**INTRODUZIONE ALLE DIRETTIVE**

**NG-BIND**

**Ng-bind ha la stessa funzione di {{expression}}.**

**Vediamo un esempio:**

**Per semplicità, consideriamo l’ultimo esempio svolto.**

**In prima battuta utilizziamo, come al solito, le espressioni tra parentesi graffe [{{expression}}].**

**Dopodiché vedremo il medesimo esempio con ng-bind**

**ESEMPIO (senza ng-bind):**

<body>

    <div id="container" ng-app>

        <div ng-init="salari=[{salario:1500},{salario:800}]">

            Mario ha un salario mensile pari a:  <input type="number" name="salario1" ng-model="salari[0].salario">

            Tote ha un salario mensile pari a:  <input type="number" name="salario2" ng-model="salari[1].salario">

        </div>

        <p>Tra Mario e Tote, chi guadagna di più è {{salari[0].salario > salari[1].salario ? 'Mario' : 'Tote'}}</p>

    </div>

    <script src="prova.js"></script>

</body>

**BROSWER:**

****

**STESSO ESEMPIO CON ng-bind:**

<body>

    <div id="container" ng-app>

        <div ng-init="salari=[{salario:1500},{salario:800}]">

            Mario ha un salario mensile pari a: <input type="number" name="salario1" ng-model="salari[0].salario">

            Tote ha un salario mensile pari a: <input type="number" name="salario2" ng-model="salari[1].salario">

        </div>

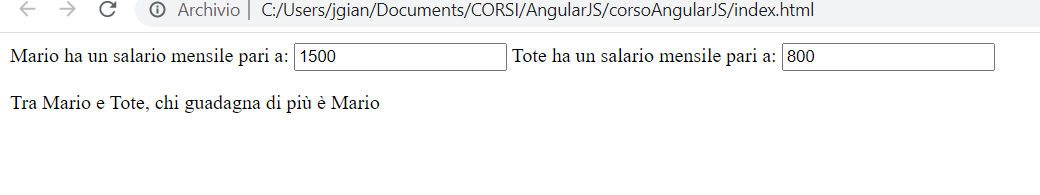
        <p>Tra Mario e Tote, chi guadagna di più è <span ng-bind="salari[0].salario > salari[1].salario ? 'Mario' : 'Tote'"></span></p>

    </div>

    <script src="prova.js"></script>

</body>

**BROSWER:**

****

**Al posto di ng-bind, avremmo potuto usare anche le seguenti direttive:**

* ngBind
* ng-bind
* ng:bind
* ng\_bind
* data:ng-bind
* x-ngBind

**Vediamo ora un esempio banale con l’utilizzo di tutte queste direttive. Vedremo che daranno tutte lo stesso risultato.**

**CODICE HTML:**

<body>

    <div id="container" ng-app>

       <div ng-init="videogame='Elden Ring'">

        <span>Inserire qui il tuo videogame preferito: <input type="text" name="videog" ng-model="videogame"></span><br><br>

        <span>Il tuo videogame preferito è {{videogame}}</span><br><br>

        <span>Il tuo videogame preferito è <span ng-bind="videogame"></span></span><br><br>

        <span>Il tuo videogame preferito è <span ng:bind="videogame"></span></span><br><br>

        <span>Il tuo videogame preferito è <span ng\_bind="videogame"></span></span><br><br>

        <span>Il tuo videogame preferito è <span data-ng-bind="videogame"></span></span><br><br>

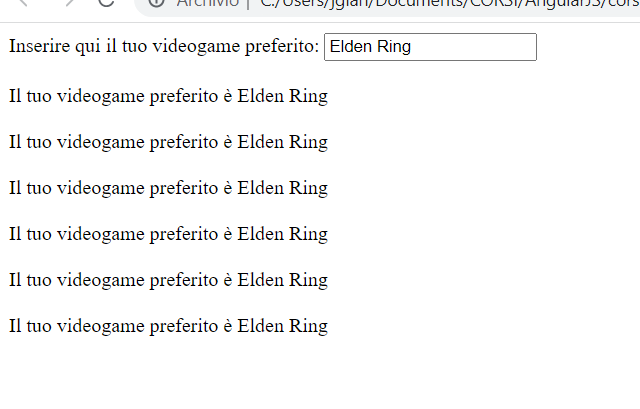
        <span>Il tuo videogame preferito è <span x-ng-bind="videogame"></span></span><br><br>

       </div>

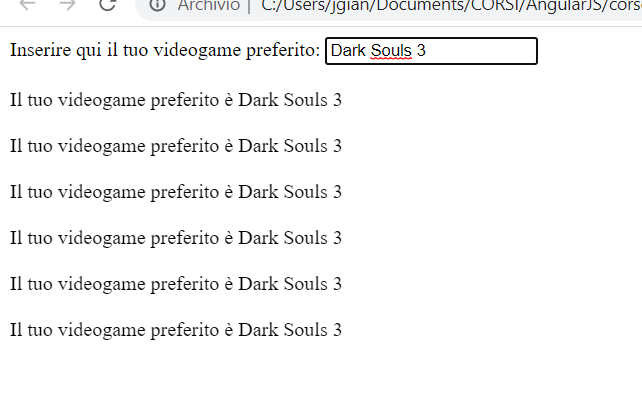
    <script src="prova.js"></script>

</body>

**BROSWER:**



**Ovviamente, cambiando il testo nell’input si modificherà anche il testo nell’ng-bind:**



**NG-BIND-TEMPLATE**

**ng-bind-template serve ad utilizzare più ng-bind contemporaneamente. Per capirne il funzionamento, vediamo un esempio senza ng-bind, ma solo con le espressioni {{expression}}. Dopodiché vediamo lo stesso esempio con ng-bind, e, infine, vediamo un esempio con ng-bind-template.**

**ESEMPIO CON {{expression}} SENZA NG – BIND**

**CODICE HTML:**

<body>

    <div id="container" ng-app="">

       <div ng-init="calciatore1={nome:'Tote',cognome:'George'}">

       Nome: <input type="text" name="calciatore" ng-model="calciatore1.nome">

       Cognome <input type="text" name="calciatore" ng-model="calciatore1.cognome">

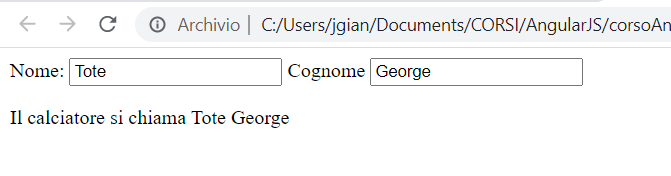
       <p>Il calciatore si chiama {{calciatore1.nome}} {{calciatore1.cognome}}</p>

       </div>

    <script src="prova.js"></script>

</body>

**BROSWER:**

****

**ESEMPIO CON ng-bind:**

**CODICE HTML:**

<body>

    <div id="container" ng-app="">

       <div ng-init="calciatore1={nome:'Tote',cognome:'George'}">

       Nome: <input type="text" name="calciatore" ng-model="calciatore1.nome">

       Cognome <input type="text" name="calciatore" ng-model="calciatore1.cognome">

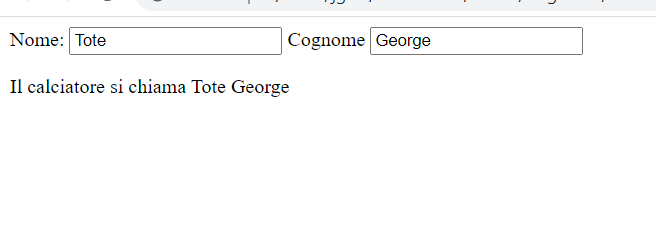
       <p>Il calciatore si chiama <span ng-bind="calciatore1.nome"></span> <span ng-bind="calciatore1.cognome"></span> </p>

       </div>

    <script src="prova.js"></script>

</body>

**BROSWER**

****

**Come visto, con l’ng-bind normale abbiamo creato due span, uno per ogni elemento presente in ng-init.**

**Vedremo con l’esempio successivo che, utilizzando ng-bind-template, sarà sufficiente un solo span.**

**CODICE HTML:**

<body>

    <div id="container" ng-app="">

       <div ng-init="calciatore1={nome:'Tote',cognome:'George'}">

       Nome: <input type="text" name="calciatore" ng-model="calciatore1.nome">

       Cognome <input type="text" name="calciatore" ng-model="calciatore1.cognome">

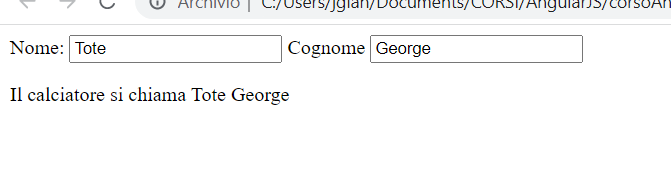
       <p>Il calciatore si chiama <span ng-bind-template="{{calciatore1.nome}} {{calciatore1.cognome}}"></span> </p>

       </div>

    <script src="prova.js"></script>

</body>

**BROSWER:**

****

**CHECKBOX**

**Le directives che vedremo ora sono ng-show ed ng-if. Queste directives permettono di visualizzare un elemento del DOM quando una checkbox è selezionata. Il contrario di ng-show è ng-hide: In questo caso, un elemento del DOM verrà nascosto quando si seleziona una checkbox.**

**Per primi vediamo ng-show ed ng-hide: dopodiché vedremo ng-if.**

**Ma quale è la differenza tra loro?**

**ng-show mostra un elemento del dom nascosto in precedenza al click sulla checkbox. Tuttavia, tale elemento ESISTE nel DOM anche se è nascosto.**

**Al contrario, ng-if**