* El index html, llama al selector root, que es el componente principal (app.component), este llama a app.personas.component que es el component padre del proceso.
* En este padre (personas.component.html), lo que se hace es colocar las cajas de texto, y en las propiedades de estas cajas lo quese hace es por medio de [NgModel] asignar los valores capturados a unas variables de este padre llamadas nombreInput, apellidoInput, edad, con el fin de cuando se de click en agregar, el evento llama a un método que coge a nombreInput, apellidoInput y edad y crea una nueva persona en personas.component.ts, y esta persona nueva con los datos capturados se agrega al arreglo de personas. Esta persona se crea gracias a un modelo de persona (persona.model.ts), donde se estipulan los atributos que tiene un tipo persona (nombre, apellido, edad ).
* Se hace en este mismo personas.component.html un condicional, donde se verifica si hay personas agregadas, esto por medio de un boolean (personaCreada) que se enciende cuando se agrega la primera persona, si esto se cumple se llama a la o las personas que se encuentran listadas. Esto llamando a un selector persona por cada persona en lista.

 <div \*ngIf="personaCreada; else sinPersona">

                <app-persona

                \*ngFor="let personaElemento of arrayPersonas; let i = index"

                [persona]="personaElemento"

                [indice]="i"

                ></app-persona>

            </div>

Dentro del selector app-persona, una de sus propiedades es la directiva \*ngFor, lo que genera un selector por cada persona. Define la variable personaElemento que es donde se van a almacenar cada persona del arreglo, de igual manera en i se guarda cada índice de este elemento. Finalmente se le asigna a la persona (atributo de persona.component) que es la variable de entrada @Input() cada elemento, de la misma forma se le hace a el índice con la variable i.

Como ya tenemos los datos en persona.component, se procede a usar interpolación en el html de persona.component.html para mostrar los datos, pues ya se ha llamado por medio de selectores a este componente.

<div >

    {{indice+1}} Nombre: {{persona.nombre}} {{persona.apellido}} Edad: {{persona.edad}}

   </div>