**Input and Output**

Input and Output are two decorators in Angular responsible for communication between two components.

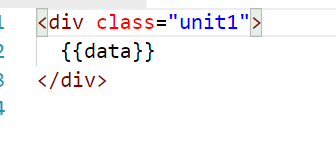
**Input**

In a component needs to receive data from another component in other words parent component, we need to decorate the variable which receives the values as input.



In the above example, the variable data is decorated with @Input, and in the template

We print data

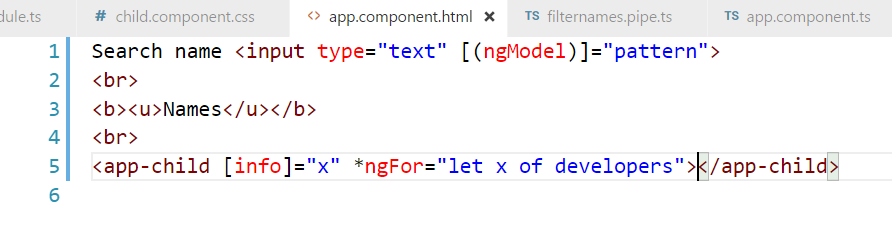


In the parent component we pass the data child as follows

The class file of parent component appears as follows



And we pass the value to the child in the following manner



It yields the following output



**Output**

Angular is based on a one-directional data flow and does not have two-way data binding. So, how do you get a component to emit an event to another component?

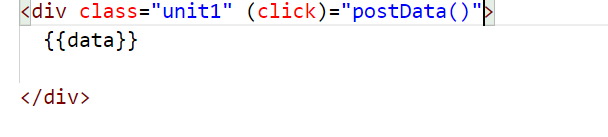
In Angular, a component can emit an event using **@Output** and **EventEmitter**. Both are parts of the **@angular/core**.

If a Child wants to communicate something the parent then we need to have a event emitter in the child declared as follows



In the above code, we have a variable called emitInfo which if of type Event Emitter and when the function postData is called , it emits data

Lets see how the postData function is called. In the template,

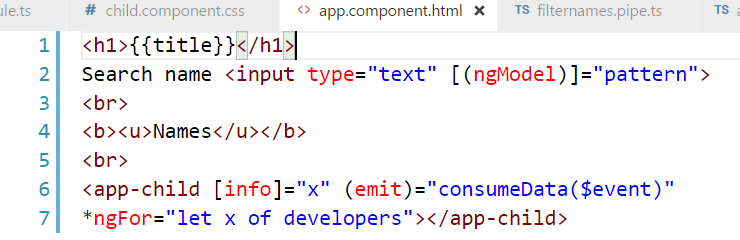


When the child template is clicked , we call post data

To subscribe to this even emitter , we do following changes in parent



In the template , you can see



We have subscribe to the event called emit using output directive notation. When we click division with name girish, it changes the title in the parent

