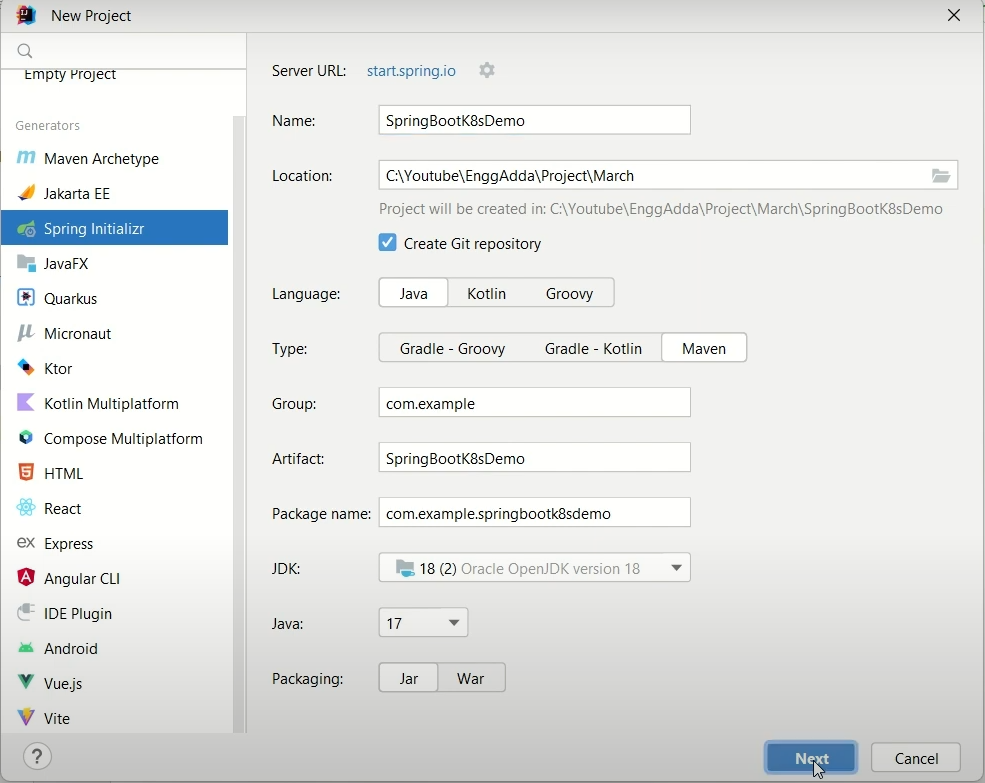
https://www.youtube.com/watch?v=zudtloZ8c9w&ab\_channel=EnggAdda

Это дока по развёртыванию простого spring boot приложение в кубере при помощи докера и миникуба

1. Устанавливаем minikube и kubectl

2. Устанавливаем Docker

3. Создаём spring-boot приложение 

Далее добавляем спринг веб

Название для примере далее использоваться не будет

4. В корне проекта создаём файл Dockerfile(Следующее содержимое файла для примера с другими названиями)

FROM openjdk:17  
EXPOSE 9090  
ADD target/docker-demo-app.jar docker-demo-app.jar  
ENTRYPOINT ["java", "-jar", "/docker-demo-app.jar"]

5. Проверяем состояние миникуба. В командной строке прописываем:

minikube version (Для проверки версии)

minikube status (Должен быть не запущен)

6. Запускаем миникуб на докере

minikube start --driver=docker

В докере у нас создаётся образ и контейнер с миникубом

7. Проверяем что миникуб запущен:

minikube status

8. Прописываем команду чтобы внутри миникуба работал докер



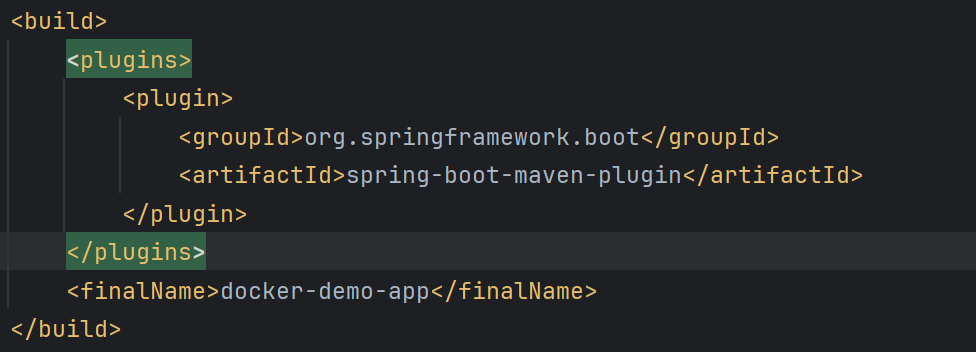
9. Прописываем команду которую нам предлагает командная строка(она может выглядеть по другому(например на маке))

10. Теперь можем выполнить команду “docker images” и увидим что она выполняется внутри миникуба а не локально. Можно выполнить ее же в пустом терминале и сравнить

11. Создаём образ нашего приложения в образе миникуба(кластера кубера)

Переходим в папку проукта (если открыли терминал не в папке проекта)

Создаём образ “docker build -t <имя образа>:<версия> . ”(ТОЧКА В КОНЦЕ ОБЯЗАТЕЛЬНА) Перед стартом команды убеждаемся что в проекте создан jar файл. Также можем в помнике указать тэг <finalName>ИМЯ\_ОБРАЗА</finalName>



12. Проверяем командой ”docker images” что образ создан

13. Разворачиваем образ докер в работающем кластере кубера

“kubectl create deployment <имя деплойта> --image=<копируем имя репозитория>:<копируем версию репозитория> --port=<номер порта>”

14. Проверяем деплоймент

“kubectl get deployment” также смотрим на статус готовности

15. Можем посмотреть описание деплойта

“kubectl describe deployment <название деплоймента>”

Там много чего интересного

16. Смотрим порт на котором работает под “kubectl get pods”

Зная его мы можем посмотреть логи

“kubectl logs <название пода>”

17. Развертываем деплоймент(но мы также можем развернуть не деплоймент а например под)

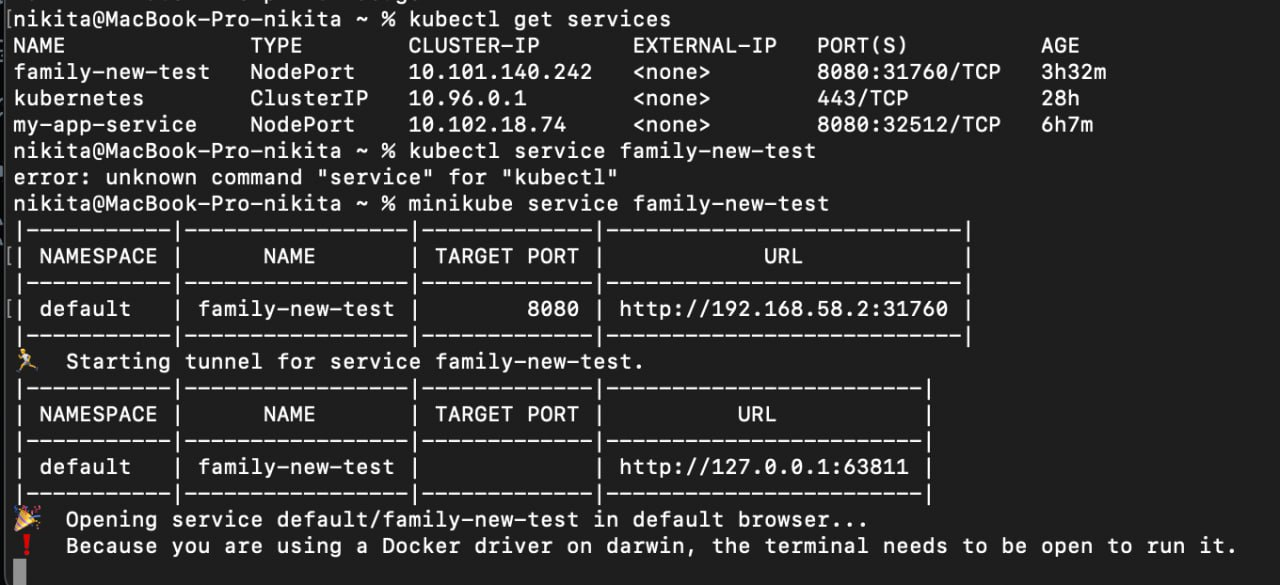
“kubectl expose deployment <имя деплоймента> --type=NodePort”(хз что значит тип)

18. Получаем доступ к сервису

“kubectl get services” видим список сервисрв

“minikube service <имя сервиса> --url”

Мы получаем url и вставляем его в браузер(на маке браузер открывается сам) не уверен что если он просто напишет url то оно откроется так как создаётся мост и в итоге он может помняться



Пример с мака

19. Далее в браузере мы можем переходить по страничкам приложения

Напрмер <http://127.0.0.1:6381/>......

Готово теперь наше приложение работает внутри кластера миникуба(: