Scale, Rollout and Rollback

**Scaling a deployment**

To scale a deployment in Kubernetes:

1. kubectl scale deployment/vote --replicas=12
3. kubectl rollout status deployment/vote

Sample output:

1. Waiting for rollout to finish: 5 of 12 updated replicas are available...
2. Waiting for rollout to finish: 6 of 12 updated replicas are available...
3. deployment "vote" successfully rolled out

**You could also update the deployment by editing it.**

1. kubectl edit deploy/vote

**[change replicas to 15 from the editor, save and observe]**

**Rolling Updates in Action**

Now, update the deployment spec to apply

file: vote-deploy.yaml

1. spec:
2. ...
3. replicas: 15
4. ...
5. labels:
6. app: python
7. role: vote
8. version: v3
9. ...
10. template:
11. spec:
12. containers:
13. - name: app
14. image: schoolofdevops/vote:v3

apply

1. kubectl apply -f vote-deploy.yaml
3. kubectl rollout status deployment/vote

Observe rollout status and monitoring screen.

1. kubectl rollout history deploy/vote
3. kubectl rollout history deploy/vote --revision=1

**Undo and Rollback**

file: vote-deploy.yaml

1. spec:
2. containers:
3. - name: app
4. image: schoolofdevops/vote:rgjerdf

apply

1. kubectl apply -f vote-deploy.yaml
3. kubectl rollout status
5. kubectl rollout history deploy/vote
7. kubectl rollout history deploy/vote --revision=xx

where replace xxx with revisions

Find out the previous revision with sane configs.

To undo to a sane version (for example revision 3)

1. kubectl rollout undo deploy/vote --to-revision=3