Notes

1. SDLC
   1. Properties
      1. Waterfall Model = breaks down the project into phases that cannot be progressed until each phase is successfully completed
         1. Requires a lot of meetings, documentation, etc, because there cannot be any movements backwards.
         2. Inflexible and very non-iterative, usually too stringent
      2. Spiral = Mini waterfalls that iterate upon the waterfall
         1. Each loop is called a phase and each phase has its own quadrants
         2. Flexible, great for large projects, customer satisfaction from tangibles each spiral
         3. Complexity, expensive, a completed project has unknown number of cycles
      3. Iterative = small iterative dev cycles, which allows learning from each previous stage
      4. V =
      5. Big Bang =
      6. CMMI =
      7. Agile =
   2. Str
   3. Lim
2. Containerization
   1. Properties
   2. Str
   3. Lim
3. Docker =
4. DevOps = the union of people, process, and products to enable continuous end value to end users
   1. Properties
      1. Replaces siloed/segregated development team and operations team by integrating into a unit consisting of multidisciplinary teams for more shared & efficient tools (combination of IT, dev, and Business teams)
         1. Agile planning
         2. Continuous integration
         3. Continuous delivery
         4. Continuous monitoring
      2. The purpose of DevOps is to shorten the cycle time in the OODA loops by streamlining the feedback process through automation + frequent deployments, thus enabling more validated learning
      3. Continuous Integration and Continuous Testing
         1. CI
         2. C
      4. OODA Loop = a pattern for thinking and approaching how to solve a business need. It occurs in a cycle time
         1. **Observe** business needs
         2. **Orient** with potential business options
         3. **Decide** what you will offer
         4. **Act** by delivering software goods & services to customers
         5. Feedback gathered from each cycle should be meaningful, actionable data to facilitate **validated learning** (basically learning that is empirically founded)
      5. Azure DevOps
         1. A cloud service to allow code testable to users
         2. Build Definition = mechanism controlling how & when build occurs
         3. Uses a .yml files
   2. Str
   3. Lim
5. YAML = “Yet another Markup Language”
   1. Properties
      1. Indentation specific
      2. Easily readable by humans
      3. Portable between languages
      4. Match native structures of OOP/Agile langs
      5. 1 pass processing
      6. Expressive and extensible
      7. Easy to implement
      8. Starts through a trigger activation in the repo
         1. They are what allow auto build of YAML
         2. Push trigger
         3. No push trigger
      9. Cannot have a stage in a job – things must be ordered to be stage specific
6. Cloud Basics =
7. Sonar
   1. Technical debt = estimated time required to fix all the problems
   2. Code smell = characteristic in the cod that indicates a potential problem
      1. What constitutes a potential problem is subjective
      2. Usually indicates issues with long-term maintainability of the code – it can be improved to ensure future developers have a minimal risk of creating an error DUE TO THAT CODE
   3. Duplication = locations where identical code is located & ought to be their own methods
   4. Quality gates and Fixing the water leaks