




# Welcome to the CoGrammar

## Task Walkthrough: Task 24 - React - Back-end Integration

The session will start shortly...

Questions? Drop them in the chat. We'll have dedicated moderators answering questions.



# Full Stack Web Development Session Housekeeping

---

- The use of disrespectful language is prohibited in the questions, this is a supportive, learning environment for all - please engage accordingly.  
**(Fundamental British Values: Mutual Respect and Tolerance)**
- No question is daft or silly - **ask them!**
- There are **Q&A sessions** midway and at the end of the session, should you wish to ask any follow-up questions. Moderators are going to be answering questions as the session progresses as well.
- If you have any questions outside of this lecture, or that are not answered during this lecture, please do submit these for upcoming Academic Sessions. You can submit these questions here: [Questions](#)

## Full Stack Web Development Session Housekeeping cont.

---

- For all **non-academic questions**, please submit a query: [www.hyperiondev.com/support](https://www.hyperiondev.com/support)
- Report a **safeguarding** incident: [www.hyperiondev.com/safeguardreporting](https://www.hyperiondev.com/safeguardreporting)
- We would love your **feedback** on lectures: [Feedback on Lectures](#)

# Skills Bootcamp

## 8-Week Progression Overview

### Fulfil 4 Criteria to Graduation

#### ✓ Criterion 1: Initial Requirements

Timeframe: First 2 Weeks

Guided Learning Hours (GLH):

Minimum of 15 hours

Task Completion: First four tasks

**Due Date: 24 March 2024**

#### ✓ Criterion 2: Mid-Course Progress

**60** Guided Learning Hours

Data Science - **13 tasks**

Software Engineering - **13 tasks**

Web Development - **13 tasks**

**Due Date: 28 April 2024**

# Skills Bootcamp Progression Overview

## ✓ Criterion 3: Course Progress

Completion: All mandatory tasks,  
including Build Your Brand and  
resubmissions by study period end  
Interview Invitation: Within 4 weeks  
post-course  
Guided Learning Hours: Minimum of  
112 hours by support end date  
(10.5 hours average, each week)

## ✓ Criterion 4: Demonstrating Employability

Final Job or Apprenticeship  
Outcome: Document within 12  
weeks post-graduation  
Relevance: Progression to  
employment or related  
opportunity

# CoGrammar

## Task Walkthrough: Task 24 - React - Back-end Integration

June 2024



# Learning Objectives

- ❖ Explain the role of Express in building web applications and its interaction with React in the context of full-stack development.
- ❖ Develop an Express server with CORS enabled and create API endpoints that return JSON responses.
- ❖ Configure a React application to proxy API requests to an Express back-end and use Axios/Fetch for HTTP communications.



## Prerequisites:

- Basic knowledge of JavaScript, React, and Node.js.
- Node.js and npm installed on your machine.





# Set Up the Back-end

- Create a new directory for your back-end and initialize a Node.js project:
  - `mkdir backend`
  - `cd backend`
  - `npm init -y`
- Install packages:
  - `npm install express cors`
- Create a server

# Set Up Front-end

- Create a React application:
  - `npx create-react-app frontend`
  - `cd frontend`
- Configure proxy to connect with the backend in your `package.json`:`
- `"proxy": "http://localhost:5000",`
- Install axios for making HTTP requests. You can also use the Fetch API:
  - `npm install axios`
- Fetch messages from endpoints and display them

# Running the Application:

- Start both the back-end and front-end servers in separate terminal windows:
  - Back-end: Navigate to the backend directory and run `node server.js`.
  - Front-end: Navigate to the frontend directory and run `npm start`.
- Access your React application at <http://localhost:3000> to see the messages fetched from your back-end displayed on the web page.

# Questions and answers



# Thank you for attending



Department  
for Education

CoGrammar

