When we build app, one way to help with the testing aspects of coding and keeping the code saleable is to make it as modular as possible. As we make the applications more and more modular, the individual components need to pass data between themselves. Angular allows you to bind data to different parts of the application, as with property binding. “The difference in creating our own properties to bind to (as opposed to a built-in property) is we have to *tell* Angular the name of the property binding, essentially exposing it for us to bind to. (Motto, 2019)”

The way a parent component communicates data to its child components is through the @Input decorator. “@Input links a property of a component (which is generally the child component) with a value that was given by another component (the parent). (Aarab, 2019)” A real-life example would be like a parent leaving a message for their children on a marker board in the kitchen.

The way a child component communicates data to its parent is through the @Input decorator.

“@Output decorator is used to link a property of a child component and emit it through the event emitter. (Aarab, 2019)” The real-life example would be like a child yelling out what they want the parent to know, and the parent has to be listening at that time.

The passing of data between different parts of the application is a vital part of making the application work, but it also allows us to keep the separate parts small and contained. Each part does not have to care how or who gets the message, only that the message is available for them when they need it. It is up to the component to get the message and act upon it.

References:

Aarab, H. (2019, April 8). *Angular: Component interaction with @Input, @Output and EventEmitter*. Retrieved February 6, 2020, from https://blog.hackages.io/angular-component-interaction-with-input-output-and-eventemitter-72526422b95c

Motto, T. (2019, September 3). *Passing data into Angular components with @Input*. Retrieved February 6, 2020, from https://ultimatecourses.com/blog/passing-data-angular-2-components-input