

One Way Binding, Events & Two Way Binding

Requisitos:

1. Al dar al botón "Generar" se elije un nombre aleatorio de un array y el resto de campos son números aleatorios del 0 al 20.
2. Solamente se puede editar el Nombre.
3. Cuando le damos a guardar se muestra en personaje guardado.
4. Tenéis que usar los 3 conceptos siguientes: One Way Binding, Events & Two Way Binding.

Código HTML

```
<body>
  <div class="title">
    <h1>Menu Principal</h1>
  </div>
  <div class="grid-layout">
    <div class="save">
      <h3>Ficha de personaje</h3>
      <form>
        <label for="nombre">Nombre</label><br>
        <input type="text" name="nombre" placeholder="Nombre del personaje"
          [(ngModel)]="personaje1.nombreElej" /><br><br>

        <label for="fuerza">Fuerza</label><br>
        <input type="number" name="nomfuerzabre" placeholder="Fuerza del personaje" [value]="personaje1.fuerza"
          disabled /><br><br>

        <label for="destreza">Destreza</label><br>
        <input type="number" name="destreza" placeholder="Destreza del personaje" [value]="personaje1.destreza"
          disabled /><br><br>

        <label for="inteligencia">Inteligencia</label><br>
        <input type="number" name="inteligencia" placeholder="Inteligencia del personaje"
          [value]="personaje1.inteligencia" disabled /><br><br>

        <label for="constitucion">Constitución</label><br>
        <input type="number" name="constitucion" placeholder="Constitución del personaje"
          [value]="personaje1.constitucion" disabled /><br><br>

        <button type="button" [value]=" 'Generar' " (click)="generadorValores()">Generar</button>
        <br><br>
        <button type="button" [value]=" 'Guardar' " (click)="guardarP()">Guardar</button>
      </form>
    </div>
  </div>
```

```
<div class="show">
  <h3>Personaje guardado</h3>
  <form>
    <label for="nombre">Nombre</label><br>
    <input type="text" name="nombre" placeholder="Nombre del personaje" [value]="personaje2.nombreElej"
      disabled /><br><br>

    <label for="fuerza">Fuerza</label><br>
    <input type="number" name="nomfuerzabre" [value]="personaje2.fuerza" disabled /><br><br>

    <label for="destreza">Destreza</label><br>
    <input type="number" name="destreza" [value]="personaje2.destreza" disabled /><br><br>

    <label for="inteligencia">Inteligencia</label><br>
    <input type="number" name="inteligencia" [value]="personaje2.inteligencia" disabled /><br><br>

    <label for="constitucion">Constitución</label><br>
    <input type="number" name="constitucion" [value]="personaje2.constitucion" disabled />
  </form>
</div>
```

Código TS

```
import { Component, } from '@angular/core';
import { Personaje } from './generatePersonaje';

@Component({
  selector: 'app-root',
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.css']
})
export class AppComponent {

  // Generación de los personajes del programa
  public personaje1: Personaje = new Personaje([''], '', 0, 0, 0, 0);
  public personaje2: Personaje = new Personaje([''], '', 0, 0, 0, 0);

  constructor() {
    this.personaje2.restart();
  }

  generadorValores() {
    this.personaje1.rebuild();
  }

  guardarP() {
    this.personaje2.save(this.personaje1);
  }
}
```

Código TS del Objeto Creado

```
export class Personaje {  
    constructor(  
        // Nombre y elección aleatoria de este  
        private Nombres: string[] = ['Neos', 'Nova Master', 'Stratos', 'Necroshade', 'Absolute Zero'],  
        public nombreElej: string,  
  
        // Puntos base del personaje  
        public fuerza: number,  
        public destreza: number,  
        public inteligencia: number,  
        public constitucion: number,  
    ) {  
        // Genera los valores aleatorios para las distintas estadísticas del personaje  
        fuerza = Math.floor(Math.random() * 20);  
        destreza = Math.floor(Math.random() * 20);  
        inteligencia = Math.floor(Math.random() * 20);  
        constitucion = Math.floor(Math.random() * 20);  
    }  
  
    rebuild() {  
        this.nombreElej = this.Nombres[Math.floor(Math.random() * this.Nombres.length)];  
        this.fuerza = Math.floor(Math.random() * 20);  
        this.destreza = Math.floor(Math.random() * 20);  
        this.inteligencia = Math.floor(Math.random() * 20);  
        this.constitucion = Math.floor(Math.random() * 20);  
    }  
  
    restart() {  
        this.nombreElej = "";  
        this.fuerza = 0;  
        this.destreza = 0;  
        this.inteligencia = 0;  
        this.constitucion = 0;  
    }  
  
    save(personaje1: Personaje) {  
        this.nombreElej = personaje1.nombreElej;  
        this.fuerza = personaje1.fuerza;  
        this.destreza = personaje1.destreza;  
        this.inteligencia = personaje1.inteligencia;  
        this.constitucion = personaje1.constitucion;  
    }  
}
```

Comprobaciones

Menu Principal

Ficha de personaje

Nombre

Edgar

Fuerza

7

Destreza

18

Inteligencia

3

Constitución

4

Generar

Guardar

Personaje guardado

Nombre

Nombre del personage

Fuerza

0

Destreza

0

Inteligencia

0

Constitución

0

Edgar Chiquillo - Visual Studio Code

Menu Principal

Ficha de personaje

Nombre

Edgar

Fuerza

7

Destreza

18

Inteligencia

3

Constitución

4

Generar

Guardar

Personaje guardado

Nombre

Edgar

Fuerza

7

Destreza

18

Inteligencia

3

Constitución

4

Edgar Chiquillo - Visual Studio Code