UNIT 2

Angular

Exercise week 8

Client-side Web Development
2nd course – DAW
IES San Vicente 2024/2025
Author: Arturo Bernal Mayordomo

Index

Exercise..... 3

Exercise

Update the exercise from last week (week 7) with the following changes:

- Update the app's change detection strategy to use **zoneless**:

  https://fullstackpro.es/courses/curso-angular/senales-signals#zoneless

- Create the component **event-form** and put the “event form” there. This component must manage actions related with adding a new event. Create an **output** event there called **added** where this component emits the event object created when you submit the form.

- Create the component **event-card** and move the HTML which goes inside the card there (also the CSS that affects the card). This component will have an **input** value called **event** (will receive the object to display) and an output event called **deleted** (emit when deleting the event).

  - You can use the event-card element like a div element, using CSS classes, (<div class="col..." ...> → <event-card class="col ..." ...>). Inside the component add the and its descendants.

- Format the date using Angular's DatePipe using this format: 'dd/MM/yyyy'

- Create a custom pipe called **intlCurrency** that formats a number into a currency using the Intl.NumberFormat class. This pipe will receive 2 additional parameters: currency ('EUR', 'USD', etc.) and language ('es-ES', 'en-US', etc.). Use this pipe to format the event's price in **spanish** and **Euros**.

- Add the search bar and order by date, price buttons (week 3 JavaScript exercise HTML) and do the following:

  - Filter the events using a **computed** signal. Take the search value (signal) from the search input and the array of events and return a new array containing the events that have the string in the title or description.

    - The array of events should be a **signal** now, so the computed signal reevaluates whenever it detect the array has changed.

  - The order by date and price buttons will reorder the array of events

(signal). They must **generate a new array** so the previous computed function detects that is has changed. You can use the **toSorted** method instead of **sort,** however this method is new (ES2023) so you must change the TypeScript configuration file (**tsconfig.json**) to compile to this version:

```
{
  "compileOnSave": false,
  "compilerOptions": {
    ...
    "target": "ES2023",
    "module": "ES2022",
    "lib": [
      "ES2023",
      "dom"
    ]
  },
  ...
}
```

- **Important**: The array of events is now a signal and the application is zoneless. This means that in order to detect that the array has been changed, you must return always a new reference:
  - Use the filter method to delete a event from the array. This will generate a new array that you should reassign to the original property
  - Use spread (...) to clone the array and add the new event in the same operation -> [...events, newEvent], reassign to the original property.
- Remember to delete all code (HTML and TS) from the events-page component that's no necessary anymore (and component's imports) and include the new components there!
- **Important:** Don't forget to delete **node_modules** and also the **.angular** folders before uploading the project