



K. J. Somaiya College of Engineering, Mumbai-77

(A Constituent College of Somaiya Vidyavihar University)

Department of Computer Engineering

Batch: D-2

Roll No.: 16010122151

Experiment / assignment / tutorial No. _2_

Grade: AA / AB / BB / BC / CC / CD / DD

Signature of the Staff In-charge with date

Title: Implementation of Express concept

AIM: To demonstrate the working Express

Problem Definition:

___Implementation of Express. js

Problem statement:

Consider the basic concepts of Express.js, which are useful in the creation of an application.

Considering the following points, demonstrate the functionality of each with a simple script

1) Serving static files using Express.js: With the help of Built in middleware, express. Static () to demonstrate the usage of serving static files in express.

To demonstrate the above make a use of

- **Use of images where it should accept any type of image**
- **Use of CSS and HTML files.**
- **Make a Use json file of employee information, add file to the static folder, and show the response on the browser.**



K. J. Somaiya College of Engineering, Mumbai-77

(A Constituent College of Somaiya Vidyavihar University)

Department of Computer Engineering

Note: Assume your own data whenever required to perform the operation.

2) Implement any one Template engine (ejs/hbs/pug) to cater the dynamic content using Express Js.

3) Scaffolding:

1. Demonstrate express scaffolding to fulfill the following requirements.

Example: Consider Grocery Delivery Application and demonstrate the Scaffolding Scaffold the application to create different routes such as.

Sign up Page: (Root/ Homepage)

Expected OUTCOME of Experiment:

.

Books/ Journals/ Websites referred:

1. Express .js Deep API reference, by Azat Marden, Apress, 2nd edition, 2015.
2. <https://codeburst.io/building-a-rest-api-using-mongo-db-75cac3403fab>
3. <https://www.edureka.co/blog/rest-api-with-node-js/>
4. <https://bezkode.com/node-express-mongodb-crud-rest-api/>

Pre Lab/ Prior Concepts:

Kindly refer the concepts of Lab/Tut to add the prior concept



K. J. Somaiya College of Engineering, Mumbai-77

(A Constituent College of Somaiya Vidyavihar University)

Department of Computer Engineering

Implementation Details:

1. Serving Static Files using Express.js

```
{
  "name": "exp1",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "keywords": [],
  "author": "",
  "license": "ISC",
  "dependencies": {
    "express": "^4.21.0"
  }
}
```

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

app.use(express.static(path.join(__dirname, 'public')));
app.get('/employees', (req, res) => {
  res.sendFile(path.join(__dirname, 'public', 'employee.json'));
});
const PORT = process.env.PORT || 3000;
app.listen(PORT, () => {
  console.log(`Server is running on http://localhost:${PORT}`);
});
```

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Static File Serving</title>
  <link rel="stylesheet" href="/css/style.css">
</head>
<body>
  <h1>Welcome to Express.js App</h1>
  <p>This page demonstrates serving static files.</p>
  
</body>
```



K. J. Somaiya College of Engineering, Mumbai-77

(A Constituent College of Somaiya Vidyavihar University)

Department of Computer Engineering

```
</html>
```

```
{
  "employees": [
    {
      "id": 1,
      "name": "John Doe",
      "position": "Manager"
    },
    {
      "id": 2,
      "name": "Jane Smith",
      "position": "Developer"
    }
  ]
}
```

< > ↻



localhost:3000

Welcome to Express.js App

This page demonstrates serving static files.





K. J. Somaiya College of Engineering, Mumbai-77

(A Constituent College of Somaiya Vidyavihar University)

Department of Computer Engineering

```
localhost:3000/employees
Pretty-print ☐
{
  "employees": [
    {
      "id": 1,
      "name": "John Doe",
      "position": "Manager"
    },
    {
      "id": 2,
      "name": "Jane Smith",
      "position": "Developer"
    }
  ]
}
```



K. J. Somaiya College of Engineering, Mumbai-77

(A Constituent College of Somaiya Vidyavihar University)

Department of Computer Engineering

2. Using a Template Engine (EJS) for Dynamic Content

```
{
  "name": "exp1",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "keywords": [],
  "author": "",
  "license": "ISC",
  "dependencies": {
    "ejs": "^3.1.10",
    "express": "^4.21.0"
  }
}
```

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Employee List</title>
</head>
<body>
  <h1>Employee List</h1>
  <ul>
    <% employees.forEach(employee => { %>
      <li><%= employee.name %> - <%= employee.position %></li>
    <% }) %>
  </ul>
</body>
</html>
```

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

app.set('view engine', 'ejs');
app.set('views', path.join(__dirname, 'views'));
app.use(express.static(path.join(__dirname, 'public')));
const employees = [
  { id: 1, name: 'John Doe', position: 'Manager' },
```



K. J. Somaiya College of Engineering, Mumbai-77

(A Constituent College of Somaiya Vidyavihar University)

Department of Computer Engineering

```
    { id: 2, name: 'Jane Smith', position: 'Developer' }  
  ];  
  app.get('/', (req, res) => {  
    res.render('index', { employees });  
  });  
  const PORT = process.env.PORT || 3000;  
  app.listen(PORT, () => {  
    console.log(`Server is running on http://localhost:${PORT}`);  
  });
```



localhost:3000

Employee List

- John Doe - Manager
- Jane Smith - Developer



K. J. Somaiya College of Engineering, Mumbai-77

(A Constituent College of Somaiya Vidyavihar University)

Department of Computer Engineering

3. Express Scaffolding (Using express-generator)

```
{
  "name": "grocery-app",
  "version": "0.0.0",
  "private": true,
  "scripts": {
    "start": "node ./bin/www"
  },
  "dependencies": {
    "cookie-parser": "~1.4.4",
    "debug": "~2.6.9",
    "express": "~4.16.1",
    "http-errors": "~1.6.3",
    "jade": "~1.11.0",
    "morgan": "~1.9.1"
  }
}
```

```
var express = require('express');
var router = express.Router();

router.get('/', function(req, res, next) {
  res.render('index', { title: 'Sign Up for Grocery Delivery' });
});

module.exports = router;
```

```
doctype html
html
  head
    title= title
  body
    h1 Sign Up for Grocery Delivery
    form(action="/" method="post")
      input(type="text" placeholder="Enter your name")
      input(type="text" placeholder="Enter your address")
      button(type="submit") Sign Up
```




K. J. Somaiya College of Engineering, Mumbai-77

(A Constituent College of Somaiya Vidyavihar University)

Department of Computer Engineering

Sign Up for Grocery Delivery

Hyder Presswala	hyder.p@somaiya.edu	Sign Up
-----------------	---------------------	---------

Conclusion:

This demonstration shows how to serve static files, use a template engine like EJS, and scaffold an Express.js application for different routes such as a signup page. Each functionality builds on core concepts of Express.js, making it easier to develop applications quickly.