**Todo interview assignment**

Start by installing the node modules in both the backend and client, todo this run npm install in the terminals of both.

The backend will create a server running at

<http://localhost:9091>

With the todo data stored under

<http://localhost:9091/api/todo>

**Adding new todos**

Server.js in the backend. Text before;

**const** express = require("express");

**const** cors = require("cors");

**const** repository = require("./repository/todo");

**const** todoService = require("./service/todo")(repository);

**const** server = () **=>** {

**const** server = express();

  server.use(express.json());

  server.use(cors());

  server.get("/api/todo", **async** (req, res) **=>** {

    res.json(await todoService.getTodos());

  });

*/\*\**

*POST /api/todo*

*{*

*"task": "Some API"*

*}*

*{*

*"todos": [*

*{*

*"task": "Some API"*

*}*

*]*

*}*

*\*\*/*

  return server;

};

module.exports = server;

We need to add a post response, or how the server responds when it receives a request to post data.

  server.post("/api/todo", **async** (req, res) **=>** {

**const** newTodo = {

      task: req.body.task,

    };

**const** todosData = await todoService.getTodos();

    todosData.todos.push(newTodo);

    res.status(201).json(todosData);

  });

This code tells the server what to do when it receives a post request. When a post request is made the data posted is put into a new object newData. The server then retrieves the current server data and saves it to the object todosData. The posted data is then added to this object, and the combined object returned.