipe operator.
        br
        div.
            But that gets tedious if you have a lot of text.
            You can use "." at the end of tag to denote block of text.
            To put tags inside this block, simply enter tag in a new line and
            indent it accordingly.
```

7. Create app.js

```
var express = require('express');
var app = express();

app.set('view engine', 'pug');
app.set('views','./views');

app.get('/home', function(req, res){
    res.render('index', {
        name: "TutorialsPoint",
        url:"http://www.tutorialspoint.com"
    });
});

app.listen(3000);
```

8. Save and run the applications

   >node app.js

Output

Result

Thus the program has been completed successfully

7.DEVELOP AN APPLICATION FOR DESIGNING ANGULAR WEBPAGE

Aim

To develop an application for designing angular webpage

Procedure

1.Install vscode
2.Install node js
3.Install angular
npm install -g @angular/cli

Output

Result

Thus the program has been completed successfully

APPENDIX - II

LIST OF EXPERIMENTS -ASSIGNMENTS -LMS

Assignment No	Title of Program
1	Develop simple web application using express JS and Heroku
2	Develop an application using Express Js and Bootstrap
3	Implement MVC concept
4	Adding middleware concept in web applications
5	Develop an application for insert a data in to database
6	Design an page using angular
7	Develop an application for update a data in to database
8	Develop an application for loginsignup page using HBS Template