
 <b>Marwadi University</b> Marwadi Chandarana Group 	<b>Marwadi University</b> <b>Faculty of Technology</b> <b>Department of Information and Communication Technology</b>	
<b>Subject: Advanced Web Technology (01CT1625)</b>	<b>Aim:</b> Backend: Using NodeJS, build a job search platform or e-commerce platform api to and filter.	
<b>Experiment :- 11</b>	<b>Date:- 6-04-2025</b>	<b>Enrollment No:- 92200133003</b>

## CODE :

### Jobmodels.js

```
module.exports = (sequelize, DataTypes) => {
  const Job = sequelize.define("Job", {
    title: { type: DataTypes.STRING, allowNull: false },
    company: { type: DataTypes.STRING, allowNull: false },
    location: { type: DataTypes.STRING },
    salary: { type: DataTypes.INTEGER },
    type: { type: DataTypes.ENUM('Full-Time', 'Part-Time', 'Internship') },
    description: { type: DataTypes.TEXT }
  });

  return Job;
};
```

### Dbconfig.js

```
require('dotenv').config();

module.exports = {
  HOST: process.env.DB_HOST,
  USER: process.env.DB_USER,
  PASSWORD: process.env.DB_PASSWORD,
  DB: process.env.DB_NAME,
  dialect: process.env.DB_DIALECT || "mysql",
};
```



### Job.controller.js

```
const db = require("../models");
const Job = db.jobs;
const Op = db.Sequelize.Op;

exports.createJob = async (req, res) => {
  try {
    const job = await Job.create(req.body);
    res.status(201).json(job);
  } catch (err) {
    res.status(500).json({ error: err.message });
  }
};

exports.getJobs = async (req, res) => {
  try {
    const where = {};
    const { title, location, type, minSalary } = req.query;

    if (title) where.title = { [Op.like]: `%${title}%` };
    if (location) where.location = location;
```

 <b>Marwadi University</b> Marwadi Chandarana Group 	<b>Marwadi University</b> <b>Faculty of Technology</b> <b>Department of Information and Communication Technology</b>	
<b>Subject: Advanced Web Technology (01CT1625)</b>	<b>Aim:</b> Backend: Using NodeJS, build a job search platform or e-commerce platform api to and filter.	
<b>Experiment :- 11</b>	<b>Date:- 6-04-2025</b>	<b>Enrollment No:- 92200133003</b>

```

if (type) where.type = type;
if (minSalary) where.salary = { [Op.gte]: minSalary };

```

```

const jobs = await Job.findAll({ where });
res.json(jobs);
} catch (err) {
  res.status(500).json({ error: err.message });
}
};

```

### Index.js

```

const { Sequelize } = require('sequelize');
const dbConfig = require("../config/db.config");

```

```

const sequelize = new Sequelize(
  dbConfig.DB,
  dbConfig.USER,
  dbConfig.PASSWORD,
  {
    host: dbConfig.HOST,
    dialect: dbConfig.dialect,
    logging: false,
  }
);

```

```

const db = {};
db.Sequelize = Sequelize;
db.sequelize = sequelize;

```

```

db.jobs = require("../job.model")(sequelize, Sequelize);

```

```

module.exports = db;

```

### Server.js

```

const express = require("express");
const app = express();
const db = require("../models");
require("dotenv").config();

```

```

const PORT = process.env.PORT || 3000;

```

```

app.use(express.json());

```

```

// Routes

```

```



const jobRoutes = require("../routes/job.routes");
app.use("/api", jobRoutes);

```

```

app.get("/", (req, res) => {
  res.send("Welcome to the Job Search API ");
});

```

 <b>Marwadi University</b> Marwadi Chandarana Group	 <b>Marwadi University</b> <b>Faculty of Technology</b> <b>Department of Information and Communication Technology</b>
<b>Subject: Advanced Web Technology (01CT1625)</b>	<b>Aim:</b> Backend: Using NodeJS, build a job search platform or e-commerce platform api to and filter.
<b>Experiment :- 11</b>	<b>Date:- 6-04-2025</b>
	<b>Enrollment No:- 92200133003</b>

});

```
// Connect to DB and start server
db.sequelize.sync().then(() => {
  app.listen(PORT, () => {
    console.log(`Server is running at http://localhost:${PORT}`);
  });
});
```

### .env

```
DB_HOST=localhost
DB_USER=root
DB_PASSWORD=
DB_NAME=mu_2425_test
DB_DIALECT=mysql
PORT=3001
```

### OUTPUT :

POST

http://localhost:3001/api/jobs

Send

Query

Headers 2

Auth

Body 1

Tests

Pre Run

JSON

XML

Text

Form

Form-encode

GraphQL

Binary

JSON Content

Format

```
1 {
2   "title": "Frontend Developer",
3   "company": "Google",
4   "location": "Remote",
5   "salary": 90000,
6   "type": "Full-Time",
7   "description": "Work on UI development with React."
8 }
9
```

Status: 201 Created

Size: 239 Bytes

Time: 62 ms

Response

Headers 6

Cookies

Results

Docs

```
1 {
2   "id": 1,
3   "title": "Frontend Developer",
4   "company": "Google",
5   "location": "Remote",
6   "salary": 90000,
7   "type": "Full-Time",
8   "description": "Work on UI development with React.",
9   "updatedAt": "2025-04-14T19:29:16.234Z",
10  "createdAt": "2025-04-14T19:29:16.234Z"
11 }
```

GET

http://localhost:3001/api/jobs

Send

Query

Headers 2

Auth

Body 1

Tests

Pre Run

JSON

XML

Text

Form

Form-encode

GraphQL

Binary

JSON Content

Format

```
1
```

Status: 200 OK

Size: 241 Bytes

Time: 29 ms

Response


Headers 6

Cookies

Results

Docs

```
1 [
2   {
3     "id": 1,
4     "title": "Frontend Developer",
5     "company": "Google",
6     "location": "Remote",
7     "salary": 90000,
8     "type": "Full-Time",
9     "description": "Work on UI development with React.",
10    "createdAt": "2025-04-14T19:29:16.000Z",
11    "updatedAt": "2025-04-14T19:29:16.000Z"
12  }
13 ]
```

<div><div><div><div>Marwadi</div><div>University</div><div>Marwadi Chandarana Group</div></div></div><div><div>NAAC</div><div>A+</div></div></div> <div><div>Marwadi University</div><div>Faculty of Technology</div><div>Department of Information and Communication Technology</div></div>	
<b>Subject: Advanced Web Technology (01CT1625)</b>	<b>Aim:</b> Backend: Using NodeJS, build a job search platform or e-commerce platform api to and filter.
<b>Experiment :- 11</b>	<div><div><b>Date:- 6-04-2025</b></div><div><b>Enrollment No:- 92200133003</b></div></div>

GET http://localhost:3001/api/jobs?title=developer&location=Remote&type=Send

QueryHeaders 2AuthBodyTestsPre Run

JSONXMLTextFormForm-encodeGraphQLBinary

JSON ContentFormat

1

Status: 200 OKSize: 241 BytesTime: 25 ms

ResponseHeaders 6CookiesResultsDocs {}

1 [2 {3 "id": 1,4 "title": "Frontend Developer",5 "company": "Google",6 "location": "Remote",7 "salary": 90000,8 "type": "Full-Time",9 "description": "Work on UI development with React.",10 "createdAt": "2025-04-14T19:29:16.000Z",11 "updatedAt": "2025-04-14T19:29:16.000Z"12 }13 ]

	id	title	company	location	salary	type	description	createdAt	updatedAt
<div><div></div><div>Edit</div><div>Copy</div><div>Delete</div></div>	1	Frontend Developer	Google	Remote	90000	Full-Time	Work on UI development with React.	2025-04-14 19:29:16	2025-04-14 19:29:16