# **Kubernetes Notes:**

# 🂣 أول إشي: شو هو الـ Deployment؟

🗸 تعریف میسط:

Deployment هو مثل "مدير ذكي" مسؤول عن تشغيل تطبيقك داخل Kubernetes، ويحرص على إنه:

- عدد النسخ (Pods) اللي طلبتيها شغالة.
- النسخة الصحيحة من التطبيق مستخدمة.
- وإذا وقع واحد من الـ Pods، يعيده تلقائيًا.
- وإذا غيرتي شي (زي الصورة أو العدد)، يعمل التحديث بطريقة ذكية وبدون توقف.

#### 🧠 طیب خلینا نرجع خطوة لورا:

## كيف بيشتغل Kubernetes عادةً؟

- أنت بدك تشغلي تطبيق ويب مثلاً.
- فبتنشئي Pod يحوي الحاوية (container) اللي فيها تطبيقك.
  - لكن:
  - Pod واحد ممكن يوقع.
  - ما فیه تکرار (replication).
  - التحدیثات لازم تعملیها یدوي.

وهنا بييجي الـ Deployment علشان يحل كل هالمشاكل.



## Web App with Mongo Database

تشغيل تطبيق ويب (Web App) بيستخدم قاعدة بيانات MongoDB، وكل الإعدادات رح تنعمل من **العقدة الرئيسية** (master node) فقط.

المطلوبة لتشغيل التطبيق Kubernetes هذا أمر لتنزيل مشروع جاهز فيه ملفات الـ

git clone https://github.com/lbrahimTalaatAh/kubernetes.git

# شو اللي بيصير هون؟

هو أداة لإدارة الأكواد والمشاريع : Git

- clone انسخ نسخة كاملة من مشروع موجود على الإنترنت (GitHub)".
- ملفات السمه بتنزل مجلد اسمه YAML.

#### : لملفات اللي بتنزل

- mongo-config.yaml
- mongo-secret.yaml
- mongo.yaml
- webapp.yaml

#### cd kubernetes/

### kubectl apply -f mongo-config.yaml

# kubectl apply -f mongo-config.yaml .4 MongoDB (اللي اسمها (kubectl ) علشان يضيف إعدادات معينة تتعلق بـ Kubernetes (اللي اسمها mongo-config.yaml) طيب، شو بيحتوي الملف mongo-config.yaml ? دملف بيخزن معلومات عادية (زي اسم الداتابيس، اسم المستخدم...). هاي المعلومات مش سرّية.

## kubectl apply -f mongo-secret.yaml

```
kubectl apply -f mongo-secret.yaml .5 •

شون بنعمل نفس الشي، بس مع ملف أسرار (Secrets).

شو يعني Secrets؟

• معلومات حساسة مثل كلمات المرور وبيانات الدخول.
```

### kubectl apply -f mongo.yaml

## kubectl apply -f mongo.yaml .6 •

🏟 هذا الملف هو المسؤول عن ت**شغيل قاعدة بيانات MongoDB نفسها** على Kubernetes.

شو بيحتوي هذا الملف؟

- تعریف Deployment لتشغیل MongoDB.
- بيربطه مع الـ ConfigMap و الـ Secret اللي عملناهم قبل شوي.
- ممكن يحتوي أيضًا على Service علشان نقدر نتواصل مع Mongo.

### kubectl apply -f webapp.yaml

## kubectl apply -f webapp.yaml .7 •

📱 هذا الملف بيشغل تطبيق الويب نفسه.

شو فیه؟

- Deployment لتطبيق الويب (مثلاً باستخدام Node.js أو Python أو أي لغة).
- Service من نوع NodePort تفتح بورت (30100) علشان نقدر نفتح التطبيق من المتصفح.

"Open the webapp website from your host machine using any node IP with port 30100."

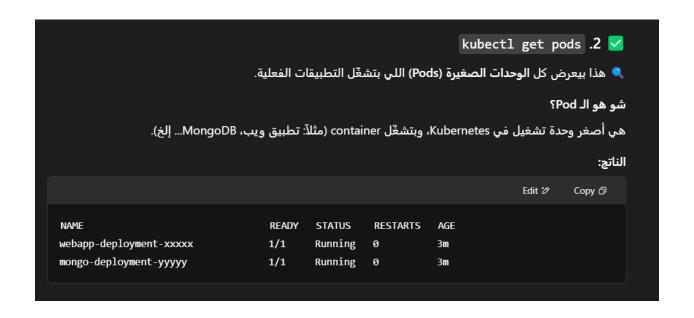
http://<IP\_of\_any\_node>:30100  $\rightarrow$  http://192.168.1.110:30100

# Managing your deployment:

## kubectl get deployments



## kubectl get pods



#### kubectl get services



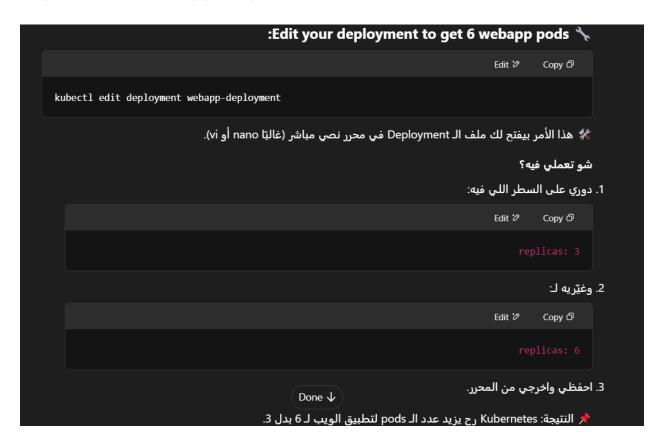
### kubectl describe <type> <type name>

<type> could be deployment, service, pod, ..
هذا بيعطيك تفاصيل شاملة عن أى كائن بالكلاستر.

## :الأمثلة

- kubectl describe deployment webapp-deployment
- kubectl describe pod mongo-deployment-xxxx
- kubectl describe service webapp-service

Edit your deployment to get 6 webapp pods, use the kubectl edit deployment webapp-deployment command for that.



Check how many pods are running in the cluster

#### kubectl get pods

#### Remove all deployments and services

kubectl delete deployment webapp-deployment kubectl delete deployment mongo-deployment

#### :services لحذف الـ

kubectl delete service webapp-service kubectl delete service mongo-service

Open mongo.yaml and change the replicas to 4 and do the same for webapp.yaml setting the replicas to 10 and redeploy mongo then webapp.

#### nano mongo.yaml

Find the line: replicas: 1 and change it to: replicas: 4

Same thing for webapp.yaml  $\rightarrow$  Open it, change it to replicas: 10

#### Then restart them:

kubectl apply -f mongo.yaml kubectl apply -f webapp.yaml

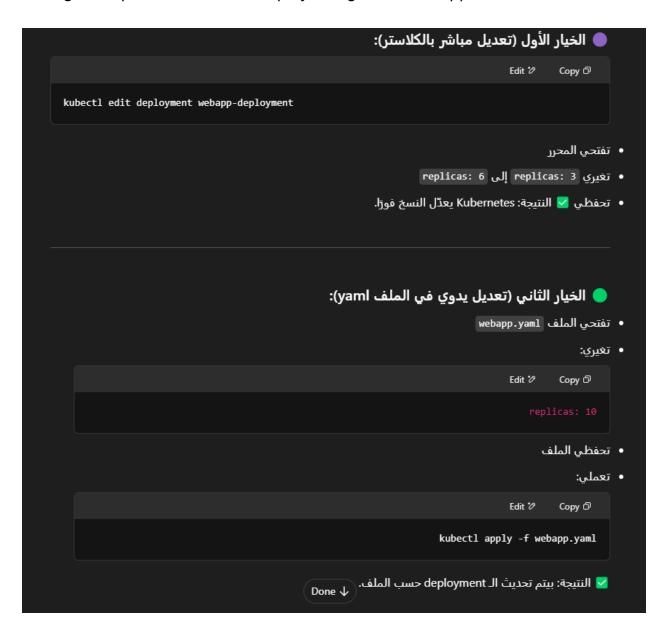
Get running pods with the -o wide option — → -o wide

ex: kubectl get pods -o wide

What is the difference between kubectl edit deployment webapp-deployment

#### and

Open mongo.yaml and change the replicas to 4 and do the same for webapp.yaml setting the replicas to 10 and redeploy mongo then webapp.



#### Another way to edit your deployment is using the following commands:

kubectl scale deployment/webapp-deployment --replicas=5



kubectl scale deployment/mongo-deployment --replicas=2

show running pods after that:

kubectl get pods

kubectl get rs



#### Update Mongo to version 5.0.14 from 5.0

kubectl set image deployment/mongo-deployment mongodb=mongo:5.0.14



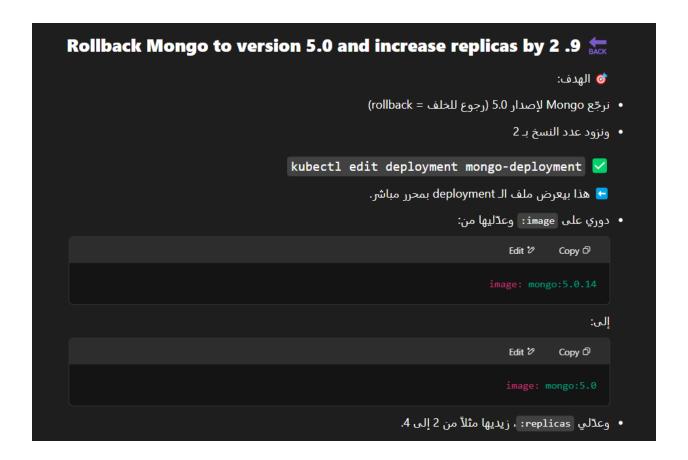
#### Check the status using the commands:

kubectl rollout status deployment/mongo-deployment





kubectl describe deployments notice the mongo version in the output. You can also rollback and return to a previous version of the container image. Let's rollback mongo to version 5.0 by editing the deployment using the command kubectl edit deployment mongo-deployment and at the same time increasing the number of pods by 2. run kubectl get pods and kubectl get rs to see things in action (be fast).



To check the history of rollouts use the command:

kubectl rollout history deployment/mongo-deployment

.deployment هذا الأمر بيعرض **تاريخ كل التحديثات** اللي عملتيها على هذا الـ

you can check the details of a specific revision using the command:

kubectl rollout history deployment/mongo-deployment --revision=2