

Wednesday: Deleting Goals

Deleting Goals

We earlier created a custom directive that strikes through a goal once we complete it by clicking it. We can replace the complete button in the *goal-detail* component with a delete button that deletes a goal once we decide we want to get rid of it. Let's add this button to the *goal-detail* template.

src/app/goal-detail/goal-detail.component.html

```
<p>{{goal.description}}</p>
<p>This goal will be complete in {{goal.completeDate|dateCount}} days</p>
<button (click)= 'goalDelete(true)'>Delete Goal</button>
```

We have modified the button to emit a `click` output event that calls the `goalDelete()` function which takes a boolean as an argument. We also need to change the function that we had defined in the *goal-detail* component class to our new `goalDelete()` function.

src/app/goal-detail/goal-detail.component.ts

```
...
export class GoalDetailComponent implements OnInit {
...
  goalDelete(complete:boolean){
    this.isComplete.emit(complete);
  }
...
}
```

We have used the `isComplete` event emitter that we defined earlier and emitted the boolean value passed in. We can now update the goal parent component.

src/app/goal/goal.component.html

```
...
<app-goal-detail *ngIf='goal.showDescription' [goal]='goal' (isComplete) = 'deleteGoal($event,i)'>
</app-goal-detail>
...
```

We have updated the goal-detail selector, *app-goal-detail*, with our event emitter `isComplete`, specifying that when the event is emitted, `deleteGoal()` is the function that should be executed.

Let's also define the logic for this `deleteGoal()` function which takes in the value of the event being emitted and the index of a goal.

src/app/goal/goal.component.ts

```
...
export class GoalComponent implements OnInit {
...
  deleteGoal(isComplete, index){
    if (isComplete) {
      let toDelete = confirm(`Are you sure you want to delete ${this.goals[index].name}?`)
    }
  }
}
```

```
        if (toDelete){
            this.goals.splice(index,1)
        }
    }
    ...
}
```

We have created the `deleteGoal()` function that takes in the boolean value and index of the goal. We have then created the `toDelete` boolean variable that calls the Javascript confirm function that creates a browser prompt for the user to confirm if they actually want to delete the goal. If the user confirms that the goal is to be deleted we use the `splice` function to remove the goal from the list of goals using the index. Note that while creating the confirm prompt, we have used backticks `` which allow us to use the `${this.goals[index].name}` logic inside the confirm prompt. Also, the javascript prompt gives a user two options, ok and cancel and depending on the option the user clicks on, a boolean value is returned. If the user clicks ok, the value true is returned, and if the user clicks cancel, the value false is returned.

Let's serve our application and interact with this feature we have created.