* Course Overview
  + Installing docker(docker for windows vs docker on windows)
  + Command line
  + Host static web site
  + Build images
  + Databases in containers
  + Web application and databases with docker compose
* Installing Software Is Often Pai..
  + Docker simplifies software interaction
    - Installing, starting, stoping, building, etc
* Challenges Discovering Software
  + Find correct version
  + Then download it
  + Extract it and run installer
  + Run it
* Challenges with Software Installation
  + OS/Build compatibility
  + Format
    - Source, executable, executable + libraries, executable + libraries + runtime
  + Installers, package managers, manual install
* Challenges with Running Software
  + Documentation
  + Path of software
  + Staring, stoping
  + Service registration
  + Licenses
  + Installing and running dependencies
  + Security and sandboxing
  + Breaking changes updates
* Containers Are About Software
  + Docker and containers are about running software
  + Image: packaged up application
    - Kinda like a zip file
* Installing and Running MongoDB
  + Traditional
    - Search for mongodb
    - Download right version
    - Run installer
    - Set up mongodb environment
    - Make data directory
    - Then run mongod.exe
    - Run mono.exe
* Finding Software on Docker Hub
  + Search for mongodb docker
  + Click on ‘library/mongo’
  + Pull the image(docker pull)
    - Contains application and dependency
* Downloading Software with docker
  + Type ‘docker pull mongo:windowsservercore’
  + Instead of installing the software we create container
    - A stopped container is like installed software
  + Take image and run it in a container
  + Container: like installed software
    - Install step
  + Running container
    - Run step
* Running Software in a Container
  + ‘docker run’ does all the steps above
    - It will find and pull image
    - It will spin up container
    - Then run container
  + Type ‘docker run [image]:windowsservercore’
    - Windowsservercore specifies a window container
  + Command prompt will be attached to container
    - Locked up with the output of the running process
  + Type ‘docker exec -it 27 mongo’
    - To connect to mongo server
  + ‘docker create’ if you just want to create a container
* A Running Container is Just A
  + Find the software on docker hub
  + Download the software with docker pull, comes down as an image
    - Much like a zip file or msi installer
    - An image is an application packaging format
  + Instead of installing the software, we create a container
    - A stopped container is like installed software
  + Can use ‘docker create’ to just create a container
  + Then we run the container
    - Running the container is exactly like running an exe
  + We often use ‘docker run’ to orchestrate all these steps with on command
  + ‘docker exec’ can be thought of as running another copy of our installed software
    - After container is create and running, we can use docker exec to run multiple applications or multiple copies of the same app inside the container
  + Type ‘docker exec -it 27 powershell’ to run powershell inside of hte container
  + A container is like process namespace
  + Application launched via container is the same as if we had launched the application normally
  + Containers are special ways to work with them software/applications
  + A running container contains one or more running applications
* Stopping Apps in a Containers
  + Type ‘docker ps’ to look at running processes
  + Type ‘docker stop [container id]’ to stops the running software in the container
* Why Docker and Containers
  + Docker hub has all the software we can use
  + Repository hold images
  + Stats about software
  + Can be trusted, official images
  + Can scan through tags for vulnerabilities
  + Only have to worry about linux vs windows container
    - Don't have to worry about os
  + Docker file shows commands to create image
  + Docker is a way to manage software
* Installing Docker for windows
  + Download ‘Docker for Windows’
    - Mostly for desktop or workstation environment
    - Not for server environment
  + Needs windows 10
  + Hyper-v package must be enabled
  + Docker uses hyper-v to run containers
  + Need hardware virtualization
  + Go to task manager
    - Performance tab
    - Lower right ‘virtualization’ should be enabled
  + If virtualization not enabled then you will have to enable in the bios
    - Google instructions on how to do so
  + Update windows
  + Download the beta for windows containers
* What Docker for Windows Install
  + Open Hyper v Manager
    - Will be installed along with docker
  + It will set up hyper v vm that contains linux so we can run linux containers
  + ‘docker info’: information about docker server
  + ‘docker version’ information about docker client
  + Docker client will use windows os
  + Docker server will use linux os
    - Runs on VM
  + Docker for windows is a wrapper around client and server
    - Manages everything for our windows environment
* Running Linux Containers on Windows
  + ‘docker ps’: show running processes inside of docker
    - ps: process status
  + Containers are just running applications
  + ‘docker run hello-world’
    - Hello-world is the image we will download and take and create container and run it
    - Simple application that runs then exits
  + ‘docker run -p [host port]:[container port] nginx’
    - Since its a web server, will need to specify port number
  + Takes downloaded image and extracted locally
  + Images are layered
* Docker Inverts Learning
  + Quickly launch software with one command
    - No need to know how to set it up
    - Inverted learning
  + Inverted learning: use software without knowing how to set it up
* Stopping Containers Means St
  + Use “ctrl + p + q” to detach from container
    - Application still running, just go to cmd
  + When you create a container, each is given a random name unless you give it a name
  + Each container is also assigned a randomly generated id
  + Name or id can be used to interact with and control the container
  + Type ‘docker stop [name or id]’ to stop container
    - Only need to use enough of id to be unique
* Restarting Containers and doc
  + Type ‘docker start [name or id]’ to restart container
  + Type ‘docker ps -a’ to list all containers, even stopped ones
  + Starting and stopping containers does not destroy container
  + Type ‘docker run [image]’ will create a new container each time
* Removing Containers is Akin to..
  + Removing containers is like uninstalling software
  + Each container is like one installed software
    - 3 containers, 3 installed software
  + Type ‘docker images’ to list images we downloaded
  + Images are application packages, like zip files
  + Containers are the installed application
  + Removing containers does not remove the image
  + Have to stop containers before we can remove it
  + Type ‘docker rm [name of container or id]’ to remove the container
  + Type ‘docker rm [name of container or id] [name of container or id] ..’ to remove multiple containers
* Removing Images is Akin to De..
  + Type ‘docker’ or ‘docker --help’ to show list of commands and what they do
  + Type ‘docker rmi [image name/repository or id]’ to remove image
    - Removes the layers of images
* Running the Docker Docks in a ..
  + Type ‘docker help search’ for instructions on how to search the docker hub
  + Type ‘docker search docs’ to get back list of repositories
  + Can do same thing on docker hub website
  + Repository can have multiple tags that link to multiple images
  + Type ‘docker run [images]:[tags]’ to download specific version of image
* Docker run --it and --name
  + ‘docker run -p [host port]:[container port] -it --name [name] [image name]’
    - -it: flag opens an interactive terminal into the container
    - --name: flag sets the name of the container
  + Add the -it flag if you want to run containers in the foreground and be able to stop the container
    - Commands are sent to the running process inside of the container
  + Type ‘ctrl+p+q’ if you want to detach but not kill the process
* Switching to Windows Containers
  + Right click on docker icon
  + Choose ‘Switch to Windows containers’
    - Docker for windows will set up windows containers on your computer
  + Will need a new containers features
    - Need windows 10 anniversary, pro, enterprise
* Running IIS in a Windows Container
  + Windows containers are completely isolated from linux containers
  + IIS is windows equivalent of nginx on linux
  + ‘docker run -p 80:80 -d microsoft/iss:nanoserver’
    - -d: tag to detach from container
    - Container will run in the background
  + Type ‘ipconfig’ to get info about ip
  + Grab ipv4 address
  + Use another computer on the network and connect to the ipv4
* Containers Have Their Own Isolation
  + Can go directly to container
  + Process are isolated, so it looks like it is running on a completely isolated computer
  + Container has its own network adapter
  + Can ping network adapter via its associated ip address, can get into container that way
  + ‘docker inspect [container id or name]’ gives information about container
  + Can use http:// + ip address : + container port to access the container
  + ex) <http://127.17.0.1:4000>
  + Windows containers have trouble accessing with localhost but linux containers are abe to just fine
* Key Takeaways
  + Containers and docker is all about helping you run software
  + Find software = docker hub
  + Download software = pull an image = docker pull
  + Install software = create a container from image = docker create
  + Start software = run the container = docker start
  + Stop software = stop the container = docker stop
  + Uninstall software = remove container = docker rm
  + docker run does all with one command(not stop, or uninstall)
  + docker exec to run another instance of software or a different instance in the same container
  + Docker for windows is not meant for server environment
  + Run windows containers for window servers
  + Run linux containers for linux servers
* How Docker for Windows Sup..