Brushups:

**steps to commit:**

1 navigate to the project directory.

(git pull)

2. git stash

3. git stash pop

4. git add . -p

5. git commit -m "message"

6. git push

in a **robots.txt** file we can configure if bots can crawl or not to our site. if they can crawl, to what extent they can crawl can also be specified in this file. separate rules for different browser-crawlers can be used.

**Cf:**

cf login -a https://api.snpaas.eu --sso google

After #1, Get token from browser as suggested in #1. Use google login here with your springernature email

select space "space name"

After login -> cf apps -> to see list of apps on that space

cf env sm-webapp-qa-perf -> see list of env vars

cf set-env sm-webapp-qa-perf SM\_API\_SERVICES http://sm-api-qa-perf-gcp.springernature.app -> example to set an env var (cf set-env <appname> <var\_key> <var\_value>)

cf restage <app name>

cf login -a https://api.dev.cf.springer-sbm.com -u Materials\_Pune@Crest.in -p 81278c0a-4b02-11e5-885d-feff819cdc9f -o materials -s qa

**Useful CF notes and commands | Deploying an app on CF**

cf api

cf login -a <apiendpoint>

<create new play app in local, init git in it,>

<move into sbt console>

dist -> deployable package is rendered.

cf push <app name> -p <path of jar>

cf logs <app name> --recent

{generate a secret: application.conf

play.http.secret.key="changethissosomethingsecret"

$ sbt playGenerateSecret | grep -i 'generated'}

allow host in application.conf.

cf restart <app name>

cf delete <app-name>

cf create-service

cf scale <app name> <-m for ram> <-i for instances> <-k for disk>

**creating a file appender for logs:**

<appender name="FileAppender" class="ch.qos.logback.core.FileAppender">

<file> testlogs.txt </file>

<append>true</append>

<encoder>

<pattern>%msg%n</pattern>

</encoder>

</appender>

**Docker compose up and then run app and also get the docker console by:**

docker-compose run app bash

docker ps -a

docker exec -it container\_name command ( if needed )

**Get images:**

docker images

**delete any image:**

docker rmi <<image id>>

docker-compose down

**List all docker containers:**

docker ps -aq

**Stop all docker containers:**

docker stop $(docker ps -aq)

**Remove all docker containers:**

docker rm $(docker ps -aq)

**FP training git repo:**

<https://github.com/misguided-coder/scala-dec-springernature>

**SEO blog:**

<https://ahrefs.com/blog/seo-friendly-urls/>

**guide to free google services:**

<https://www.appsadmins.com/blog/complete-guide-to-free-google-services>

**SEO tools:**

<https://seosly.com/seo-tools/>