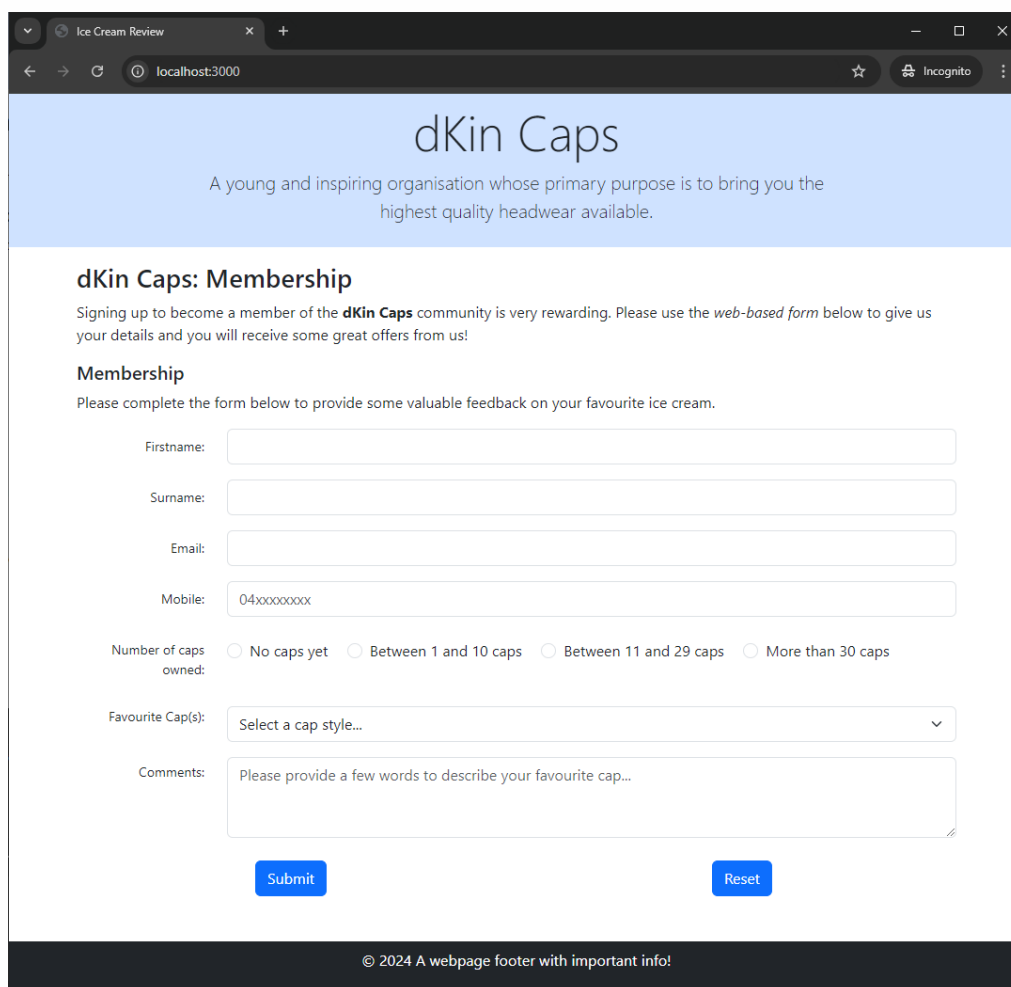


9.1P: Send a Response to the Received Data

Tasks

In previous weeks we have looked at how we can handle `GET` requests in an express server. Here, we will look at how to handle `POST` requests, as generated when submitting a web form. You are supplied with the code that presents the following web page form for accepting some information for a user's membership registration, including their name, an email address, phone number, their favourite cap style and a comment:



The screenshot shows a web browser window with the address bar displaying 'localhost:3000'. The page has a light blue header with the text 'dKin Caps' and a subtitle 'A young and inspiring organisation whose primary purpose is to bring you the highest quality headwear available.' Below the header, the main content area is titled 'dKin Caps: Membership' and includes a paragraph: 'Signing up to become a member of the dKin Caps community is very rewarding. Please use the web-based form below to give us your details and you will receive some great offers from us!'. The form itself is titled 'Membership' and asks the user to 'Please complete the form below to provide some valuable feedback on your favourite ice cream.' The form fields include: 'Firstname:' (text input), 'Surname:' (text input), 'Email:' (text input), 'Mobile:' (text input with placeholder '04xxxxxxxx'), 'Number of caps owned:' (radio buttons for 'No caps yet', 'Between 1 and 10 caps', 'Between 11 and 29 caps', and 'More than 30 caps'), 'Favourite Cap(s):' (a dropdown menu with 'Select a cap style...' as the selected option), and 'Comments:' (a text area with placeholder text 'Please provide a few words to describe your favourite cap...'). At the bottom of the form are two buttons: 'Submit' and 'Reset'. The footer of the page contains the text '© 2024 A webpage footer with important info!'.

Task9.1.1 Example form layout

In this task, your web page should use **Encapsulated JS** templates for the *main form page* and for the *thank you* response page. The template files for the main form page (broken up into `header.ejs`, `index.ejs`, `footer.ejs` sections) is provided on the Ontrack website, in the task 9.1P page through the *Resources* link found in the bottom right-hand corner of the page.

You are required to:

1. Save the provided EJS template files into a new `views` folder in your project directory (following the example provided in the Week 08 content page [8.10 Using templates](#)).
2. Make a `Node.js` application program (i.e., an `index.js` file) in your `Node.js` server folder. Add a `GET` handler for the default `/` route to render the `index.ejs` template with an appropriate `title` parameter.
3. Create a new `thankyou.ejs` template file (that can also *include* the `header.ejs` and `footer.ejs` templates to maintain consistency in presentation). The template should include fields that can be populated by arguments it is provided for the *name*, *favourite ice cream*, *rating* and *feedback*.
4. Add a new `POST` handler to accept data from the **form page**. The route address can be found in the form's `action` value. The request handler should:
 - extract the forms fields (from the *body* of the *request*)
 - render a `thankyou` template for the response page. The form's fields can be passed as parameters to the template to populate the *response* message.
5. Visit the form web page via the local `Node.js` server (e.g., `http://localhost:3000/`) using a web browser; which should render the template `index.ejs` file **NOT** a static file.
6. Enter some data into the form, like the following:

Ice Cream Review

localhost:3000

dKin Caps

A young and inspiring organisation whose primary purpose is to bring you the highest quality headwear available.

dKin Caps: Membership

Signing up to become a member of the **dKin Caps** community is very rewarding. Please use the *web-based form* below to give us your details and you will receive some great offers from us!

Membership

Please complete the form below to provide some valuable feedback on your favourite ice cream.

Firstname:

Surname:

Email:

Mobile:

Number of caps owned: ☐ No caps yet ☐ Between 1 and 10 caps ☒ Between 11 and 29 caps ☐ More than 30 caps

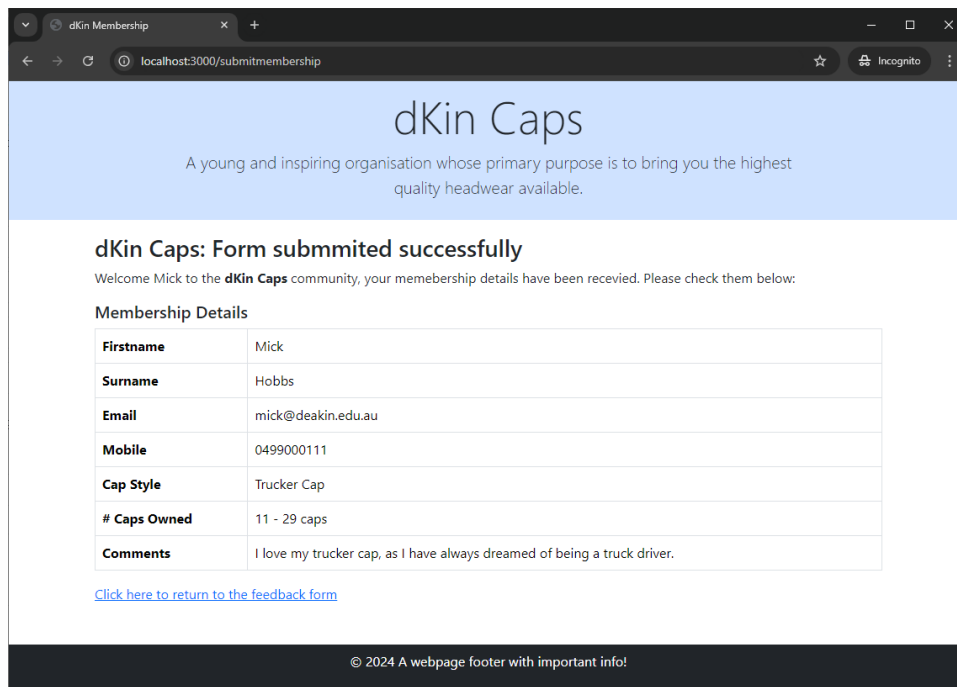
Favourite Cap(s):

Comments:

© 2024 A webpage footer with important info!

Task9.1.2 Example form with input

7. When the “**Submit**” button is clicked, then a response page (message) generated from the `thankyou.ejs` template and should be displayed in the browser, like the following:



The screenshot shows a web browser window with the title "dKin Membership" and the URL "localhost:3000/submitmembership". The page has a light blue header with the "dKin Caps" logo and a tagline: "A young and inspiring organisation whose primary purpose is to bring you the highest quality headwear available." Below the header, the main content area has a title "dKin Caps: Form submmited successfully" (note the typo in the original image). A welcome message follows: "Welcome Mick to the dKin Caps community, your memebership details have been received. Please check them below:". Below this is a section titled "Membership Details" containing a table with the following data:

Firstname	Mick
Surname	Hobbs
Email	mick@deakin.edu.au
Mobile	0499000111
Cap Style	Trucker Cap
# Caps Owned	11 - 29 caps
Comments	I love my trucker cap, as I have always dreamed of being a truck driver.

Below the table is a link: [Click here to return to the feedback form](#). At the bottom of the page is a dark footer with the text: "© 2024 A webpage footer with important info!"

Task9.1.3 Response from the server

What will you submit?

You should submit:

- Source code of the *Node.js* file (i.e., the `index.js` file).
- Source code of the *template* file for your **Thank You** for registering page (i.e., the `thankyou.ejs` file).
- Screenshot of the browser window showing the form web page with entered data.
- Screenshot of the browser window showing response message after the “**Submit**” button is clicked.