Passing variables to back-end

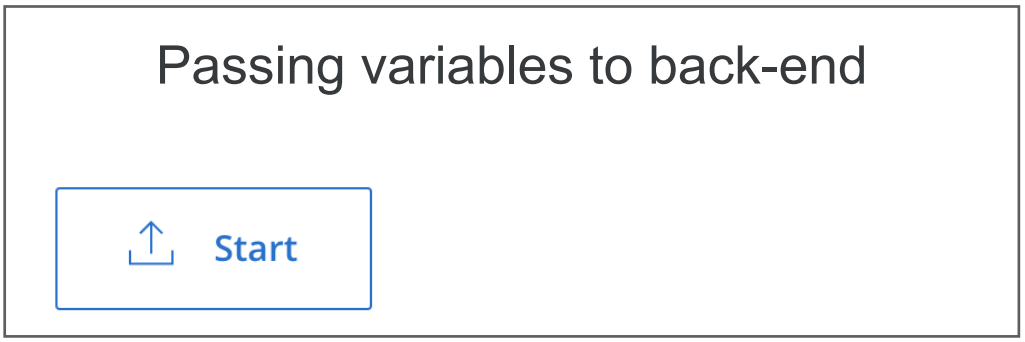
Welcome to this Lab activity

In this lab activity you will be exploring how to pass different variables to your application back-end; How about adding a new book to your database? For the purpose of this lab activity you will be working on the *topic6/expressDynamicApp*.

Start the Lab environment application

It is simple to launch a lab exercise. You only need to click on the button "Start" below the activity title to enter a lab environment.

Let's explore this lab activity. Go ahead and click on the "Start" button!



Task 1: Add a new page and a new route to your web application

The folder structure has already been partially constructed for you and organised into different topics. For the purpose of this lab, you will be making changes inside the *topic6/ expressDynamicApp* folder structure. Let's get started!

Please do not delete or move any existing folders/files inside the lab environment.

In this lab activity you will be challenged to create part of the application code yourself in preparation for your mid-term assessment.

Your task is to add a new page and a new route to your web application called “addbook”. The newly created webpage should ask the user to enter a book name and its price. Hitting the submit button will save the data in the database. Furthermore you should have a link to the “/list” route from the “addbook” page so that you can view your books without manually visiting the “/list” route.

Hint:

- First of all, ask yourself which page of your web application has similar functionalities to the “/addbook” page? Can you re-use parts of the code for this new page?
- Update your “main.js” file, adding the new route “/addbook”.
- Create a new html file called “addbook.html”, containing the input form.
- Hitting the submit button on the “addbook” page, should redirect the user to a new page called “/bookadded”, save the data in database and display a message that the data is saved in database.
- Adding **View all the books** to your “addbook” page should redirect you to the “/list” route.

I have included the piece of code that you need to add to your “main.js” file:

```
app.post("/addbook", function (req,res) {
  // saving data in database
  let sqlquery = "INSERT INTO books (name, price) VALUES (?,?)";
  // execute sql query
  let newrecord = [req.body.name, req.body.price];
  db.query(sqlquery, newrecord, (err, result) => {
    if (err) {
      return console.error(err.message);
    }else
      res.send(" This book is added to database, name: "+ req.body.name + " price "+
      req.body.price);
  });
});
```

Just a reminder that you should modify your application inside the *topic6/expressDynamicApp* folder.

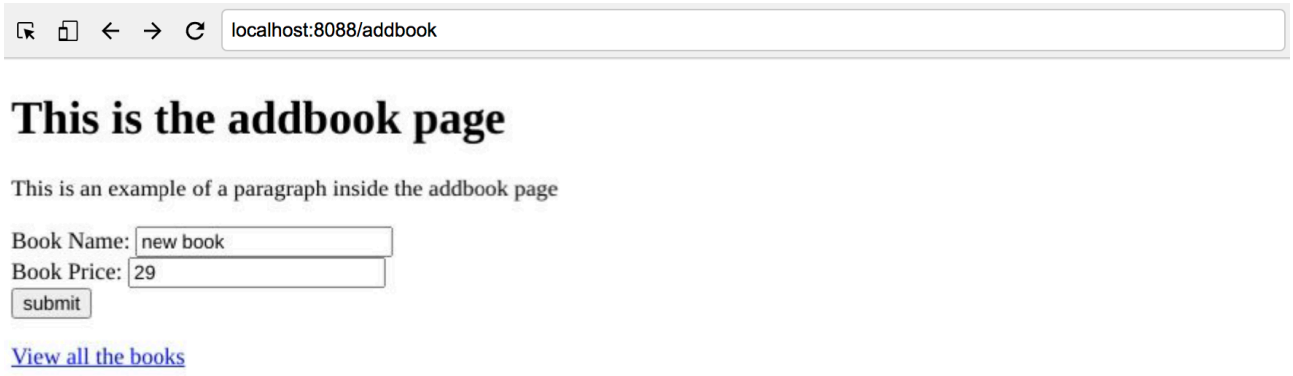
Task 2: Access your server via HTTP

Now that your code is running, serving your application on port 8088, you can access your web page from a browser!

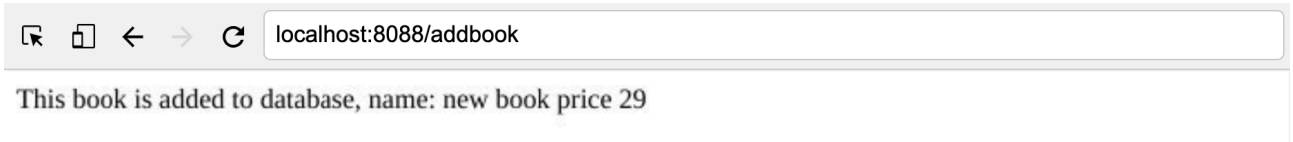
Use the “Browser Preview” plugin to visualise your web application.

If you do not remember how to use the “Browser Preview” plugin, please refer to the “Creating my first Node.js web server” lab instructions.

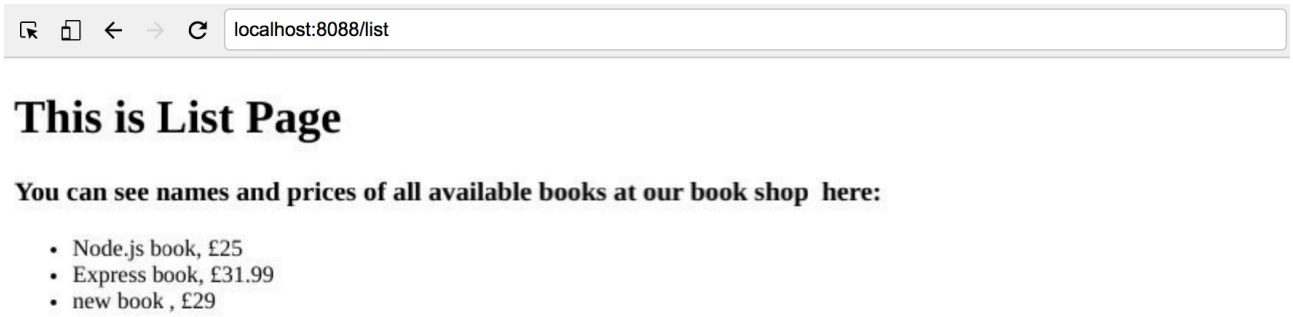
Type *localhost:8088/addbook* on the “Browser Preview” tab and press *Enter* on your keyboard to visualise your web application:



When you add a book to your database, you should see the following page:



Finally clicking the “View all the books” link on the “/addbook” should direct you to the “/list” route and you should be able to see your newly added book:



Just a reminder that you might have different data in your “/list” route page.

End of Section

Congratulations for completing this section. As long as you have saved your work, your files will remain when you close this lab activity so do not worry about losing your data. You have successfully created a web application that connects to a database and passes variables to it in order to add items. In the next lab activity you will learn how to use the “search” query in your database to search a particular item.