

Day 3 Assignment

Name – SOHAM SARKAR

College Name - MCKV INSTITUTE OF ENGINEERING



June 15, 2025

✅ ASSIGNMENT – DAY 3

📘 Topic: Express with Pug and EJS Templates

📂 Type: Practical – Questions Only

**📝 Questions**

**1.EJS Practical**

**Create an Express application using EJS. Create a dynamic webpage that accepts a user's name and age from a form and displays a personalized greeting on the next page.**

**Folder Structure:**

assignmentClass03\_1**/**

├── public/

│ └── style.css ← CSS file

├── views/

│ └── user-form.ejs ← HTML form in EJS Template

├── server.js ← Express server

├── package.json

**Install Express and EJS**

npm init -y

npm install express ejs

**CSS File Code**

* /\* public/style.css \*/
* body {
* font-family: Arial, sans-serif;
* background-color: #f4f4f4;
* padding: 20px;
* }
* form {
* background-color: white;
* padding: 20px;
* border-radius: 8px;
* max-width: 400px;
* margin: auto;
* box-shadow: 0 0 10px rgba(0,0,0,0.1);
* }
* input, button {
* width: 100%;
* padding: 10px;
* margin-top: 10px;
* }

**User Form EJS File Code**

* <!DOCTYPE html>
* <html>
* <head>
* <title>User Form</title>
* <link rel="stylesheet" href="/style.css">
* </head>
* <body>
* <h1>User Registration</h1>
* <form action="/user" method="POST">
* <label>Name:</label><br />
* <input type="text" name="name" required /><br />
* <label>Email:</label><br />
* <input type="email" name="email" required /><br />
* <label>Age:</label><br />
* <input type="number" name="age" required /><br />
* <button type="submit">Register</button>
* </form>
* </body>
* </html>

**Server Code**

* const express = require('express');
* const app = express();
* const port = 3000;
* // Middleware to parse the data from the user inputed form
* app.use(express.urlencoded({ extended: true }));
* // Serve static files from /public
* app.use(express.static('public'));
* // Set EJS view engine
* app.set('view engine', 'ejs');
* // Routes
* app.get('/user', (req, res) => {
* res.render('userForm');
* });
* app.post('/user', (req, res) => {
* const { name, email, age } = req.body;
* res.send(`Registered: ${name} (${email}), Age: ${age}`);
* });
* app.listen(port, () => {
* console.log(`Server running at http://localhost:${port}`);
* });

**2.PUG Practical**

**Build an Express app using the Pug template engine. Create a webpage that shows a list of student names passed from the server to the template and display them in a styled list format.**

**Folder Structure**

assignmentClass03\_2/

├── views/

│ └── students.pug

├── public/

│ └── style.css

├── server.js

├── package.json

**CSS File Code**

* body {
* font-family: sans-serif;
* background-color: #f0f8ff;
* padding: 20px;
* }
* h1 {
* text-align: center;
* color: #333;
* }
* ul.student-list {
* list-style-type: none;
* padding: 0;
* max-width: 400px;
* margin: 0 auto;
* }
* ul.student-list li {
* background: #d1e7dd;
* padding: 10px;
* margin-bottom: 10px;
* border-radius: 6px;
* box-shadow: 0 0 5px rgba(0,0,0,0.1);
* text-align: center;
* }

**Pug File Code**

* doctype html
* html
* head
* title= title
* link(rel="stylesheet", href="/css/style.css")
* body
* h1= title
* ul.student-list
* each student in studentList
* li= student

**Server Code**

* const express = require('express');
* const app = express();
* const port = 3000;
* // Set Pug as view engine
* app.set('view engine', 'pug');
* app.set('views', './views');
* // Serve static files
* app.use(express.static('public'));
* // Sample student data
* const students = ["Soham", "Riya", "Ankit", "Megha", "Arjun"];
* // Route to render Pug template
* app.get('/students', (req, res) => {
* res.render('students', {
* title: 'Student List',
* studentList: students
* });
* });
* app.listen(port, () => {
* console.log(`App running at http://localhost:${port}/students`);
* });

**3.Template Engine Comparison**

**Create a simple Express app that uses both Pug and EJS in two different routes (/pug and /ejs). Render the same content (e.g., "Welcome to Template Engine Example") in both engines and compare the syntax by viewing both pages.**

**Folder Structure**

assignmentClass03\_3/

├── views/

│ ├── pug-template.pug

│ └── ejs-template.ejs

├── server.js

├── package.json

**Setup & Install Dependencies**

npm init -y

npm install express pug ejs

**Server Code**

* const server = require('express')
* const app = server();
* const path = require('path')
* const port=8000;
* // Register EJS
* app.engine('ejs', require('ejs').\_\_express);
* // Register Pug
* app.engine('pug', require('pug').\_\_express);
* // Default view engine (used when no extension is given)
* app.set('view engine', 'ejs');
* // Set views folder
* app.set('views', path.join(\_\_dirname, 'views'));
* app.get('/ejs',(req,res)=>{
* res.render('ejs\_template',{
* message:"This is ejs example!"
* })
* })
* app.get('/pug',(req,res)=>{
* res.render('pug\_template.pug',{
* message:"This is pug example!"
* })
* })
* app.listen(port,()=>{
* console.log(`1Server is running on localhost:${port}`)
* })

**EJS File Code**

* <!DOCTYPE html>
* <html lang="en">
* <head>
* <meta charset="UTF-8">
* <meta name="viewport" content="width=device-width, initial-scale=1.0">
* <title>Document</title>
* </head>
* <body>
* <h1><%= message %></h1>
* </body>
* </html>

**PUG File Code**

* doctype html
* html
* head
* title Pug Template
* body
* h1= message