walkthrough

January 21, 2024

1 Module-01 Walkthrough

Start-to-Finish setup and concept. *ellacharmed, 2024.01.21*

intro_notes

hi. so nervous, I wrote a script! imposter syndrome - who AM I? to do such a walkthrough. but still one week to homework submission so I thought it may still be of some help to somebody.

this module is partition into 2 parts. can start with either one, sequence does not matter. in fact, the repo and video playlist are in opposite order with repo having terraform first

1.1 Goals of Module-01

- docker_sql
  - setup and access database using containers
  - populate database by ingesting data NYC Taxi 2021 yellow
  - **query database** <— main focus

1.2 Concept: container (1)

- why?
  - local experiments, reproducibility, isolation, integration test (CI/CD), automation (pipelines)

1.3 Concept: container (2)

- how?
  - using config file(s), docker hub registry

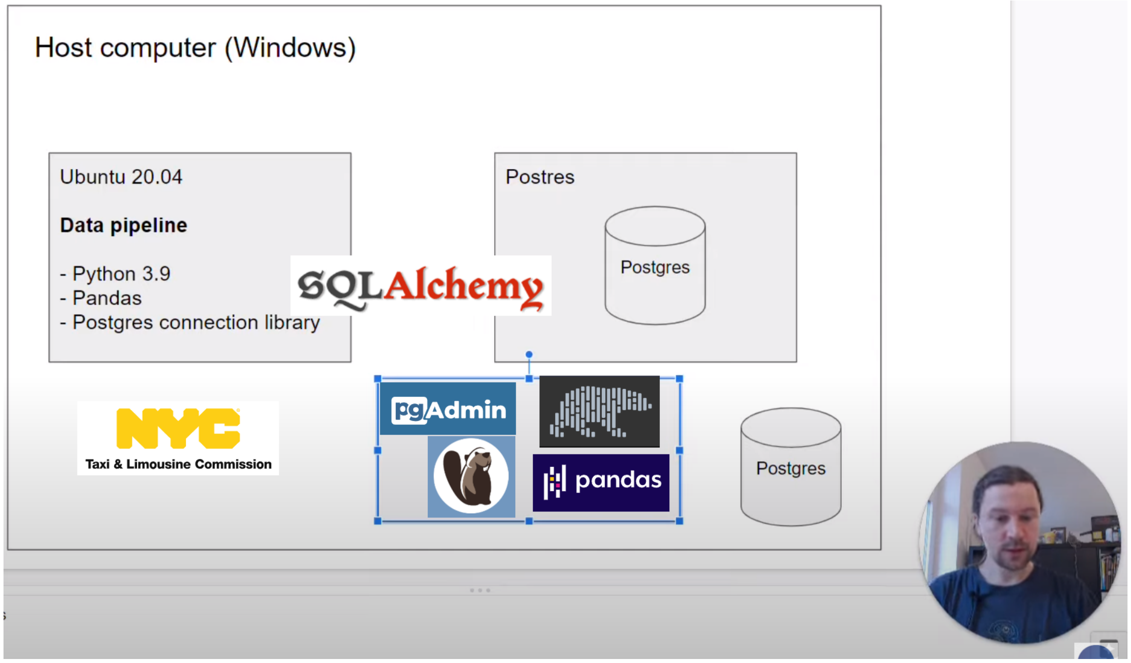
1.4 Concept: container (3)

- Dockerfile vs docker-compose
  - Dockerfile : standalone, unit tests
  - docker-compose : recipe, integration

1.5 Concept: database

- server, client
- ports
- practice with postgres

1.6 Concept: db diagram


Host computer (Windows)

Ubuntu 20.04

Data pipeline

- Python 3.9
- Pandas
- Postgres connection library

Postres

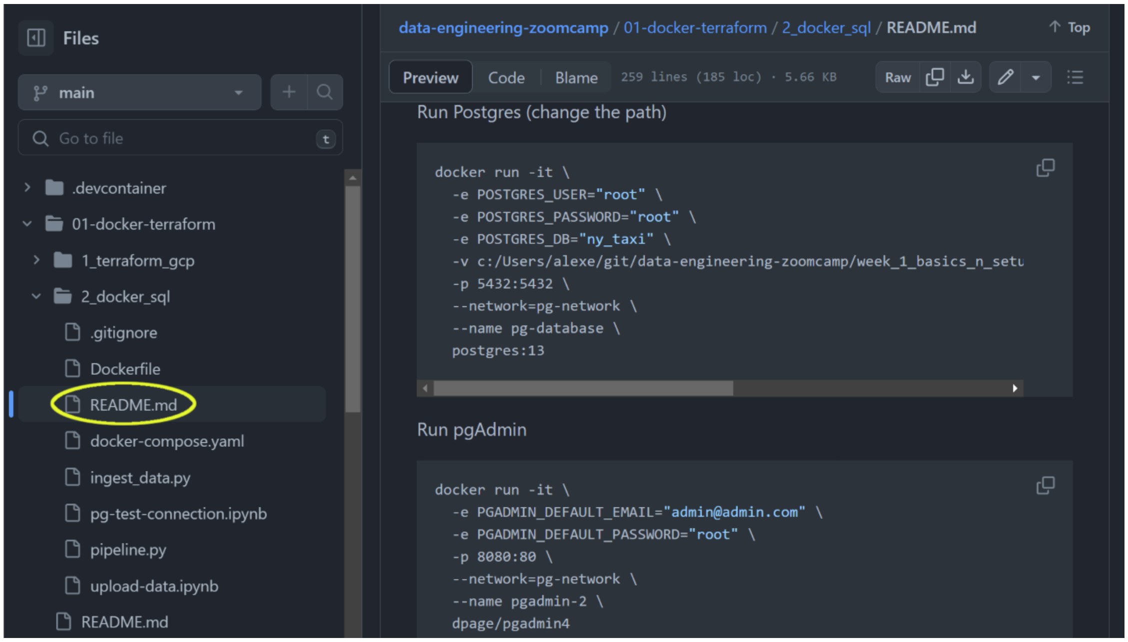
Postgres

Postgres

docker_sql notes

I'm following the video playlist order so, we starting with docker the main takeaway of the docker module is to fully appreciate the difference of Dockerfile and the docker-compose.yaml file pull up each file in vs code show the code to `docker run` and `docker compose up`

1.7 Resource: docker chapter



- .devcontainer
- 01-docker-terraform
  - 1_terraform_gcp
  - 2_docker_sql
    - .gitignore
    - Dockerfile
    - README.md
    - docker-compose.yaml
    - ingest_data.py
    - pg-test-connection.ipynb
    - pipeline.py
    - upload-data.ipynb
  - README.md

data-engineering-zoomcamp / 01-docker-terraform / 2_docker_sql / README.md

259 lines (185 loc) · 5.66 KB

Run Postgres (change the path)

```
docker run -it \
  -e POSTGRES_USER="root" \
  -e POSTGRES_PASSWORD="root" \
  -e POSTGRES_DB="ny_taxi" \
  -v c:/Users/alexe/git/data-engineering-zoomcamp/week_1_basics_n_setu
  -p 5432:5432 \
  --network=pg-network \
  --name pg-database \
  postgres:13
```

Run pgAdmin

```
docker run -it \
  -e PGADMIN_DEFAULT_EMAIL="admin@admin.com" \
  -e PGADMIN_DEFAULT_PASSWORD="root" \
  -p 8080:80 \
  --network=pg-network \
  --name pgadmin-2 \
  dpage/pgadmin4
```

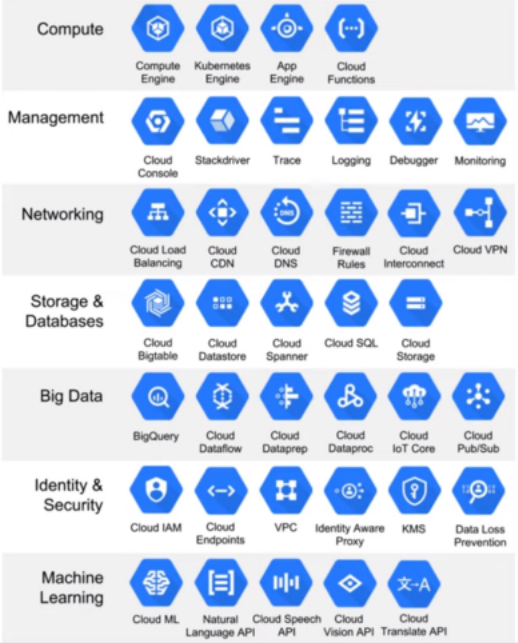
1.8 Goals of Module-01

- gcp_terraform
  - setup GCP account and service account
  - setup terraform IaC, with and without variables
  - learn the cycle of init-plan-apply-destroy

1.9 Concepts: google cloud platform

- Compute, Storage (bucket), BigQuery

1.10 Concepts: gcp diagram



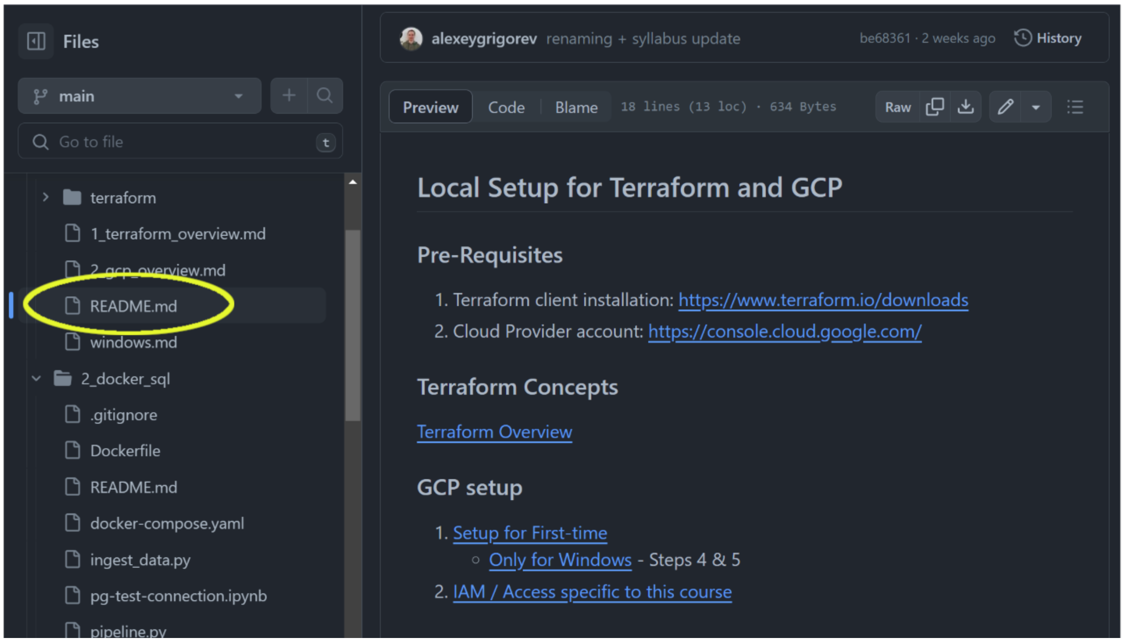
1.11 Concepts: infrastructure

- what is IaC
- what is terraform
- the life-cycle of an infrastructure
  - init, plan, apply, destroy

gcp_terraform notes

using google cloud platform & terraform (hashicorp); also offered by Azure and Amazon there's an equivalent offering from Azure and Amazon, though I can never remember what service is whose emphasis on IAM service accounts and the right permissions for the right tasks not really that familiar, so I'll let the experts teach you this one

1.12 Resource: terraform chapter



Local Setup for Terraform and GCP

Pre-Requisites

1. Terraform client installation: https://www.terraform.io/downloads
2. Cloud Provider account: https://console.cloud.google.com/

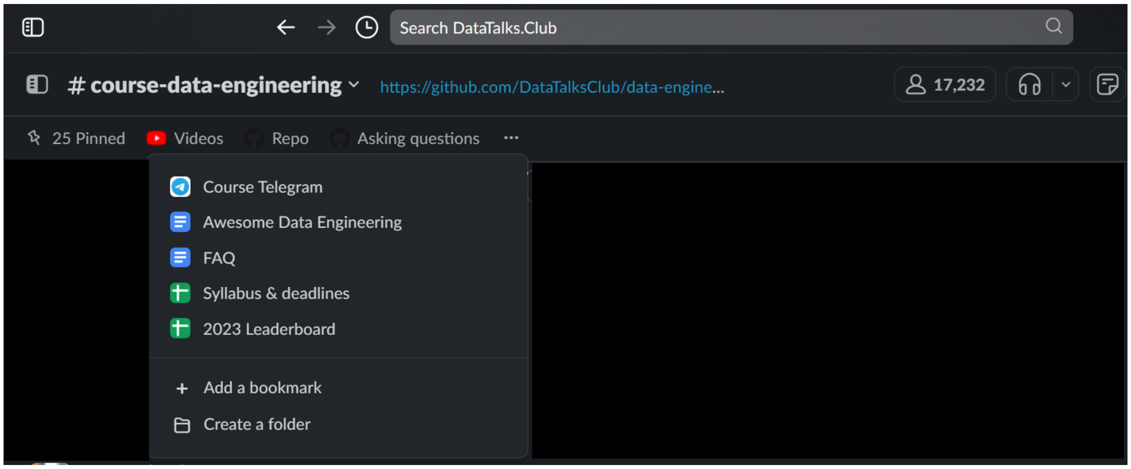
Terraform Concepts

Terraform Overview

GCP setup

1. Setup for First-time
   - Only for Windows - Steps 4 & 5
2. IAM / Access specific to this course

1.13 Locating resources

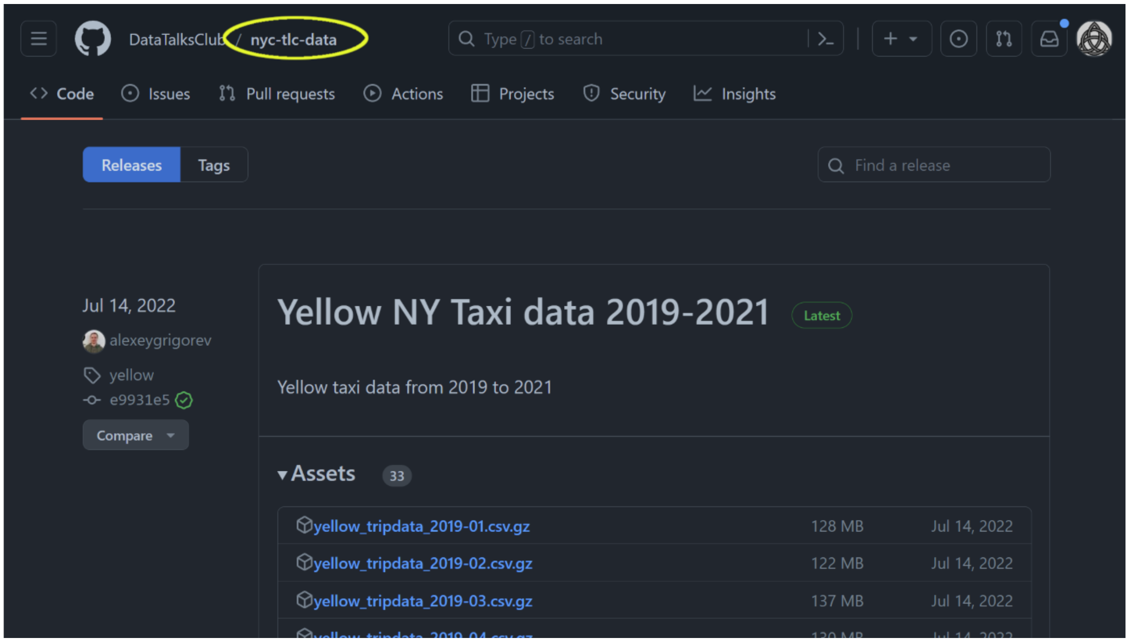


resources notes

- slack: communication
  - library of tools at fingertips: ctrl+f is your friend!
  - show slack's search features
  - pins
  - bookmarks

```
[ ]: - slack
       - FAQs (pull it up), ctrl+f is your friend!
       - asking-questions.md (pull it up, also in FAQ)
```

1.14 Resource: ny_taxi dataset



Jul 14, 2022
alexeygrigorev
yellow
e9931e5

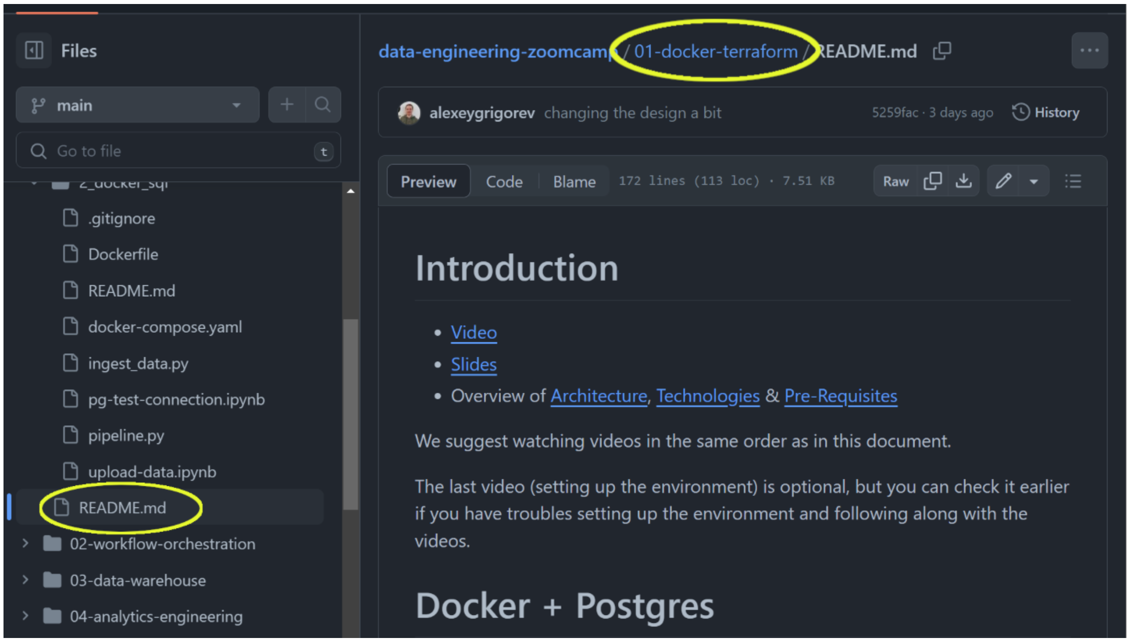
Yellow NY Taxi data 2019-2021

Yellow taxi data from 2019 to 2021

▾Assets 33

yellow_tripdata_2019-01.csv.gz	128 MB	Jul 14, 2022
yellow_tripdata_2019-02.csv.gz	122 MB	Jul 14, 2022
yellow_tripdata_2019-03.csv.gz	137 MB	Jul 14, 2022

1.15 Resource: Module-01



alexeygrigorev changing the design a bit 5259fac · 3 days ago

172 lines (113 loc) · 7.51 KB

Introduction

- Video
- Slides
- Overview of Architecture, Technologies & Pre-Requisites

We suggest watching videos in the same order as in this document.

The last video (setting up the environment) is optional, but you can check it earlier if you have troubles setting up the environment and following along with the videos.

Docker + Postgres

resources notes

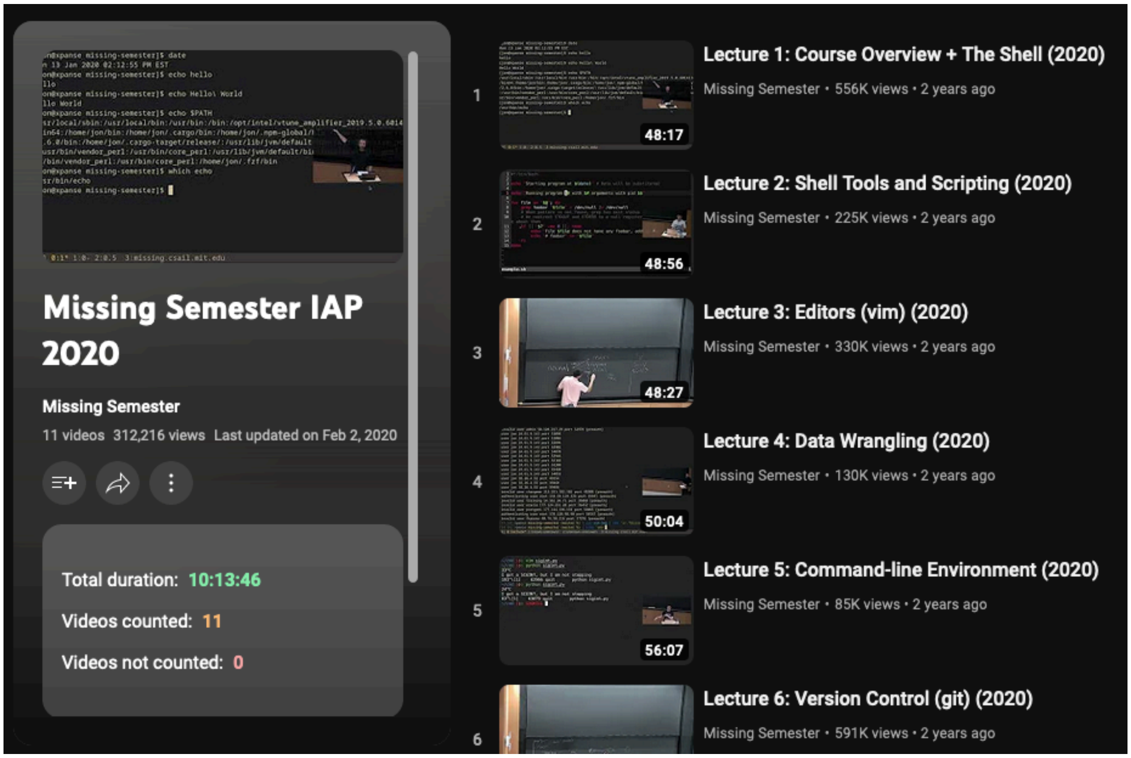
all resources that you could ever need is already posted and made available - repository: source of truth. always sync before push our own work - each module has it's own README and each submodule too; some may have additional notes - also notes from past cohorts and office-hours: watch them before asking QNs

1.16 Pre-requisite knowledge: CLI

Assumed to already know

- how to use command line (PATHs, ls, / vs , shells)
- The Missing Semester of Your CS Education
- youtube playlist
- Scott Hanselman's My Top Tips for using Windows Terminal like a Pro

1.17 Resource: missing semester of CS



```
on@xpanse missing-semester]$ date
n 13 Jan 2020 02:12:55 PM EST
on@xpanse missing-semester]$ echo hello
llo
on@xpanse missing-semester]$ echo Hello\ World
llo World
on@xpanse missing-semester]$ echo $PATH
sr/local/sbin:/usr/local/bin:/usr/bin:/bin:/opt/intel/vtune_amplifier_2019.5.0.6014
in64:/home/jon/bin:/home/jon/.cargo/bin:/home/jon/.npm-global/
.6.0/bin:/home/jon/.cargo-target/release/:/usr/lib/jvm/default
usr/bin/vendor_perl:/usr/bin/core_perl:/usr/lib/jvm/default/bi
/bin/vendor_perl:/usr/bin/core_perl:/home/jon/.fzf/bin
on@xpanse missing-semester]$ which echo
sr/bin/echo
on@xpanse missing-semester]$
 0:1* 1:0- 2:0.5  3:missing.csail.mit.edu
```

Missing Semester IAP 2020

Missing Semester

11 videos 312,216 views Last updated on Feb 2, 2020

Total duration: 10:13:46

Videos counted: 11

Videos not counted: 0

1 48:17 Lecture 1: Course Overview + The Shell (2020)
Missing Semester • 556K views • 2 years ago

2 48:56 Lecture 2: Shell Tools and Scripting (2020)
Missing Semester • 225K views • 2 years ago

3 48:27 Lecture 3: Editors (vim) (2020)
Missing Semester • 330K views • 2 years ago

4 50:04 Lecture 4: Data Wrangling (2020)
Missing Semester • 130K views • 2 years ago

5 56:07 Lecture 5: Command-line Environment (2020)
Missing Semester • 85K views • 2 years ago

6 Lecture 6: Version Control (git) (2020)
Missing Semester • 591K views • 2 years ago

1.18 Pre-requisite knowledge: git

Assumed to already know

- how to use git and github; git clone / push / fetch / pull, remotes, branches,

git notes

need to submit a git commit - a github page in homework submission form (pull up form) show what to put where

1.19 Pre-requisite knowledge: venv

Assumed to already know

- how to use conda, venv, pipenv, poetry, aka environment managers etc

env notes

partition our workspace

1.20 Pre-requisite knowledge: markdown

- how to write and read Markdown

course notes are all written in markdown so are slack messages - markdown formatting

need to write companion notes on our code to help us remember a month later on *why* we wrote what we wrote because a few weeks later, after we are more "learned" and "wise", that code look overly complicated or dumb, right? or might even be totally wrong, because we interpreted the question incorrectly

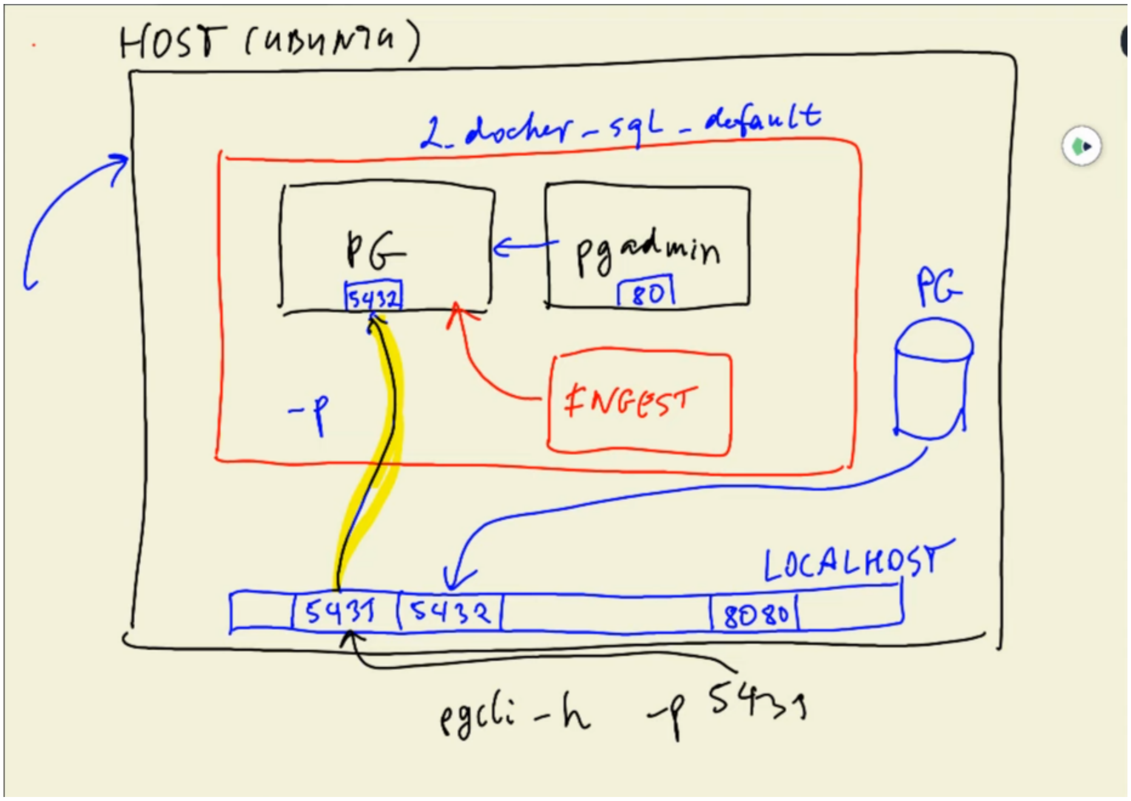
1.21 Pre-requisite knowledge: Python

- how to write and read code in Python

this is a given, no explanation necessary

1.22 Pre-requisite knowledge: networking

- concept of server and client
- concept of host and service
- watch 1.4.2 - Port Mapping and Networks in Docker (Bonus)



notes on dockers and containers

don't just keep spinning up containers when things fail.

ask why did it fail? - check for typos, if you didn't copy the command block - is the port already used? are the resources being shared like network, volumes? - run out of space for images? - for every failed command, take a beat and delete images and containers and volumes etc

1.23 Pre-requisite knowledge: docs

- how to read documentation
    - Check DATES of posts, libraries, docs last updated etc

reading docs

- when reading docs or blog posts, pay special attention to the version. is it the same as what you had installed?
- is the date the post is written current? or is it for an outdated package library?

1.24 Pre-requisite knowledge: asking questions

1.24.1 recommended skills to pick up

- GPT prompt-engineering : learn it!

asking questions notes

asking questions is a skill. as demonstrated by prompt engineering - a new field that sprung up due to ChatGPT bad questions -> bad results; just like garbage in garbage out give examples so they know what expected outcome is just like when asking for help - say what you did, say what is the expected and what error occured instead of the expected, I can tell the amount of effort you put in by the questions you asked. I've been on gaming tech support since Sims 2 days. - say what you've tried!!! IMPT!!! - mention your setup : everyone has different systems, might not even have same python versions on same OSes by using venv.

how is it fair that you expect people to help you and spent the time and effort to understand your issue, when you don't to put in the effort to provide information?

1.25 Tools to install: env setup

videos: - 1.4.1 - Setting up the Environment on Google Cloud - 1.4.2 - Using Github Codespaces for the Course -

env setup videos

the gcp vm is similar for wsl, linux and macOS if using docker, it is fine to use (base) but if on local, cannot use (base) - not recommended, not best practice. will bite you in the behind in weeks to come, might not be right NOW but we don't want it to happen when we doing capstones (personal experience), do we?

anaconda shell vs gitbash

1.26 Tools to install: vscode

- vs code
    - useful extensions
- docker desktop

1.27 Tools to install: wsl

- wsl2 + ubuntu
  - move_docker_distro_location
  - move_wsl_distro

1.28 Tools to install: auth

- gh auth or ssh keys

1.29 Tools to install: terminal

- terminal from msft store
- hanselman's blog on console, terminal, shell

1.30 Tools to install

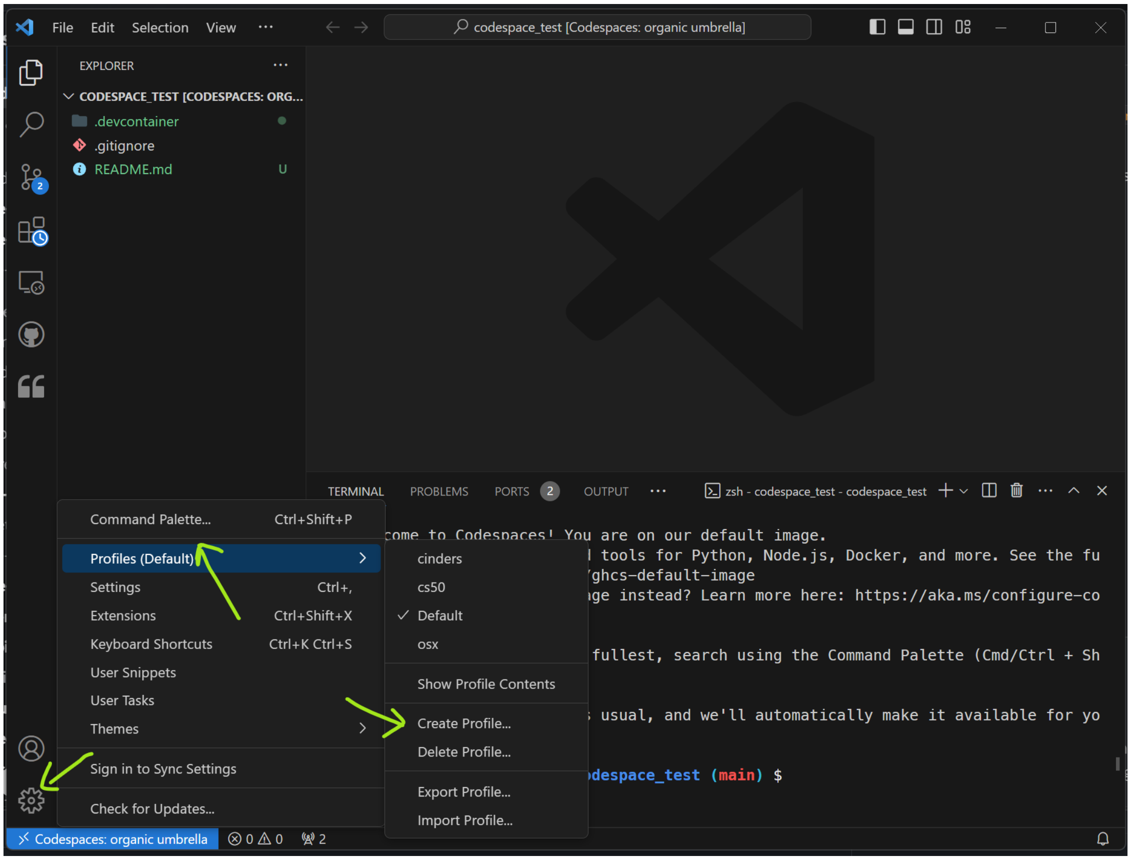
- gcp sdk: see terraform videos
- container VMs vs local dev box
  - micromamba env manager

1.31 Demo: environment setup

Demo.

Demo notes.

- docker desktop, enable wsl ubuntu box
- vs code extensions
- vs code profiles
- colorize prompts, zsh, oh-my-posh
- try not to mix conda and pip, stick to one. but conda has the "solving environment" hang issue, was on slow machine at that time (PCsaurus)
- why I switch to micromamba because of the linking, doesnt seem to matter if I use either method



```
come to Codespaces! You are on our default image.
l tools for Python, Node.js, Docker, and more. See the fu
/ghcs-default-image
ge instead? Learn more here: https://aka.ms/configure-co

 fullest, search using the Command Palette (Cmd/Ctrl + Sh

 usual, and we'll automatically make it available for yo

odespace_test (main) $
```

vs code extension and other tips

- outline, timeline
- create symbolic link to your documents path on windows `ln -s /mnt/c/Users/ellabelle/Github ~/projects`

1.32 Time for a break

know when to take a break! if everything we do results in an error or different outcome, might be time to go let the brain rest and do “difuse mode” things. because have been on “focus mode” for too long battery has run down - go recharge. preferably with a physical activity instead of scrolling our devices.

1.33 THANK YOU!

download slides here

and with that I thank you for spending time with me and giving me your attention.

if this helps, like it, subscribe, comment, star my repos. engage in slack even if you’re not having problems. help others...go forth and learn!

thank you!

test