

CLOUD NATIVE APPLICATIONS

TASK8.1P

TASK

Here is the github link for the project: <https://github.com/DanielTownleyyy/AddingADatabase>

Screenshots of task

Docker File

```
Dockerfile — Edited
# Setup for simple Node.js Application!!!
# Use Node.js base image
FROM node:16

# Set working directory
WORKDIR /app

# Copy package.json and package-lock.json
COPY package*.json ./

# Install dependencies
RUN npm install

# Copy the rest of the app
COPY . .

# Expose port (adjust if different)
EXPOSE 3000

# Run the app
CMD ["node", "server.js"]
```

Mongo

```
JS server.js {} package.json ! mongo-pv.yaml X
! mongo-pv.yaml
1  apiVersion: v1
2  kind: PersistentVolume
3  metadata:
4    name: mongo-pv
5    namespace: my-app
6  spec:
7    capacity:
8      storage: 1Gi
9    accessModes:
10     - ReadWriteOnce
11    hostPath:
12      path: /data/db
13
14  apiVersion: v1
15  kind: PersistentVolumeClaim
16  metadata:
17    name: mongo-pvc
18    namespace: my-app
19  spec:
20    accessModes:
21     - ReadWriteOnce
22    resources:
23      requests:
24        storage: 1Gi
25
26
```

Server.js

```
JS server.js X {} package.json ! mongo-pv.yaml
JS server.js > app.listen() callback
1 // Import required libraries
2 const express = require('express'); // Web framework to build
3 const mongoose = require('mongoose'); // Library to connect to M
4 const bodyParser = require('body-parser'); // Middleware to handle JS
5
6 // Create the Express app
7 const app = express();
8
9 // Define the port number the server will run on
10 const port = 3000;
11
12 // Use body-parser to read JSON data from incoming requests
13 app.use(bodyParser.json());
14
15 // MongoDB connection string (in Docker, "mongo" is the hostname of the
16 const mongoURI = 'mongodb://mongo:27017/mydatabase';
17
18 // Connect to MongoDB using Mongoose
19 mongoose.connect(mongoURI, {
20   useNewUrlParser: true, // Use the new URL parser
21   useUnifiedTopology: true, // Use the new connection engine
22 });
23
24 // Listen for errors or successful connection
25 const db = mongoose.connection;
26 db.on('error', (err) => console.error('MongoDB connection error:', err));
27 db.once('open', () => console.log('Connected to MongoDB'));
28
29 // Define a schema (structure) for our data
30 const ItemSchema = new mongoose.Schema({
31   name: String, // Each item will have a "name" field
32   createdAt: {
33     type: Date, // Automatically adds the date it was created
34     default: Date.now
35   }
36 });
37
38 // Create a model from the schema
39 const Item = mongoose.model('Item', ItemSchema);
40
41 // A test route to check if the server is running
42 app.get('/', (req, res) => {
43   res.send('Hello from Dockerized Node.js app with MongoDB!');
44 });
45
```

Package.json

```
JS server.js {} package.json X ! mongo-pv.yaml
{} package.json > ...
1 {
2   "name": "addingadatabase",
3   "version": "1.0.0",
4   "main": "server.js",
5   "scripts": {
6     "start": "node server.js"
7   },
8   "dependencies": {
9     "express": "^4.18.2",
10    "mongoose": "^7.0.3",
11    "body-parser": "^1.20.2"
12  }
13 }
14
```

```
000-code.cs — Magic8ball
Users > danieltownley > Downloads > 1376779 > 000-code.cs

51 public void Add(T element)
52 {
53     if (Count == Capacity) ExtendData(DEFAULT_CAPACITY);
54     data[Count++] = element;
55 }
56
57 public int IndexOf(T element)
58 {
59     for (int i = 0; i < Count; i++)
60     {
61         if (EqualityComparer<T>.Default.Equals(data[i], element)) return i;
62     }
63     return -1;
64 }
65
66 public void Insert(int index, T element)
67 {
68     if (index < 0 || index > Count) throw new IndexOutOfRangeException();
69     if (Count == Capacity) ExtendData(DEFAULT_CAPACITY); // this expands
70
71     for (int i = Count; i > index; i--)
72     {
73         data[i] = data[i - 1];
74     }
75
76     data[index] = element;
77     Count++;
78 }
79
80 public void Clear()
81 {
82     data = new T[DEFAULT_CAPACITY];
83     Count = 0;
84 }
85
86 public bool Contains(T element)
87 {
88     return IndexOf(element) != -1;
89 }
90
91 public bool Remove(T element)
```

```
000-code.cs — Magic8ball
000-code.cs
Users > danieltownley > Downloads > 1376779 > 000-code.cs
94
95 public bool Remove(T element)
96 {
97     int index = IndexOf(element);
98     if (index == -1) return false;
99
100     RemoveAt(index);
101     return true;
102 }
103
104 public void RemoveAt(int index)
105 {
106     if (index < 0 || index >= Count) throw new IndexOutOfRangeException();
107
108     // Manually shift elements left
109     for (int i = index; i < Count - 1; i++)
110     {
111         data[i] = data[i + 1];
112     }
113
114     Count--;
115 }
116
117 public override string ToString()
118 {
119     if (Count == 0) return "[]";
120     StringBuilder sb = new StringBuilder("[");
121     for (int i = 0; i < Count; i++)
122     {
123         sb.Append(data[i]);
124         if (i < Count - 1) sb.Append(", ");
125     }
126     sb.Append("]");
127     return sb.ToString();
128 }
129
130 }
131
Ln 85, Col 10 Spaces: 4 UTF-8 LF C# Go Live
```