Typescript and Angular

1. Typescript = “a static typechecker, that is a superset of JavaScript”
   1. Properties
      1. When you’re writing in TS, you’re writing in TS but with a stricter ruleset “layered”/placed on top
      2. TS is transpiled, meaning 2 things:
         1. Browsers don’t understand TS
         2. A TS source file is converted to a JS source file
         3. In general, transpilers are “source-to-source compilers - a subset of compilers which intake source code files and convert to another **source code file** in another language **or** a different version of the same language.” [ES6 to ES5]
      3. TS makes use of duck-typing
         1. “If it walks like a duck, looks like a duck, quacks like a duck…it’s a duck”
         2. If two objects have identical properties, then TS will view both as one & the same.
         3. It’s like when viruses/pathogens ‘bind’ to some of your immune system cells
      4. Type Annotations/assertions
      5. Type definition
      6. Erased types
      7. Rest parameters
      8. Type script modules
         1. Analogous to “using” blocks in C#
         2. Key differences are that you must import/export implicitly
   2. Str
   3. Lim
      1. Whenever you make a new TS app, you need to install TS into the workspace of that app
2. Angular = “Angular is a JavaScript framework for creating efficient and sophisticated single-page applications”
   1. Properties
      1. Angular utilizes Ahead-of-Time compilation
      2. Angular workspaces
      3. Angular forms = “gather user input , validate input, create a form & data model to display”
         1. Reactive vs Template-driven forms
            1. Reactive is best when you will be using tons of forms
            2. Template-driven is best when you can manage the form update in a template
         2. Angular forms differ by their form-control implementation
            1. Reactive forms are centered around observable streams

Observables are basically “things that return to subscribers asynchronously ” & Angular uses RxJS to implement them

* + - * 1. Template forms are simple Angular templates (think about how MVC views has templates based off models)
      1. Similarities:
         1. Created with the same 4 base classes

FormControl

FormGroup

FormArray

CVA

* 1. Str
  2. Lim

1. Reactive Programming = “a paradigm that centers completely around asynchronous data streams, and the propagation of respective changes in an application”