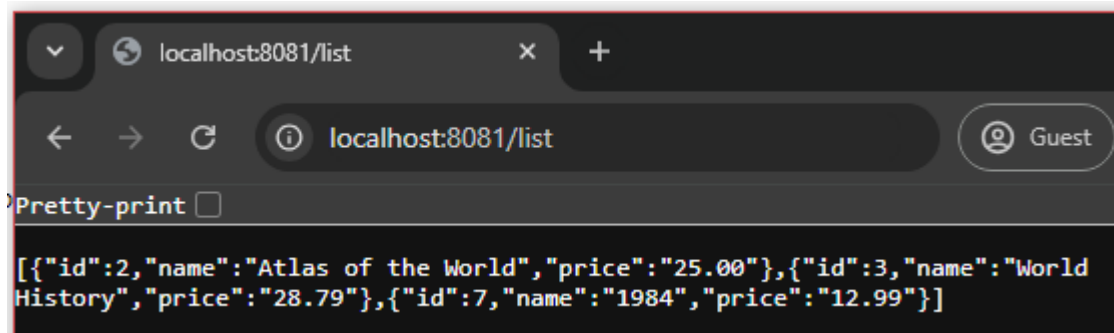# Passing database data to EJS templates

Welcome to this lab activity

In this lab activity, you will explore how to select data from the database and pass it to an EJS template for displaying to a user.

In the previous lab, you retrieved a list of books from the database and presented them in a `list'` page in your web application:

The data is being shown as raw JSON. This isn't the most attractive way to present data. In this lab, you will use EJS to embed this data in a nicely formatted web page.

Task 1: Make a copy of the lab code

You will start with your previous Node.js lab code.

1. Make a copy of the folder `15_access_database` from your previous lab and call it `16_pass_to_template`.
2. Open up the new folder in Visual Studio Code.
3. Run the code and confirm that the list page still works (refer to the previous lab if you need a reminder of how to do this).

Task 2: Create an EJS file that displays the data

4. Create a new EJS file called `list.ejs` in the `views` folder. Add the following EJS code to it:

```
<!doctype html>
<html>
  <head>
    <title>List webpage!</title>
  </head>
  <body>
```

```
    <h1>Our Books</h1>
    <h3>Here is a list of the books we sell:</h3>
    <ul>
      <% availableBooks.forEach(function(book){ %>
      <li><%= book.name %>, £<%= book.price %></li>
     <% }) %>
    </ul>
  </body>
</html>
```

Notice the EJS code embedded in the HTML:

```
    <% availableBooks.forEach(function(book){ %>
    <li><%= book.name %>, £<%= book.price %></li>
   <% }) %>
```

The EJS code is expecting an object called `availableBooks` and is looping through the data within it, displaying each book name and price.

Task 3: Render the EJS file

In the previous lab, you sent the raw results from the query `SELECT * FROM books` in the HTTP response. Let's change this to embed these results into an HTML page and send that HTML page back. You will use EJS to achieve this.

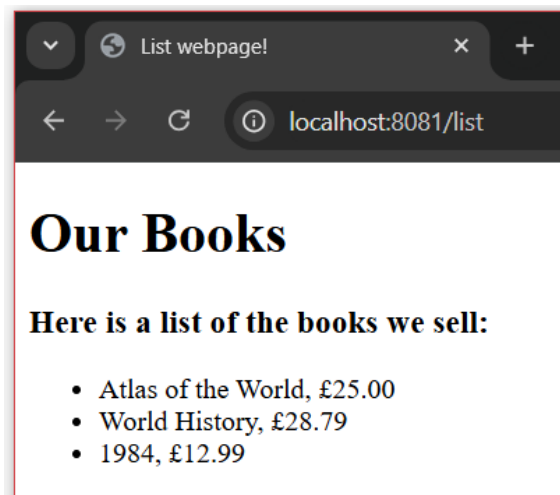**5.** In main.js, locate the /list route handler. Replace the following line:

```
res.send(result);
```

with this:

```
res.render("list.ejs", {availableBooks: result});
```

The above code renders `list.ejs`, passing the `result` object as an object called `availableBooks`.

6. Now run the code and browse to the /list route. You should see a nicely formatted list of books:

Task 4: Explore further

When tackling these lab activities, it's always good to stretch yourself by doing a bit of research and attempting some changes on your own.

In the previous lab, the 'explore further' activity was to show a 'bargain books' page. Can you make this page display the results nicely using the same `list.ejs` code that you created in this lab? Don't duplicate the code. You should use the exact same `list.ejs` file for both pages.

End of lab

Congratulations on completing this lab.

In the next lab activity, you will learn how to pass variables from a form to the backend and add books to your database.