**Databinding**

**Databinding** = Communication

2 Ways of accomplishing this

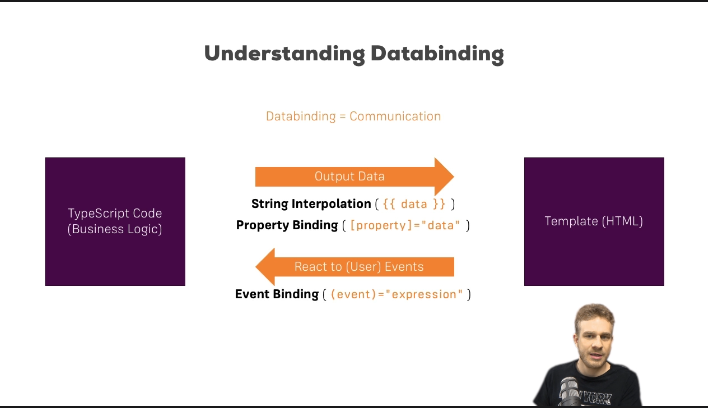
**String Interpolation** ( {{ data }} )

* Whatever is in the curly brackets can be anything as long as it **resolves to a string in the end**.

**Property Binding** ( [property]=”data” )

* E.g. [disabled]="!allowNewServer"
* String interpolation **does not work** inside of property binding.

**Event Binding**



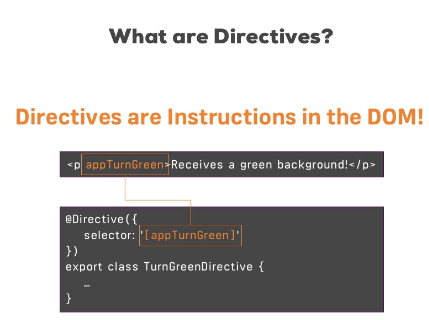
**Functions**

* **ES6 function shorthand notation**
  + **() => {}**

**Two-Way Data Binding**

* A combination of property binding and event binding
* The property will change the value of whatever it equals **AND** if that value is changed somewhere else in the code, it will also update the value of the property, hence the name two-way.
  + **Notation**: **[( property )]**

**Directives**



**Structural Directive**: Changes our DOM.

* e.g.: ng-if 🡪Adds an element to the DOM or doesn’t

**Attribute Directive**:

Does not change the DOM. They only **change the element they were placed on**.

* Useful for changing css properties dynamically.
  + You can have a JavaScript function return a property value as a string for example.

**ngStyle**: A directive that specifies the style attribute for the HTML element.

* The **attribute** must be an **object**, or an expression returning an object.

**ngClass**: Allows you to dynamically add and/or remove CSS classes.

* Also passes a JavaScript object just like ngStyle.
* **example** ngClass="{online: serverStatus === 'online'}"
  + ‘**Display** the ‘online’ class **if** serverStatus == ‘online.