

Pushkala. S
21011102075
WT LAB ASSGN 8

APP.js

```
// Import necessary modules and models
const express = require('express');
const mongoose = require('mongoose');
const Task = require('./models/Task'); // Import your Mongoose model here
const app = express();
const port = 3000;

// Connect to MongoDB
mongoose.connect('mongodb://0.0.0.0:27017/my-database', {
  useNewUrlParser: true,
  useUnifiedTopology: true,
});

// Middleware
app.use(express.json()); // Parse JSON request bodies

// Define a route handler for creating a task (POST request)
app.post('/tasks', async (req, res) => {
  try {
    // Parse the request body and create a new task
    const task = new Task(req.body);
    await task.save();

    // Send a success response (HTTP status code 201) with the created task
    res.status(201).json(task);
  } catch (error) {
    // Handle errors, and send an error response (HTTP status code 400)
    res.status(400).json({ error: 'Could not create task' });
  }
});

// Define a route handler for retrieving all tasks (GET request)
app.get('/tasks', async (req, res) => {
  try {
    const tasks = await Task.find();
    res.status(200).json(tasks);
  } catch (error) {
    res.status(500).json({ error: 'Server error' });
  }
});

// Define a route handler for the root URL (GET request)
app.get('/', (req, res) => {
```

```
    res.send('Welcome to the task management API');
  });

// Start the Express server
app.listen(port, () => {
  console.log(`Server is running on http://localhost:${port}`);
});
```

TASK.js

```
const mongoose = require('mongoose');

const taskSchema = new mongoose.Schema({
  title: String,
  description: String,
  completed: Boolean,
});

const Task = mongoose.model('Task', taskSchema);

module.exports = Task;
```

taskRoutes.js

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

// Import the Task model and set up the CRUD operations

// Define routes for tasks
router.post('/tasks', async (req, res) => {
  // Create a new task
});

router.get('/tasks', async (req, res) => {
  // Retrieve all tasks
});

router.get('/tasks/:id', async (req, res) => {
  // Retrieve a specific task by ID
});
```

```
router.put('/tasks/:id', async (req, res) => {
  // Update a task by ID
});

router.delete('/tasks/:id', async (req, res) => {
  // Delete a task by ID
});

module.exports = router;
```

