

Welcome to Angular Class!

<http://moxie.pro/2019april8ng>

[System source checklist:](#)

[Connect to remote machines:](#)

[Get access to course repositories](#)

[List of chapters](#)

[Tech Resources:](#)

[Other Notes and Links:](#)

[Music Requests:](#)

[Review Session Tuesday morning](#)

[Review Session Thursday morning](#)

if your machien becomes slow - try this to restart

Shutdown -r -t 0

System source checklist:

- Set resolution on monitors
- Open Command Prompt

Connect to remote machines:

- [HTML 5 Remote Lab Connection PDF](#) (Use for class)
- You can reach this document from the ProTech machine:
- <http://moxie.pro/2019april8ng>
- List of machine assignments: [link](#)
  - Once you are connected - add your name next to your assigned machine

Get access to course repositories

This is very much a hands on course.

You will be using modern tools and techniques to create applications.

You will be committing code to GitHub.

Visit <https://GitHub.com/> and sign into an existing account of yours or sign up for a new account.

Do this from c:\

git clone <https://github.com/JudyLipinski/angular100-demos>

A project with people used as a demo

Runs on port 5555

git clone <https://github.com/JudyLipinski/angular100-labs> [CLICK HERE](#)

Needed for instructions

git clone <https://github.com/JudyLipinski/angular100-albums>

Solutions for the exercises, by branches

Port 4444

YOUR repo. Runs on port 3333

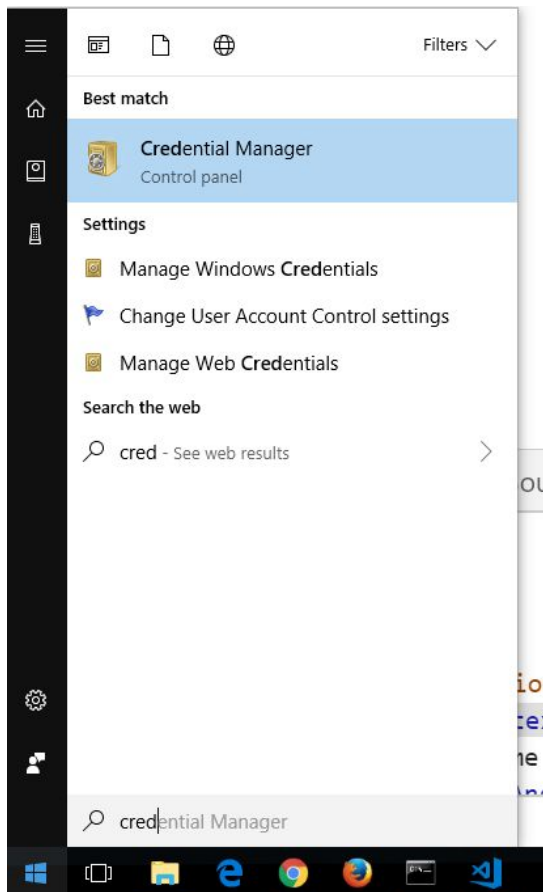
List of chapters

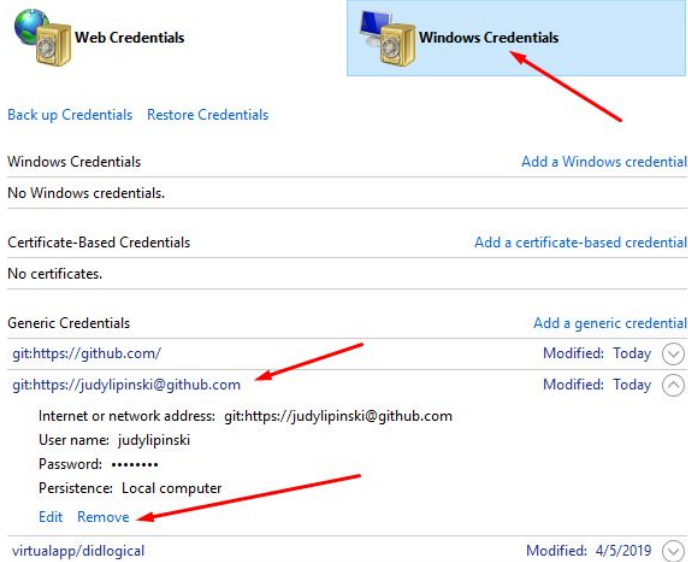
	<a href="#">Overview of Course and Angular</a>
1	<a href="#">Development</a>
2	<a href="#">Creating and Exploring Angular Projects</a>
3	<a href="#">Data Binding in Components</a>
4	<a href="#">Passing Data &amp; Event Binding</a>
5	<a href="#">Using Directives</a>
6	<a href="#">Using and Creating Pipes</a>
7	<a href="#">Understanding Change Detection and Lifecycle hooks</a>

8	<a href="#">Services, HttpClient and Observables</a>
9	<a href="#">Leveraging Routing Features</a>
10	<a href="#">Working with Forms</a>
11	<a href="#">Unit Testing in Angular</a>
12	<a href="#">Summary</a>

## Tech Resources:

- <https://git-scm.com/book/en/v1/Git-Tools-Stashing>
- <https://git-scm.com/docs/git-stash>
- <https://www.typescriptlang.org/play/>
- <https://augury.rangle.io/>
- <https://www.npmjs.com/package/json-server>
- <https://stackblitz.com/edit/angular-b5r2gc>
- Changing credentials that have been stored with wincred
- 





- Ways to quickly select and work with text in VSCode:  
<https://ihatetomatoes.net/vscode-tips-select-everything-between-brackets-quotes/>

- **Example of shallow copy in action with slice()**

```
obj1: any = { name: "John" };
obj2: any = { other: this.obj1 };
```

```
objArray = [this.obj1, this.obj2];
let newArray = this.objArray.slice();
// this.objArray[0].name = "Mary";
newArray[0].name = "Sam";
console.log(this.objArray[0]);
```

- 

delete the TS lint file in the src folder.	<a href="https://stackoverflow.com/questions/52525947/use-of-the-2-tslint-json-file-in-angular-6-project">https://stackoverflow.com/questions/52525947/use-of-the-2-tslint-json-file-in-angular-6-project</a>
--	---

- 

#### Other Notes and Links:

- <https://lolsnaps.com/funny/2857660>
- [https://www.wired.com/story/scientists-reveal-the-first-picture-of-a-black-hole?utm\\_campaign=wired&utm\\_medium=social&utm\\_source=facebook&mbid=social\\_fb&utm\\_brand=wired&utm\\_social-type=owned&fbclid=IwAR0mwwDLEgksvN3eAOwBOK7EFnLOJcpbTioX7tx57-gjAFGIOnEYXzhceP0](https://www.wired.com/story/scientists-reveal-the-first-picture-of-a-black-hole?utm_campaign=wired&utm_medium=social&utm_source=facebook&mbid=social_fb&utm_brand=wired&utm_social-type=owned&fbclid=IwAR0mwwDLEgksvN3eAOwBOK7EFnLOJcpbTioX7tx57-gjAFGIOnEYXzhceP0)

- Put the avocado pits in your guacamole and it wont turn brown
- On FreeNode join room #wplug (the western PA Linux User Group, also see [www.wplug.org](http://www.wplug.org))
- <https://www.amazon.com/L-Laxman-Angular-Tutorial/dp/B07JX44NL5>
- <https://www.ssa.gov/hqalert>
- **GRUB HUB** COUPON: <http://fbuy.me/IB8Db>

## BE IMPECCABLE WITH YOUR WORD

---

Speak with integrity. Say  
only what you mean.  
Avoid using the word  
to speak against yourself  
or to gossip about others.  
Use the power of your  
word in the direction  
of truth and love.

## DON'T TAKE ANYTHING PERSONALLY

---

Nothing others do is  
because of you. What others  
say and do is a projection of  
their own reality, their  
own dream. When you are  
immune to the opinions  
and actions of others, you  
won't be the victim  
of needless suffering.

# THE FOUR AGREEMENTS

## DON'T MAKE ASSUMPTIONS

---

Find the courage to ask  
questions and to express  
what you really want.  
Communicate with others  
as clearly as you can to  
avoid misunderstandings,  
sadness, and drama. With  
just this one agreement,  
you can completely  
transform your life.

## ALWAYS DO YOUR BEST

---

Your best is going to  
change from moment to  
moment; it will be different  
when you are tired as  
opposed to well-rested.  
Under any circumstance,  
simply do your best, and  
you will avoid self-judgment,  
self-abuse, and regret.

### Music Requests:

- Vitamin String Quartet
- Barber: Adagio For Strings, Op.11/2 · Baltimore Symphony Orchestra · David Zinman
- [Filk List](#)
- [Folk](#)
- [Sad songs](#)
- [Music from Rein](#)
- [Spring music](#)
- (not for playing in class but...) [The original Hitchhikers Guide to the Galaxy BBC radio Drama](#)
- [Random 70's and 80's music \(CCR, Eagles, America, etc\)](#)

### Review Session Tuesday morning

- What was something new or that stood out for you yesterday?
- Do you have any questions?
- Getting GitHub to sync worked!
- Extensions for Angular
- Review of package.json was helpful
  - And package-lock.json usage
- Finding out that bash isn't working like the command prompt
  - With angular prompting questions about routing / CSS
- Using **ng new** to create a new project
- VSCode new to some, extensions
- Purpose of files in Angular
- Pretty much everything
- Good review of ES6 !!
- Architectural view of Angular - seeing how it all fits together
- Shortcuts to make life easier !
- Upgrading Angular 6 -> 7 using the website to help with steps

**Good Morning! Happy Wednesday!!!**

**If you are all caught up with labs - be sure to commit your work so far and push to your repository.**

- What was something new or that stood out for you yesterday?
- Do you have any questions?
- Working with timers, setInterval setTimeout
  - Working in classes, be careful with anonymous functions that refer to this they will break if inside of a timer - so the workaround is to use - arrow functions `function() { console.log(this.thing) } -----> () => {console.log(this.thing)}`
- Data Binding (2 ways)
  - {{ interpolation }}
  - Dont try to change the data - just displaying
  - Call a function? YES {{ getFullName() }}
  - [propertyBinding]="expression"
  - <p [hidden]="property or expression">
- Change detection
  - Triggers for change detection
  - Zone.js is used to rewrite a lot of the browser JS
  - Event generation (clicking), asynchronous callbacks (AJAX, timers)
- Augury tool for debugging
- Template Expressions
- Using a model to validate types for the data
  - Use of interface to indicate types
  - Using ? in case the data is undefined
- Use of Git
- Adding bootstrap for styling
  - `npm i bootstrap --save (-S)`
  - Modify angular.json to let Webpack know about it
- Formatting - using Prettier
- Adding new items to an array doesn't trigger change detection, but modifying an object property does
  - `this.myArray = [...this.myArray]`



**Good Morning! Happy Thursday!!!**

**If you are all caught up with labs - be sure to commit your work so far and push to your repository.**

Review Session Thursday morning

- What was something new or that stood out for you yesterday?
  - Do you have any questions?
  - Create new components
    - ng g c component-name
      - creates component.ts, component.html, component.css, component.spec in a subdirectory of the same name
      - --flat doesnt create a component-name directory
      - --it inline template rather than templateUrl
      - --is inline style rather than separate
      - If want in a subdir - ng g c **subdir**/component-name
    -
  - Event Binding
    - <button (click)="functionHandler()">Click Me</button>
    - <button (click)="toggleValue=!toggleValue">Click Me</button>
  - Passing Data from parent to child - property binding
    - in child ts file
      - @Input nameInChild
      - @Input anotherProp
    - in parent template
- <app-child [nameInChild]="objectInParent"  
[second]="anotherProp" >
- Child to Parent (Event Binding)
    - In Child

- @Output() **customEventName**:EventEmitter<string>= new EventEmitter<string>();
    - in function at some point: customEventName.emit()
  - In Parent
    - <app-child [nameInChild]="objectInParent" [second]="anotherProp" (customEventName)="parentFunctionHandler()" >
- Directives
  - \*ngSwitch if you have multiple choices
  - \*ngFor
    - trackBy: function-name
    - Angular selectively update the DOM, only what is changed, and it compares the new array to the old using what is in the trackByFunction
- Communicate between sibling components
  - Pass up using event binding to parent, and then down to other sibling
- [hidden] versus \*ngIf
  - If using hidden, it is still in the DOM (memory)
    - If toggling a view
    - Especially if ngOnInit() has to do a bit of work - talk to server, setup timers
  - \*ngIf manipulate the DOM add/remove
    - Not likely to be needed, otherwise it takes up memory
    - ngOnInit may start a timer, and if hidden, still running, unless you use ngOnDestroy

Poll Question:

are you okay if we use a diffuser?

if not, just put no here and we wont. ⇒

Challenge Project:

WeatherApp

Use the api from <https://openweathermap.org/current>

- First look at the API
- Get some cities (4 of them)

- Hard-code an array in WeatherList
- Display data in a WeatherCard
  - Can use weather icons
  - \*ngSwitch
- Service with HttpClient to get the live data

019-04-11T15:11:25.872Z - Hash: 69dd3ae726e240fee281 - Time: 1121ms

nged chunks

main} main.js, main.js.map (main) 11.9 kB [initial] [rendered]

: Compiled successfully.

Activate Windows  
Go to Settings to activate Windows.

Ln 30, Col 24 Spaces: 2 UTF-8 CRLF TypeScript 3.4.1 Prettier: ✓

11:11 AM  
4/11/2019

PS C:\angular100-demos> npm start

> angular100-demos@0.0.0 start C:\angular100-demos

> ng serve --port 5555 -o

■

Activate Windows  
Go to Settings to activate Windows.

PS C:\angular100-demos> npm start

> angular100-demos@0.0.0 start C:\angular100-demos

> ng serve --port 5555 -o

\*\* Angular Live Development Server is listening on localhost:5555, open your browser on  
http://localhost:5555/ \*\*

25% building 27/28 modules 1 active ...ode\_modules\@angular\core\fesm5\core.js

Activate Windows  
Go to Settings to activate Windows.

From scratch exercises:

- AboutMe app
  - Load friends, family, hobbies from a JSON file using json-server
- Overview of MyWeatherApp
  - Create a display of the weather for 5 cities
  - Use a List with an array of objects
    - each object with city name and weather as described by calling this API
    - <http://api.openweathermap.org/data/2.5/group?id=524901,703448,2643743&units=metric>

- Use a card to display
- Get data to start with from <https://openweathermap.org/current>
  - You will need to create an API key
- - Use HttpClient to get

**Good Morning! Happy FRIDAY!!!**

**If you are all caught up with labs - be sure to commit your work so far and push to your repository.**

Review Session Thursday morning

- What was something new or that stood out for you yesterday?
- Do you have any questions?
- Pipes
  - Variable in component.ts **today**
  - In template {{ today | date: 'd/M/yy' }}
  - Currency, slice, decimal
  - Create custom with AngularCLI: ng g p name ⇒ NamePipe
    - Transform needs to be implemented
- Used json-server to create a local server
  - Could then practice using an API
  - Ssa uses wiremock
  - Judy to look into if it can be configured to do custom objects
- | async
  - Subscribe to the observable
    - Necessary for the requests to be sent to the server, etc
  - Destroy it when component is unmounted
- If not using async pipe
 

```
methodThatReturnsObservable()
  .subscribe (
    successFunction(receivedData),
    errorFunction(receivesError)
  )
```

  - The HttpClient methods return observables
    - http.get() needs to be subscribed

- How to dependency inject into a service / component / pipe?
  - constructor(localName: WhatYouWantInjected)
  - In our Service: constructor(http: HttpClient)
  - How do we make services available to the project?
    - Option 1: @NgModule() { providers: [ServiceNames, etc] }
    - Option 2: In the Service use @Injectable('providedIn: 'root')