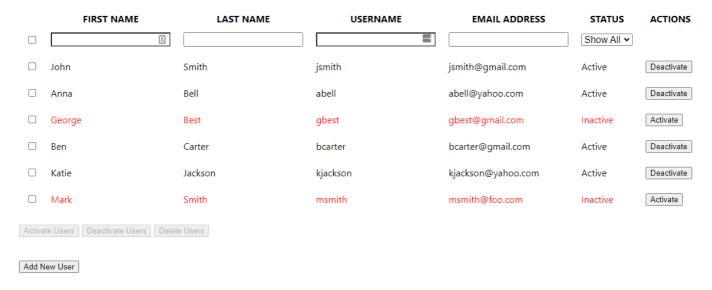# Vue.js Event Handling Exercise

In this exercise, you'll build on the previous exercise that asked you to display a list of users in a table. First, you'll start by creating a new user form to add users to the list. Second, you'll add the ability to select one or more users and then perform actions on them such as activating, deactivating, and deleting. This is what the final application looks like:

	FIRST NAME	LAST NAME	USERNAME	EMAIL ADDRESS	STATUS	ACTIONS
☐					Show All ▾	
☐	John	Smith	jsmith	jsmith@gmail.com	Active	Deactivate
☐	Anna	Bell	abell	abell@yahoo.com	Active	Deactivate
☐	George	Best	gbest	gbest@gmail.com	Inactive	Activate
☐	Ben	Carter	bcarter	bcarter@gmail.com	Active	Deactivate
☐	Katie	Jackson	kjackson	kjackson@yahoo.com	Active	Deactivate
☐	Mark	Smith	msmith	msmith@foo.com	Inactive	Activate

Activate Users   Deactivate Users   Delete Users

Add New User

Before you begin the exercise, run `npm install` to install any dependencies. Run the end-to-end tests with the command `npm run test:e2e` or `npm run test:e2e-headless`.

Step One: Implement the new user form

The markup for the new user form is already in the template. Complete the following tasks:

- Mark the form as hidden by default.
  - Create a property to toggle the visibility of the form when a user clicks the "Add New User" button.
- Bind each form field to the correct property in the `newUser` object.
- Create a method that adds a new user to the `users` array on the form submission.
  - Get an id for the `newUser` object by calling the `getNextUserId()` method.
  - Clear the form after saving the user.

After you complete step one, the tests under "Step One Tests: New User Form" pass.

Step Two: Make Activate/Deactivate button in action column dynamic

In the "Action" column of the table, there's a button that activates or deactivates the user. Complete the following tasks:

- If the user status is `Active`, the button text displays `Deactivate`.
- If the user status is `Inactive`, the button text displays `Activate`.
- When you click the button, change the user's status from `Active` to `Inactive`, or `Inactive` to `Active`.
  - Tip: Create a method and pass the user ID to it. You can use the user ID to find the user in the users array and change their status.

Once you complete this step, the tests under "Step Two Tests: Activate/Deactivate Action Button" pass.

Step Three: Activate, deactivate, and delete selected users

There are three buttons following the user listing table. Complete the following tasks associated with those buttons:

- Add a property to track the selected user IDs. Set the default value to an empty array. Bind the property to the checkboxes in the left column (except the top-most one).
- Enable the "Activate Users", "Deactivate Users", and "Delete Users" buttons only when the selected user IDs array isn't empty.
- Activate Users button
  - Sets the status of each selected user to `Active`.
  - Clears all checkboxes when action completes.
- Deactivate Users button
  - Sets the status of each selected user to `Inactive`.
  - Clears all checkboxes when action completes.
- Delete Users button
  - Deletes the user from the `users` array.
  - Clears all checkboxes when action completes.

Once you complete this step, the tests under "Step Three Tests: Activate, Deactivate, and Delete Selected Users" pass.

Step Four: Implement the "select all" checkbox

The "select all" checkbox is the checkbox on the first row with the filter inputs. Complete the following tasks for the "select all" checkbox:

- When clicked to "checked" state, set all boxes to checked and add all users to the selected user IDs array. When clicked again to "unchecked" state, set all boxes to unchecked and remove all users from the selected user IDs array.
- If some of the boxes are in a checked state, clicking "select all" selects all users. If clicked again, then it deselects all users.
- Check "Select all" when all individual checkboxes get checked.
- Add/remove the user IDs to/from the array depending upon the checked status of all checkboxes.

At this point, all tests pass, as shown here:

| | Tests | ✔ 14 | ✘ -- | ◯ -- | 08.17 | | ● ↕ | ⟳ | | ⊕ | http://localhost:8080/ | 1000 x 660 (88%) ⓘ |

All Specs

- ▼ Event Handling Exercise
  - ▼ Step One Tests: New User Form
    - ✔ should be hidden by default
    - ✔ should be toggled by Add New User button
    - ▼ Save New User Tests
      - ✔ should submit form to add to users array
      - ✔ should add the new user form values to the correct fields
      - ✔ should clear the add new user form after saving a new user
  - ▼ Step Two Tests: Activate/Deactivate Action Button
    - ✔ should change user's status when button clicked
  - ▼ Step Three Tests: Activate, Deactivate, and Delete Selected Users
    - ✔ should toggle Activate, Deactivate, and Delete buttons when rows in the user table are checked/unchecked
    - ✔ should activate selected users when Activate button is clicked
    - ✔ should deactivate selected users when Deactivate button is clicked
    - ✔ should delete users when Delete button is clicked
  - ▼ Step Four Tests: Select All Checkbox
    - ✔ should check all user checkboxes when selectAll checkbox is checked
    - ✔ should uncheck all user checkboxes when selectAll checkbox is unchecked
    - ✔ should check/uncheck all boxes if some are already selected
    - ✔ should check selectAll checkbox if all user checkboxes are checked individually

	FIRST NAME	LAST NAME	USERNAME	EMAIL ADDRESS	STA
☑					Show
☑	John	Smith	jsmith	jsmith@gmail.com	Activ
☑	Anna	Bell	abell	abell@yahoo.com	Activ
☑	George	Best	gbest	gbest@gmail.com	Inacti
☑	Ben	Carter	bcarter	bcarter@gmail.com	Activ
☑	Katie	Jackson	kjackson	kjackson@yahoo.com	Activ
☑	Mark	Smith	msmith	msmith@foo.com	Inacti

[Activate Users] [Deactivate Users] [Delete Users]

[Add New User]