Continuous Integration and Continuous Deployment

Image optimizing

```
PS D:\kubernetes> $env:DOCKER_BUILDKIT=0
PS D:\kubernetes> docker build -t simple ./labs/docker/simple/
DEPRECATED: The legacy builder is deprecated and will be removed in a future release.
            BuildKit is currently disabled; enable it by removing the DOCKER_BUILDKIT=0
            environment-variable.

Sending build context to Docker daemon  2.048kB
Step 1/10 : FROM alpine:3.13 AS base
3.13: Pulling from library/alpine
Digest: sha256:469b6e04ee185740477efa44ed5bdd64a07bbdd6c7e5f5d169e540889597b911
Status: Downloaded newer image for alpine:3.13
 ---> 469b6e04ee18
Step 2/10 : RUN echo 'Adding deps...' > /deps.txt
 ---> Running in 0a98b1954326
 ---> Removed intermediate container 0a98b1954326
 ---> b589281783be
Step 3/10 : FROM base AS build
 ---> b589281783be
Step 4/10 : RUN echo 'Building...' > /build.txt
 ---> Running in e6e02ace2c87
 ---> Removed intermediate container e6e02ace2c87
 ---> 15be25fa9756
Step 5/10 : FROM base AS test
```

```
PS D:\kubernetes> docker run simple
Adding deps...
Building...
PS D:\kubernetes> _
```

```
PS D:\kubernetes> $env:DOCKER_BUILDKIT=1
PS D:\kubernetes> docker build -t simple:buildkit ./labs/docker/simple/
[+] Building 4.3s (9/9) FINISHED                                                docker:desktop-linux
 => [internal] load build definition from Dockerfile                                          0.1s
 => => transferring dockerfile: 445B                                                          0.0s
 => [internal] load metadata for docker.io/library/alpine:3.13                                0.3s
 => [internal] load .dockerignore                                                             0.0s
 => => transferring context: 2B                                                               0.0s
 => [base 1/2] FROM docker.io/library/alpine:3.13@sha256:469b6e04ee185740477efa44ed5bdd64a07  3.7s
 => => resolve docker.io/library/alpine:3.13@sha256:469b6e04ee185740477efa44ed5bdd64a07bbdd6c  3.7s
 => [auth] library/alpine:pull token for registry-1.docker.io                                 0.0s
 => CACHED [base 2/2] RUN echo 'Adding deps...' > /deps.txt                                   0.0s
 => CACHED [build 1/1] RUN echo 'Building...' > /build.txt                                    0.0s
 => CACHED [stage-3 1/1] COPY --from=build /build.txt /build.txt                              0.0s
 => exporting to image                                                                        0.1s
 => => exporting layers                                                                       0.0s
 => => exporting manifest sha256:f2be4dad75298928aa6077627a1ead80e46edc0312a7f9cfc79251f7c239  0.0s
```

```
PS D:\kubernetes> docker run simple:buildkit
Adding deps...
Building...
PS D:\kubernetes> docker image ls simple
REPOSITORY   TAG         IMAGE ID       CREATED          SIZE
simple       latest      09dbda24e360   3 minutes ago    9.15MB
simple       buildkit    2b5220b87ff5   39 hours ago     9.12MB
PS D:\kubernetes>
```

```
PS D:\kubernetes> docker build -t simple:test --target test ./labs/docker/simple/
[+] Building 1.4s (9/9) FINISHED                                                docker:desktop-linux
 => [internal] load build definition from Dockerfile                                           0.0s
 => => transferring dockerfile: 445B                                                           0.0s
 => [internal] load metadata for docker.io/library/alpine:3.13                                 0.0s
 => [internal] load .dockerignore                                                              0.0s
 => => transferring context: 2B                                                                0.0s
 => [base 1/2] FROM docker.io/library/alpine:3.13@sha256:469b6e04ee185740477efa44ed5bdd64a07bbdd6c7e5f5d169e54088  0.0s
 => => resolve docker.io/library/alpine:3.13@sha256:469b6e04ee185740477efa44ed5bdd64a07bbdd6c7e5f5d169e540889597b  0.0s
 => CACHED [base 2/2] RUN echo 'Adding deps...' > /deps.txt                                    0.0s
 => CACHED [build 1/1] RUN echo 'Building...' > /build.txt                                     0.0s
 => CACHED [test 1/2] COPY --from=build /build.txt /build.txt                                  0.0s
 => [test 2/2] RUN echo 'Testing...' >> /build.txt                                             0.8s
 => exporting to image                                                                         0.3s
 => => exporting layers                                                                        0.1s
 => => exporting manifest sha256:cbada0025f296fddef04a6c7776710fbb0572f9285ea2ca1d11db1d12878504a  0.0s
 => => exporting config sha256:56c97e4348cf012de724c5c0fec35bb71937b4898127080dd88da816bdc50b00  0.0s
 => => exporting attestation manifest sha256:14a2f2f458f5c14e0ae323b64c8cdad7680db817c73f02eaa76916d9713e5b45  0.0s
 => => exporting manifest list sha256:82f5c469ac37498e9a2c188fba8d5fc15fd3dae1318759962d9fa6434077e072  0.0s
 => => naming to docker.io/library/simple:test                                                 0.0s
 => => unpacking to docker.io/library/simple:test                                              0.0s
PS D:\kubernetes> docker run simple:test
PS D:\kubernetes> docker run simple:test cat /build.txt
Building...
Testing...
```

```
PS D:\kubernetes> docker build -t whoami ./labs/docker/whoami/
[+] Building 125.9s (6/15)                                                      docker:desktop-linux
 => [internal] load build definition from Dockerfile                                           0.0s
 => => transferring dockerfile: 562B                                                           0.0s
 => [internal] load metadata for docker.io/library/golang:1.16.4-alpine                        5.1s
 => [auth] library/golang:pull token for registry-1.docker.io                                  0.0s
 => [internal] load .dockerignore                                                              0.0s
 => => transferring context: 2B                                                                0.0s
 => [builder 1/7] FROM docker.io/library/golang:1.16.4-alpine@sha256:0dc62c5cc2d97657c17ff3bc0224214e10226e245c  111.4s
 => => resolve docker.io/library/golang:1.16.4-alpine@sha256:0dc62c5cc2d97657c17ff3bc0224214e10226e245c94317e352e  0.0s
 => => sha256:3df88182f7acff97ffde9f614a0fd86e8a26590e445aa76e442c3a79d9e4c4f4 155B / 155B      0.8s
 => => sha256:c5e7595549f7536d76f08a8a23fb67e3e6fae08ccf3add715c5c1c956f9445d2 105.75MB / 105.75MB  105.5s
 => => sha256:adcc1eea9eeabb6de296adb3e0c1b0722cf13251ff3e4e2d0a5f7ed8e3d48342 281.27kB / 281.27kB  2.3s
 => => sha256:4c4ab2625f07be8d5c6e48046a05ff3ecc7f374b794a926fb62247b66b511909 154B / 154B      1.9s
 => => extracting sha256:adcc1eea9eeabb6de296adb3e0c1b0722cf13251ff3e4e2d0a5f7ed8e3d48342      0.2s
 => => extracting sha256:4c4ab2625f07be8d5c6e48046a05ff3ecc7f374b794a926fb62247b66b511909      0.0s
 => => extracting sha256:c5e7595549f7536d76f08a8a23fb67e3e6fae08ccf3add715c5c1c956f9445d2      5.6s
 => => extracting sha256:3df88182f7acff97ffde9f614a0fd86e8a26590e445aa76e442c3a79d9e4c4f4      0.1s
 => [internal] load build context                                                              0.1s
 => => transferring context: 6.32kB                                                            0.0s
 => [builder 2/7] RUN apk --no-cache --no-progress add      ca-certificates     git      tzdata     && update-ca-  9.3s
 => => # (1/7) Installing brotli-libs (1.0.9-r3)
 => => # (2/7) Installing nghttp2-libs (1.42.0-r1)
 => => # (3/7) Installing libcurl (7.79.1-r3)
 => => # (4/7) Installing expat (2.2.10-r8)
```

```
PS D:\kubernetes> docker pull golang:1.16.4-alpine
>>
1.16.4-alpine: Pulling from library/golang
Digest: sha256:0dc62c5cc2d97657c17ff3bc0224214e10226e245c94317e352ee8a2c54368b4
Status: Downloaded newer image for golang:1.16.4-alpine
docker.io/library/golang:1.16.4-alpine
PS D:\kubernetes>
>> docker image ls -f reference=whoami -f reference=golang
REPOSITORY   TAG             IMAGE ID       CREATED          SIZE
whoami       latest          34217e97725c   36 seconds ago   14.9MB
golang       1.16.4-alpine   0dc62c5cc2d9   4 years ago      444MB
PS D:\kubernetes> _
```

```
PS D:\kubernetes> docker run -d -P --name whoami1 whoami
>>
cf6e9653de95d62d3d4adfa5f7c56ac670ed79171a057e88d572635f659112ac
PS D:\kubernetes> docker port whoami1
80/tcp -> 0.0.0.0:32768
PS D:\kubernetes> curl http://localhost:80
```

BUILDKIT

```
PS D:\kubernetes> kubectl apply -f labs/buildkit/specs/buildkitd
service/buildkitd created
deployment.apps/buildkitd created
PS D:\kubernetes> kubectl logs -l app=buildkitd
Error from server (BadRequest): container "buildkitd" in pod "buildkitd-77f5ff9797-vjhw8" is waiting to start: Container
Creating
PS D:\kubernetes> kubectl apply -f labs/buildkit/specs/sleep
pod/sleep created
```

```
PS D:\kubernetes> kubectl apply -f labs/buildkit/specs/sleep
pod/sleep created
PS D:\kubernetes> kubectl get pods
NAME                         READY   STATUS    RESTARTS      AGE
buildkitd-77f5ff9797-vjhw8   1/1     Running   1 (8m3s ago)   6h41m
sleep                        1/1     Running   0              16s
```

```
PS D:\kubernetes> kubectl exec -it sleep -- sh
/ # ^C
/ # wget https://github.com/moby/buildkit/releases/download/v0.9.0/buildkit-v0.9.0.linux-amd64.tar.gz
Connecting to github.com (20.87.245.0:443)
Connecting to objects.githubusercontent.com (185.199.108.133:443)saving to 'buildkit-v0.9.0.linux-amd64.tar.gz'
buildkit-v0.9.0.linu  82% |*********************************************************************************************  buildkit-v0.9.0.linu 100% |**************
**| 45.2M  0:00:00 ETA
'buildkit-v0.9.0.linux-amd64.tar.gz' saved
/ # tar xvf buildkit-v.0.9.0.linux-amd64.tar.gz
tar: can't open 'buildkit-v.0.9.0.linux-amd64.tar.gz': No such file or directory
/ # tar xvf buildkit-v0.9.0.linux-amd64.tar.gz
bin/
bin/buildctl
bin/buildkit-qemu-aarch64
bin/buildkit-qemu-arm
bin/buildkit-qemu-i386
bin/buildkit-qemu-mips64
bin/buildkit-qemu-mips64el
bin/buildkit-qemu-ppc64le
bin/buildkit-qemu-riscv64
bin/buildkit-qemu-s390x
bin/buildkit-runc
bin/buildkitd
```

```
/bin # ./buildctl --addr tcp://buildkitd:1234 build --frontend=dockerfile.v0 --local context=. --local dockerfile=. --output type=image,name=simple
[+] Building 10.0s (3/7)
 => [internal] load build definition from Dockerfile                                                                0.1s
 => => transferring dockerfile: 429B                                                                                0.0s
 => [internal] load .dockerignore                                                                                   0.1s
 => => transferring context: 2B                                                                                     0.0s
 => [internal] load metadata for docker.io/library/alpine:3.13                                                      5.4s
 => [base 1/2] FROM docker.io/library/alpine:3.13@sha256:469b6e04ee185740477efa44ed5bdd64a07bbdd6c7e5f5d169e540889597b911   4.4s
 => => resolve docker.io/library/alpine:3.13@sha256:469b6e04ee185740477efa44ed5bdd64a07bbdd6c7e5f5d169e540889597b911   0.0s
 => => sha256:72cfd02ff4d01b1f319eed108b53120dea0185b916d2abeb4e6121879cbf7a65 0B / 2.83MB                           4.4s
# final app image
FROM base
COPY --from=build /build.txt /build.txt
CMD cat /deps.txt && cat /build.txt/bin # ./buildctl --adr
```

```
/bin # exit
PS D:\kubernetes> $REGISTRY_SERVER='https://index.docker.io/v1/'
PS D:\kubernetes> $REGISTRY_USER=Read-Host -Prompt 'Username'
Username: Kalekye123
PS D:\kubernetes> $password = Read-Host -Prompt 'Password'-AsSecureString
Password: ************
PS D:\kubernetes> $REGISTRY_PASSWORD = [System.Net.NetworkCredential]::new("", $password).Password
PS D:\kubernetes>
```

```
PS D:\kubernetes> kubectl create secret docker-registry registry-creds --docker-server=$REGISTRY_SERVER --docker-username=$REGISTRY_USER --docker-password=$REGISTRY_PASSWOR
D
secret/registry-creds created
PS D:\kubernetes> kubectl create configmap build-config --from-literal=REGISTRY=docker.io  --from-literal=REPOSITORY=courselabs
configmap/build-config created
PS D:\kubernetes>
```

```
PS D:\kubernetes> kubectl apply -f labs/buildkit/specs/buildkit-cli
>>
pod/buildkit-cli created
PS D:\kubernetes> kubectl exec -it buildkit-cli -- sh
/ # printenv | grep RE
REPOSITORY=courselabs
REGISTRY=docker.io
/ # ls -l /root/.docker
total 0
lrwxrwxrwx    1 root     root            18 May 14 17:12 config.json -> ..data/config.json
/ # cd ~
~ # wget --no-check-certificate https://raw.githubusercontent.com/courselabs/kubernetes/main/labs/docker/simple/Dockerfile
Connecting to raw.githubusercontent.com (185.199.111.133:443)
saving to 'Dockerfile'
Dockerfile           100% |****************|   390  0:00:00 ETA
'Dockerfile' saved
```

HELM

```
PS D:\kubernetes> choco install kubernetes-helm
Chocolatey v2.3.0
Installing the following packages:
kubernetes-helm
By installing, you accept licenses for the packages.
Downloading package from source 'https://community.chocolatey.org
/api/v2/'
Progress: Downloading kubernetes-helm 3.17.3... 100%

kubernetes-helm v3.17.3 [Approved]
kubernetes-helm package files install completed. Performing other
 installation steps.
The package kubernetes-helm wants to run 'chocolateyInstall.ps1'.
```

```
PS D:\kubernetes> helm version --short
v3.17.3+ge4da497
```

```
PS D:\kubernetes> helm install whoami-default labs/helm/charts/whoami
NAME: whoami-default
LAST DEPLOYED: Wed May 14 20:26:24 2025
NAMESPACE: default
STATUS: deployed
REVISION: 1
TEST SUITE: None
PS D:\kubernetes> helm ls
NAME            NAMESPACE       REVISION        UPDATED                                 STATUS          CHART           APP VERSION
whoami-default  default         1               2025-05-14 20:26:24.6875858 +0300 EAT   deployed        whoami-0.1.0    1.0.0
PS D:\kubernetes> kubectl get all -l app.kubernetes.io/managed-by=Helm
NAME                          TYPE       CLUSTER-IP       EXTERNAL-IP   PORT(S)          AGE
service/whoami-default-server   NodePort   10.109.135.220   <none>        8020:30028/TCP   15s

NAME                                    READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/whoami-default-server   2/2     2            2           15s
PS D:\kubernetes>
```

```
PS D:\kubernetes>
                kubectl get po -o wide --show-labels
NAME                                  READY   STATUS    RESTARTS   AGE   IP          NODE             NOMINATED NODE   READINESS GATES   LABELS
whoami-default-server-56d6d8d6b-2rpb4   1/1     Running   0          46s   10.1.0.97   docker-desktop   <none>           <none>            app=whoami-default,component=serv
er,pod-template-hash=56d6d8d6b
whoami-default-server-56d6d8d6b-42ctd   1/1     Running   0          46s   10.1.0.98   docker-desktop   <none>           <none>            app=whoami-default,component=serv
er,pod-template-hash=56d6d8d6b
PS D:\kubernetes>
```

```
PS D:\kubernetes>
>> kubectl describe svc whoami-default-server
Name:                     whoami-default-server
Namespace:                default
Labels:                   app=whoami-default
                          app.kubernetes.io/managed-by=Helm
Annotations:              meta.helm.sh/release-name: whoami-default
                          meta.helm.sh/release-namespace: default
Selector:                 app=whoami-default,component=server
Type:                     NodePort
IP Family Policy:         SingleStack
IP Families:              IPv4
IP:                       10.109.135.220
IPs:                      10.109.135.220
Port:                     <unset>  8020/TCP
TargetPort:               http/TCP
NodePort:                 <unset>  30028/TCP
Endpoints:                10.1.0.98:80,10.1.0.97:80
Session Affinity:         None
External Traffic Policy:  Cluster
Internal Traffic Policy:  Cluster
Events:                   <none>
```

```
PS D:\kubernetes> curl http://localhost:30028
>>


StatusCode        : 200
StatusDescription : OK
Content           : Hostname: whoami-default-server-56d6d8d6b-2rpb4
                    IP: 10.1.0.97
                    IP: fe80::3cea:c7ff:fefe:f9f3

RawContent        : HTTP/1.1 200 OK
                    Content-Length: 92
                    Content-Type: text/plain; charset=utf-8
                    Date: Wed, 14 May 2025 17:29:01 GMT

                    Hostname: whoami-default-server-56d6d8d6b-2rpb4
                    IP: 10.1.0.97
                    IP: fe80::3cea:c7ff:f...
Forms             : {}
Headers           : {[Content-Length, 92], [Content-Type, text/plain; charset=utf-8], [Date, Wed, 14 May 2025 17:29:01 GMT]}
Images            : {}
InputFields       : {}
Links             : {}
ParsedHtml        : mshtml.HTMLDocumentClass
RawContentLength  : 92
```

```
PS D:\kubernetes> helm install whoami-custom --set replicaCount=1 --set serviceNodePort=30038 labs/helm/charts/whoami
NAME: whoami-custom
LAST DEPLOYED: Wed May 14 20:29:22 2025
NAMESPACE: default                                    STATUS: deployed                                    REVISION: 1
                    TEST SUITE: None                                    PS D:\kubernetes> helm ls
>>
NAME            NAMESPACE       REVISION        UPDATED                                 STATUS          CHART           APP VERSION
whoami-custom   default         1               2025-05-14 20:29:22.854054 +0300 EAT    deployed        whoami-0.1.0    1.0.0
whoami-default  default         1               2025-05-14 20:26:24.6875858 +0300 EAT   deployed        whoami-0.1.0    1.0.0
PS D:\kubernetes>
>> kubectl get pods -l component=server -L app
NAME                                    READY   STATUS    RESTARTS   AGE     APP
whoami-custom-server-bdc446cbd-sz4gz    1/1     Running   0          15s     whoami-custom
whoami-default-server-56d6d8d6b-2rpb4   1/1     Running   0          3m14s   whoami-default
whoami-default-server-56d6d8d6b-42ctd   1/1     Running   0          3m14s   whoami-default
```

```
PS D:\kubernetes> curl http://localhost:30038
>>


StatusCode        : 200
StatusDescription : OK
Content           : Hostname:
                    whoami-custom-server-bdc446cbd-sz4gz
                    IP: 10.1.0.99
                    IP: fe80::dc72:fdff:fe8e:754

RawContent        : HTTP/1.1 200 OK
                    Content-Length: 90
                    Content-Type: text/plain; charset=utf-8
                    Date: Wed, 14 May 2025 17:29:46 GMT

                    Hostname:
                    whoami-custom-server-bdc446cbd-sz4gz
                    IP: 10.1.0.99
                    IP: fe80::dc72:fdff:fe...
Forms             : {}
Headers           : {[Content-Length, 90], [Content-Type,
                    text/plain; charset=utf-8], [Date, Wed, 14
                    May 2025 17:29:46 GMT]}
Images            : {}
InputFields       : {}
Links             : {}
ParsedHtml        : mshtml.HTMLDocumentClass
RawContentLength  : 90
```

```
PS D:\kubernetes> helm upgrade whoami-custom --set serverMode=V labs/helm/charts/whoami
Error: UPGRADE FAILED: cannot patch "whoami-custom-server" with kind Service: Service "whoami-custom-server" is invalid: spec.ports[0].nodePort: Invhelm upgrade whoami-cust
om --reuse-values --set serverMode=V labs/helm/charts/whoami
>> D:\kubernetes>
Release "whoami-custom" has been upgraded. Happy Helming!
NAME: whoami-custom
LAST DEPLOYED: Wed May 14 20:30:49 2025
NAMESPACE: default
STATUS: deployed
REVISION: 3
TEST SUITE: None
```

```
PS D:\kubernetes> curl http://localhost:30038
>>


StatusCode        : 200
StatusDescription : OK
Content           : Hostname:
                    whoami-custom-server-55c985c587-vn76w
                    IP: 10.1.0.101
                    IP: fe80::e0d1:b5ff:feea:d649
                    RemoteAddr: 192.168.65.3:47156
                    GET / HTTP/1.1
                    Host: localhost:30038
                    User-Agent: Mozilla/5.0 (Windows NT; ...
RawContent        : HTTP/1.1 200 OK
                    Content-Length: 260
                    Content-Type: text/plain; charset=utf-8
                    Date: Wed, 14 May 2025 17:30:56 GMT

                    Hostname:
                    whoami-custom-server-55c985c587-vn76w
                    IP: 10.1.0.101
                    IP: fe80::e0d1:b5ff...
Forms             : {}
Headers           : {[Content-Length, 260], [Content-Type,
                    text/plain; charset=utf-8], [Date, Wed, 14
                    May 2025 17:30:56 GMT]}
Images            : {}
InputFields       : {}
Links             : {}
ParsedHtml        : mshtml.HTMLDocumentClass
RawContentLength  : 260
```

```
PS D:\kubernetes> kubectl get rs -l app=whoami-custom
NAME                              DESIRED   CURRENT   READY   AGE
whoami-custom-server-55c985c587   1         1         1       65s
whoami-custom-server-bdc446cbd    0         0         0       2m14s
PS D:\kubernetes>
```

```
PS D:\kubernetes>
>> helm rollback whoami-custom 1
Rollback was a success! Happy Helming!
PS D:\kubernetes> curl http://localhost:30038
>> #


StatusCode        : 200
StatusDescription : OK
Content           : Hostname:
                    whoami-custom-server-bdc446cbd-hk5qf
                    IP: 10.1.0.103
                    IP: fe80::d4c8:e2ff:fe49:4c15

RawContent        : HTTP/1.1 200 OK
                    Content-Length: 92
                    Content-Type: text/plain; charset=utf-8
                    Date: Wed, 14 May 2025 17:32:13 GMT

                    Hostname:
                    whoami-custom-server-bdc446cbd-hk5qf
                    IP: 10.1.0.103
                    IP: fe80::d4c8:e2ff:f...
Forms             : {}
Headers           : {[Content-Length, 92], [Content-Type,
                    text/plain; charset=utf-8], [Date, Wed, 14
                    May 2025 17:32:13 GMT]}
Images            : {}
InputFields       : {}
Links             : {}
ParsedHtml        : mshtml.HTMLDocumentClass
RawContentLength  : 92
```

```
PS D:\kubernetes>



                helm repo ls
Error: no repositories to show
PS D:\kubernetes>
>> helm repo add kiamol https://kiamol.net
"kiamol" has been added to your repositories
PS D:\kubernetes> helm repo update
Hang tight while we grab the latest from your chart repositories...
...Successfully got an update from the "kiamol" chart repository
Update Complete. ⎈Happy Helming!⎈
PS D:\kubernetes> helm search repo vweb --versions
NAME            CHART VERSION   APP VERSION     DESCRIPTION
kiamol/vweb     2.0.0           2.0.0           Simple versioned web app
kiamol/vweb     1.0.0           1.0.0           Simple versioned web app
PS D:\kubernetes>
```

```
PS D:\kubernetes> helm show values kiamol/vweb --version 2.0.0
# port for the Service to listen on
servicePort: 8090
# type of the Service:
serviceType: LoadBalancer
# number of replicas for the web Pod
replicaCount: 2
PS D:\kubernetes> helm install --set replicaCount=1 --set serviceType=NodePort --
set servicePort=30039 vweb kiamol/vweb --version 2.0.0
NAME: vweb
LAST DEPLOYED: Wed May 14 20:34:36 2025
NAMESPACE: default
STATUS: deployed
REVISION: 1
TEST SUITE: None
PS D:\kubernetes> kubectl get svc -l app.kubernetes.io/instance=vweb
>>
NAME   TYPE       CLUSTER-IP       EXTERNAL-IP   PORT(S)          AGE
vweb   NodePort   10.107.177.103   <none>        30039:30039/TCP   5s
PS D:\kubernetes>
```

```
PS D:\kubernetes> helm show values kiamol/vweb --version 1.0.0
servicePort: 8090
replicaCount: 2
PS D:\kubernetes>
>> helm upgrade --reuse-values vweb kiamol/vweb --version 1.0.0
Release "vweb" has been upgraded. Happy Helming!
NAME: vweb
LAST DEPLOYED: Wed May 14 20:35:19 2025
NAMESPACE: default
STATUS: deployed
REVISION: 2
TEST SUITE: None
```

ROLLOUTS

```
PS D:\kubernetes> kubectl apply -f labs/rollouts/specs/vweb
deployment.apps/vweb created
service/vweb-lb created
service/vweb-np created
```

```
PS D:\kubernetes> kubectl get svc
>>
NAME                     TYPE           CLUSTER-IP      EXTERNAL-IP   PORT(S)
      AGE
kubernetes               ClusterIP      10.96.0.1       <none>        443/TCP
      9d
products-api             ClusterIP      10.105.86.29    <none>        80/TCP
      9d
products-db              ClusterIP      None            <none>        5432/TCP
      9d
stock-api                ClusterIP      10.111.245.8    <none>        80/TCP
      9d
vweb-lb                  LoadBalancer   10.98.142.209   localhost     8090:31733
/TCP   7s
vweb-np                  NodePort       10.103.249.17   <none>        8090:30018
/TCP   7s
widgetario-web-internal  ClusterIP      10.98.196.254   <none>        80/TCP
      9d
widgetario-web-lb        LoadBalancer   10.105.99.212   localhost     8080:30226
/TCP   9d
widgetario-web-np        NodePort       10.97.32.116    <none>        8080:30008
/TCP   9d
```

```
PS D:\kubernetes> curl http://localhost:30018/v.txt


StatusCode        : 200
StatusDescription : OK
Content           : v1
RawContent        : HTTP/1.1 200 OK
                    Connection: keep-alive
                    Accept-Ranges: bytes
                    Content-Length: 2
                    Content-Type: text/plain
                    Date: Wed, 14 May 2025 17:37:36 GMT
                    ETag: "67b65329-2"
                    Last-Modified: Wed, 19 Feb 2025 21:...
Forms             : {}
Headers           : {[Connection, keep-alive], [Accept-Ranges, bytes],
                    [Content-Length, 2], [Content-Type, text/plain]...}
Images            : {}
InputFields       : {}
Links             : {}
ParsedHtml        : mshtml.HTMLDocumentClass
RawContentLength  : 2
```

```
PS D:\kubernetes> kubectl apply -f labs/rollouts/specs/vweb/update-fast
deployment.apps/vweb configured
PS D:\kubernetes> kubectl get rs -l app=vweb
NAME              DESIRED   CURRENT   READY   AGE
vweb-866df8c7fb   3         3         3       4m25s
vweb-b578f58c5    3         3         0       5s
```

```
PS D:\kubernetes> curl http://localhost:30018/v.txt


StatusCode        : 200
StatusDescription : OK
Content           : v1
RawContent        : HTTP/1.1 200 OK
                    Connection: keep-alive
                    Accept-Ranges: bytes
                    Content-Length: 2
                    Content-Type: text/plain
                    Date: Wed, 14 May 2025 17:41:43 GMT
                    ETag: "67b65329-2"
                    Last-Modified: Wed, 19 Feb 2025 21:...
Forms             : {}
Headers           : {[Connection, keep-alive], [Accept-Ranges, bytes],
                    [Content-Length, 2], [Content-Type, text/plain]...}
Images            : {}
InputFields       : {}
Links             : {}
ParsedHtml        : mshtml.HTMLDocumentClass
RawContentLength  : 2
```

```
PS D:\kubernetes> kubectl rollout history deploy/vweb
deployment.apps/vweb
REVISION  CHANGE-CAUSE
1         <none>
2         <none>
```

```
PS D:\kubernetes> kubectl rollout undo deploy/vweb
deployment.apps/vweb rolled back
PS D:\kubernetes> kubectl apply -f labs/rollouts/specs/vweb/update-slow
deployment.apps/vweb configured
PS D:\kubernetes> kubectl apply -f labs/rollouts/specs/vweb/update-broken
deployment.apps/vweb configured
```

```
PS D:\kubernetes> curl http://localhost:30018/v.txt
curl : The underlying connection was closed: The connection was closed
unexpectedly.
At line:1 char:1
+ curl http://localhost:30018/v.txt
+ ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
    + CategoryInfo          : InvalidOperation: (System.Net.HttpWebRequest:Http
   WebRequest) [Invoke-WebRequest], WebException
    + FullyQualifiedErrorId : WebCmdletWebResponseException,Microsoft.PowerShel
   l.Commands.InvokeWebRequestCommand

PS D:\kubernetes> kubectl rollout history deploy/vweb
deployment.apps/vweb
REVISION  CHANGE-CAUSE
3         <none>
4         <none>
5         <none>

PS D:\kubernetes>
>> kubectl rollout undo deploy/vweb
deployment.apps/vweb rolled back
PS D:\kubernetes> _
```

```
PS D:\kubernetes> kubectl apply -f labs/rollouts/specs/nginx-daemonset
daemonset.apps/nginx created
service/nginx-lb created
service/nginx-np created
PS D:\kubernetes> kubectl apply -f labs/rollouts/specs/nginx-daemonset/update-ond
elete
daemonset.apps/nginx configured
PS D:\kubernetes> kubectl delete po -l app=nginx
pod "nginx-ncc9b" deleted
PS D:\kubernetes> kubectl delete ds  nginx
daemonset.apps "nginx" deleted
PS D:\kubernetes> kubectl apply -f labs/rollouts/specs/nginx-statefulset
statefulset.apps/nginx created
service/nginx-statefulset created
service/nginx-lb unchanged
service/nginx-np unchanged
PS D:\kubernetes> _
```

```
PS D:\kubernetes> kubectl apply -f labs/rollouts/specs/nginx-statefulset/update-p
artition
statefulset.apps/nginx configured
```

JENKINS

```
PS D:\kubernetes> kubectl apply -f labs/jenkins/specs/infrastructure
namespace/infra created
service/buildkitd created
deployment.apps/buildkitd created
service/gogs created
deployment.apps/gogs created
deployment.apps/jenkins created
serviceaccount/jenkins created
clusterrole.rbac.authorization.k8s.io/jenkins created
clusterrolebinding.rbac.authorization.k8s.io/jenkins created
configmap/jenkins-setup created
The Service "jenkins" is invalid: spec.ports[0].nodePort: Invalid value: 30008: p
rovided port is already allocated
PS D:\kubernetes> kubectl get deploy -n infra
NAME        READY   UP-TO-DATE   AVAILABLE   AGE
buildkitd   1/1     1            1           6s
gogs        0/1     1            0           6s
jenkins     0/1     1            0           6s
```

```
PS D:\kubernetes> kubectl get po -n infra -l app=jenkins
>>
NAME                       READY   STATUS              RESTARTS   AGE
jenkins-7bf4559645-h9hzc   0/1     ContainerCreating   0          31s
PS D:\kubernetes> $REGISTRY_SERVER='https://index.docker.io/v1/'
PS D:\kubernetes> $REGISTRY_USER=Read-Host -Prompt 'Username'
Username: Kalekye123
PS D:\kubernetes> $password = Read-Host -Prompt 'Password'-AsSecureString
Password: ************
PS D:\kubernetes> $REGISTRY_PASSWORD = [System.Net.NetworkCredential]::new("", $p
assword).Password
PS D:\kubernetes>
```

```
PS D:\kubernetes> kubectl create secret docker-registry -n infra registry-creds -
-docker-server=$REGISTRY_SERVER --docker-username=$REGISTRY_USER --docker-passwor
d=$REGISTRY_PASSWORD
secret/registry-creds created
PS D:\kubernetes> kubectl rollout restart -n infra deploy/jenkins
deployment.apps/jenkins restarted
PS D:\kubernetes> kubectl get po -n infra -l app=jenkins
NAME                       READY   STATUS              RESTARTS   AGE
jenkins-7bf4559645-h9hzc   0/1     ContainerCreating   0          119s
jenkins-7f649d7587-zsjnf   0/1     ContainerCreating   0          8s
PS D:\kubernetes>
```

```
PS D:\kubernetes> kubectl create configmap -n infra build-config --from-literal=R
ELEASE_VERSION=21.09 --from-literal=REGISTRY_DOMAIN=docker.io  --from-literal=REG
ISTRY_REPOSITORY=<your-registry-id>
configmap/build-config created
PS D:\kubernetes> kubectl get deploy -n infra
NAME        READY   UP-TO-DATE   AVAILABLE   AGE
buildkitd   1/1     1            1           2m26s
gogs        1/1     1            1           2m26s
jenkins     0/1     1            0           2m26s
```

```
PS D:\kubernetes> helm upgrade --install whoami-dev --set serverImage=docker.io/c
ourselabs/whoami-lab:21.09-1 labs/jenkins/project/helm/whoami
Release "whoami-dev" does not exist. Installing it now.
NAME: whoami-dev
LAST DEPLOYED: Wed May 14 20:51:09 2025
NAMESPACE: default
STATUS: deployed
REVISION: 1
TEST SUITE: None
```

```
PS D:\kubernetes> kubectl create ns integration-test
namespace/integration-test created
PS D:\kubernetes>
>> kubectl label ns integration-test kubernetes.courselabs.co=jenkins
namespace/integration-test labeled
PS D:\kubernetes>
```