

Bindings de dades

Introducció a Bindings de dades

El enllaç de dades es una de les coses que faciliten la programació amb AngularJS, en anglès Data Binding, essent que amb Angular tenim fins a un doble Binding.

Realment el Binding es algú que ja coneixes en Javascript comú, e inclús en llibreries com JQuery, encara que es possible que l'hagis anomenat així encara. La traducció de "binding" correspondria a "enllaç" i serveix per això justament., realitzar un nexa d'unió entre uns elements i uns altres. Data Binding es traduiria com a "Enllaç de dades".

A demés una de les característiques de AngularJS es produir automàticament el que s'anomena "Doble Binding" que ens facilita molt la nostre feina i permet estalviar-nos moltes línies de codi, que tindrien que ser afegides manualment. El Doble Binding no existeix en tots els frameworks MVC de Javascript com BackboneJS, a on només tenim un "Binding" simple.

En resum el Binding no es més que enllaçar la informació que tenim en el "Scope" amb el que mostrem en el HTML. El que permetrà classificar aquest com a simple o doble, es el fet del sentit de la consulta.

Bindings simples de dades?

Es quan la informació només flueix des de el Scope fins la part visual, en altres paraules, obtenim la informació del model i la interpretem en el HTML. Tot això es possible amb la sintaxis "Mustache" de les dos claus.

```
{{ dato }}
```

Aquest element "dato" s'obtindria des de el "Scope" i mostrant-t'ho en la pagina. La informació flueix en un únic sentit des de el "Scope", cap a la representació i oferint la característica, de que si el element "dato" (que tenim emmagatzemat en el model "Scope") varia el seu valor, aquest s'actualitzarà automàticament en la presentació (pàgina).

Doble Binding de dades?

En aquest segon cas la informació flueix des de el "Scope" cap a la part visual i també des de la part visual fins el "Scope". La integració d'aquest es realitza amb "ngModel".

```
<input type="text" ng-model="miDato" />
```



En aquest cas quan el model es modifiqui, el element “dato” que esta escrit en el camp de text (o en l’element de formulari a on s’utilitzi) s’actualitzarà automàticament amb el nou valor. A demés, gracies al doble binding en aquest cas, quan l’usuari editi el valor del camp de text, el “Scope” s’actualitzarà automàticament.

[Consultar Exercici 1](#)

Bones pràctiques en bindings de dades

A causa de la naturalesa del mateix JavaScript i de la seva manera de passar per valor versus referència, es considera una millor pràctica en Angular per enllaçar referències a les visualitzacions, amb un atribut en un objecte, en lloc de en brut per el objecte mateix.

Podem veure el [ejercici2.html](#) i [ejercici2.js](#).