

Lab: Components

Exercise 1: Splitting up components

Our contact component is bloated! Let's place the form of our page in a separate component, `<contact-form>`.

1. in `src/app/`, create a file `contact-form.component.ts` and create a new component.

```
import { Component } from '@angular/core';

@Component({
  selector: 'contact-form',
  templateUrl: './contact-form.component.html'
})
export class ContactFormComponent {
  constructor() { }
}
```

2. Also create the `contact-form.component.html`. Leave it empty for now.
3. Register this component with the module:

```
import { ContactFormComponent } from './contact-form.component';

@NgModule({
  imports: [...],
  declarations: [..., ContactFormComponent],
  exports: [...]
})
export class AppModule { }
```

4. The component will let its surrounding component know when a contact has been added by using an `@Output()` `EventEmitter`.

```
export class ContactFormComponent {
  @Output() add = new EventEmitter<Contact>();
}
```

Remember to import `Output()` and `EventEmitter`.

```
import { Component, Output, EventEmitter } from '@angular/core';
```

5. Move the form HTML to the `contact-form.component.html`.
6. Move the form validation logic and the submission handling to the `ContactFormComponent`. Only, instead of directly adding to the list, use the `Output()` variable to emit that a new contact has been added:

```
addContact() {
  this.add.emit(this.addContactForm.value);
}
```

7. In `src/app/app.component.html`, include the new component and register for the custom event.

```
<contact-form (add)="addContact($event)"></contact-form>
```

Modify this `addContact()` to correctly work with the `contact` parameter being passed in.

Our new component should now be working!

Exercise 2: Splitting up more

Do the same thing for the table on our `AppComponent` by creating a `<contact-list>` component.

Challenge

If you're fast enough, here's an extra challenge: Create a component for the `contact-row`. To do this, you will need to configure your component with an attribute selector.