

CS571  
MANSI SHAH  
(19526)  
HOMEWORK 7

Create Nodejs app called "app.js" and Docker file

```
cs571@ubuntu:~/app_js_unhealthy$ cat app.js
const http = require('http');
const os = require('os');

console.log("Kubia server starting...");

var requestCount = 0;

var handler = function(request, response) {
  console.log("Received request from "
    + request.connection.remoteAddress);
  requestCount++;
  if (requestCount > 5) {
    response.writeHead(500);
    response.end("I'm not well. Please restart me!");
    return;
  }
  response.writeHead(200);
  response.end("You've hit " + os.hostname() + "\n");
};

var www = http.createServer(handler);
www.listen(8080);
cs571@ubuntu:~/app_js_unhealthy$ cat Dockerfile
FROM node:7
ADD app.js /app.js
ENTRYPOINT ["node", "app.js"]
```

## Create Docker image

```
cs571@ubuntu:~/app_js_unhealthy$ docker build -t kubia-unhealthy .
Sending build context to Docker daemon 3.584kB
Step 1/3 : FROM node:7
--> d9aed20b68a4
Step 2/3 : ADD app.js /app.js
--> 15ca9e29e862
Step 3/3 : ENTRYPOINT ["node", "app.js"]
--> Running in 8fb794bac645
Removing intermediate container 8fb794bac645
--> 6783c694a02e
Successfully built 6783c694a02e
Successfully tagged kubia-unhealthy:latest
cs571@ubuntu:~/app_js_unhealthy$ docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
kubia-unhealthy	latest	6783c694a02e	33 seconds ago	660MB
kubia	latest	fac6faaace7c	4 weeks ago	660MB
mansi2210/shahm888	kubia	fac6faaace7c	4 weeks ago	660MB
ubuntu	latest	ccc6e87d482b	6 weeks ago	64.2MB
hello-world	latest	fce289e99eb9	14 months ago	1.84kB
node	7	d9aed20b68a4	2 years ago	660MB

## Push the docker image to docker hub.

```
cs571@ubuntu:~/app_js_unhealthy$ docker login
Authenticating with existing credentials...
WARNING! Your password will be stored unencrypted in /home/cs571/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
cs571@ubuntu:~/app_js_unhealthy$ docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
kubia-unhealthy	latest	6783c694a02e	29 minutes ago	660MB
mansi2210/shahm888	kubia	fac6faaace7c	4 weeks ago	660MB
kubia	latest	fac6faaace7c	4 weeks ago	660MB
ubuntu	latest	ccc6e87d482b	6 weeks ago	64.2MB
hello-world	latest	fce289e99eb9	14 months ago	1.84kB
node	7	d9aed20b68a4	2 years ago	660MB

```
cs571@ubuntu:~/app_js_unhealthy$ docker tag kubia-unhealthy mansi2210/shahm888:kubia-unhealthy
cs571@ubuntu:~/app_js_unhealthy$ docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
kubia-unhealthy	latest	6783c694a02e	31 minutes ago	660MB
mansi2210/shahm888	kubia-unhealthy	6783c694a02e	31 minutes ago	660MB
kubia	latest	fac6faaace7c	4 weeks ago	660MB
mansi2210/shahm888	kubia	fac6faaace7c	4 weeks ago	660MB
ubuntu	latest	ccc6e87d482b	6 weeks ago	64.2MB
hello-world	latest	fce289e99eb9	14 months ago	1.84kB
node	7	d9aed20b68a4	2 years ago	660MB

```
cs571@ubuntu:~/app_js_unhealthy$ docker push mansi2210/shahm888:kubia-unhealthy
The push refers to repository [docker.io/mansi2210/shahm888]
af62a6a32531: Pushed
ab90d83fa34a: Layer already exists
8ee318e54723: Layer already exists
e6695624484e: Layer already exists
da59b99b3bd3b: Layer already exists
5616a6292c16: Layer already exists
f3ed6cb59ab0: Layer already exists
654f45ecb7e3: Layer already exists
2c40c66f7667: Layer already exists
kubia-unhealthy: digest: sha256:7f076c86d881e79a8a2850173775cee8f32a6364b5d86c01e597e07fe14f7f6d size: 2213
```

## Creating an HTTP-based liveness probe

```
cs571@ubuntu:~/app_js_unhealthy$ cat kuba-liveness-probe.yaml
apiVersion: v1
kind: Pod
metadata:
  name: kuba-liveness
spec:
  containers:
  - image: mansi2210/shahm888:kuba-unhealthy
    name: kuba-unhealthy
    livenessProbe:
      httpGet:
        path: /
        port: 8080
```

## Create a pod from the kuba-liveness-probe.yaml

```
cs571@ubuntu:~/app_js_unhealthy$ kubectl create -f kuba-liveness-probe.yaml
pod/kuba-liveness created
```

```
cs571@ubuntu:~/app_js_unhealthy$ kubectl get po kuba-liveness
NAME          READY   STATUS    RESTARTS   AGE
kuba-liveness 1/1     Running   0           58s
```

## Seeing a liveness probe in action

```
cs571@ubuntu:~/app_js_unhealthy$ kubectl get po kuba-liveness
NAME          READY   STATUS    RESTARTS   AGE
kuba-liveness 1/1     Running   2           5m3s
cs571@ubuntu:~/app_js_unhealthy$ kubectl logs kuba-liveness --previous
Kuba server starting...
Received request from ::ffff:172.17.0.1
Received request from ::ffff:172.17.0.1
Received request from ::ffff:172.17.0.1
Received request from ::ffff:172.17.0.1
Received request from ::ffff:172.17.0.1
Received request from ::ffff:172.17.0.1
Received request from ::ffff:172.17.0.1
Received request from ::ffff:172.17.0.1
```

Here the kuba-liveness pod restarted 2 times. We can obtain the logs of previous container by running kubectl logs "kuba-liveness" --previous.



## Description after pod restarted:

```
cs571@ubuntu:~/app_js_unhealthy$ kubectl describe po kubia-liveness
Name:          kubia-liveness
Namespace:     default
Priority:       0
Node:          minikube/192.168.39.20
Start Time:    Tue, 03 Mar 2020 00:19:20 -0800
Labels:        <none>
Annotations:   <none>
Status:        Running
IP:           172.17.0.4
IPs:
  IP: 172.17.0.4
Containers:
  kubia-unhealthy:
    Container ID:   docker://ddbf4262fe308318e5007e2679e968425e45bcfaf37d448aa14b9e8b0fe65388
    Image:          mansi2210/shahm888:kubia-unhealthy
    Image ID:       docker-pullable://mansi2210/shahm888@sha256:7f076c86d881e79a8a2850173775cee8f32a6364b5d86c01e597e07fe14f7f6d
    Port:           <none>
    Host Port:      <none>
    State:          Running
      Started:      Tue, 03 Mar 2020 00:26:47 -0800
    Last State:     Terminated
      Reason:       Error
      Exit Code:    137
      Started:      Tue, 03 Mar 2020 00:24:58 -0800
      Finished:     Tue, 03 Mar 2020 00:26:46 -0800
    Ready:          True
    Restart Count:  4
    Liveness:       http-get http://:8080/ delay=0s timeout=1s period=10s #success=1 #failure=3
    Environment:    <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from default-token-pxw49 (ro)
Conditions:
  Type            Status
  Initialized      True
  Ready            True
  ContainersReady  True
  PodScheduled     True
Volumes:
  default-token-pxw49:
    Type:          Secret (a volume populated by a Secret)
    SecretName:    default-token-pxw49
    Optional:      false
QoS Class:        BestEffort
Node-Selectors:   <none>
Tolerations:      node.kubernetes.io/not-ready:NoExecute for 300s
                  node.kubernetes.io/unreachable:NoExecute for 300s
Events:
  Type    Reason      Age    From          Message
  ----    -
  Normal  Scheduled   <unknown>    default-scheduler    Successfully assigned default/kubia-liveness to minikube
  Normal  Pulling     7m32s      kubelet, minikube    Pulling image "mansi2210/shahm888:kubia-unhealthy"
  Normal  Pulled      7m29s      kubelet, minikube    Successfully pulled image "mansi2210/shahm888:kubia-unhealthy"
  Normal  Killing     2m30s (x3 over 6m10s)  kubelet, minikube    Container kubia-unhealthy failed liveness probe, will be restarted
  Normal  Created     119s (x4 over 7m29s)   kubelet, minikube    Created container kubia-unhealthy
  Normal  Pulled      119s (x3 over 5m38s)   kubelet, minikube    Container image "mansi2210/shahm888:kubia-unhealthy" already present on machine
  Normal  Started     118s (x4 over 7m27s)   kubelet, minikube    Started container kubia-unhealthy
  Warning Unhealthy   60s (x10 over 6m30s)  kubelet, minikube    Liveness probe failed: HTTP probe failed with statuscode: 500
```

## Configuring additional properties of the liveness probe

```
Liveness:      http-get http://:8080/ delay=0s timeout=1s period=10s #success=1 #failure=3
```

### liveness probe with an initial delay

```
cs571@ubuntu:~/app_js_unhealthy$ kubectl delete po kubia-liveness
pod "kubia-liveness" deleted
cs571@ubuntu:~/app_js_unhealthy$ cat kubia-liveness-probe-initial-delay.yaml
apiVersion: v1
kind: Pod
metadata:
  name: kubia-liveness
spec:
  containers:
  - image: mansi2210/shahm888:kubia-unhealthy
    name: kubia-unhealthy
    livenessProbe:
      httpGet:
        path: /
        port: 8080
        initialDelaySeconds: 15
cs571@ubuntu:~/app_js_unhealthy$ kubectl create -f kubia-liveness-probe-initial-delay.yaml
pod/kubia-liveness created
cs571@ubuntu:~/app_js_unhealthy$ kubectl get po kubia-liveness
NAME          READY   STATUS    RESTARTS   AGE
kubia-liveness 1/1     Running   0          9s
cs571@ubuntu:~/app_js_unhealthy$
```

### Pod describe shows the delay of 15s

```
cs571@ubuntu:~/app_js_unhealthy$ kubectl describe po kubia-liveness
Name:          kubia-liveness
Namespace:     default
Priority:       0
Node:          minikube/192.168.39.20
Start Time:    Tue, 03 Mar 2020 00:34:53 -0800
Labels:        <none>
Annotations:   <none>
Status:        Running
IP:            172.17.0.4
IPs:
  IP: 172.17.0.4
Containers:
  kubia-unhealthy:
    Container ID:  docker://2acfb34e79f292f84c895b7e68fb50abe20412e9e01bcb691ab4fd7c6f85fef
    Image:         mansi2210/shahm888:kubia-unhealthy
    Image ID:      docker-pullable://mansi2210/shahm888@sha256:7f076c86d881e79a8a2850173775cee8f32a6364b5d86c01e597e07fe14f7f6d
    Port:          <none>
    Host Port:     <none>
    State:         Running
      Started:     Tue, 03 Mar 2020 00:34:57 -0800
    Ready:         True
    Restart Count: 0
    Liveness:      http-get http://:8080/ delay=15s timeout=1s period=10s #success=1 #failure=3
    Environment:   <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from default-token-pxw49 (ro)
Conditions:
  Type           Status
  Initialized    True
  Ready          True
  ContainersReady True
  PodScheduled   True
Volumes:
  default-token-pxw49:
    Type: Secret (a volume populated by a Secret)
    SecretName: default-token-pxw49
    Optional: false
QoS Class:       BestEffort
Node-Selectors:  <none>
Tolerations:     node.kubernetes.io/not-ready:NoExecute for 300s
                  node.kubernetes.io/unreachable:NoExecute for 300s
Events:
  Type    Reason      Age    From          Message
  ----    -
  Normal  Scheduled   <unknown>    default-scheduler    Successfully assigned default/kubia-liveness to minikube
  Normal  Pulled      83s       kubelet, minikube    Container image "mansi2210/shahm888:kubia-unhealthy" already present on machine
  Normal  Created     82s       kubelet, minikube    Created container kubia-unhealthy
  Normal  Started     82s       kubelet, minikube    Started container kubia-unhealthy
  Warning  Unhealthy   7s (x2 over 17s)  kubelet, minikube    Liveness probe failed: HTTP probe failed with statuscode: 500
```

## Introducing Replication Controllers

### Creating a Replication Controller

```
cs571@ubuntu:~/app_js$ vim kubia-rc.yaml
cs571@ubuntu:~/app_js$ cat kubia-rc.yaml
apiVersion: v1
kind: ReplicationController
metadata:
  name: kubia
spec:
  replicas: 3
  selector:
    app: kubia
  template:
    metadata:
      labels:
        app: kubia
    spec:
      containers:
      - name: kubia
        image: "mansi2210/shahm888:kubia"
        ports:
        - containerPort: 8080
cs571@ubuntu:~/app_js$ kubectl create -f kubia-rc.yaml
replicationcontroller/kubia created
```

### Seeing the Replication Controller in action

```
cs571@ubuntu:~/app_js$ kubectl get pods
NAME          READY   STATUS    RESTARTS   AGE
kubia-wznch   1/1     Running   0           114s
kubia-x4bcm   1/1     Running   0           114s
kubia-xt5jg   1/1     Running   0           114s
```

```
cs571@ubuntu:~/app_js$ kubectl delete pod kubia-wznch
pod "kubia-wznch" deleted
cs571@ubuntu:~/app_js$ kubectl get pods
NAME          READY   STATUS    RESTARTS   AGE
kubia-ptr25   1/1     Running   0           42s
kubia-x4bcm   1/1     Running   0          3m16s
kubia-xt5jg   1/1     Running   0          3m16s
cs571@ubuntu:~/app_js$
```

### Getting information about a Replication Controller

```
cs571@ubuntu:~/app_js$ kubectl get rc
NAME    DESIRED   CURRENT   READY   AGE
kubia   3         3         3       4m3s
```



## Displaying details of a Replication Controller with kubectl describe

```
cs571@ubuntu:~/app_js$ kubectl describe rc kubia
Name:          kubia
Namespace:     default
Selector:      app=kubia
Labels:        app=kubia
Annotations:   <none>
Replicas:      3 current / 3 desired
Pods Status:   3 Running / 0 Waiting / 0 Succeeded / 0 Failed
Pod Template:
  Labels:  app=kubia
  Containers:
    kubia:
      Image:        mansi2210/shahm888:kubia
      Port:         8080/TCP
      Host Port:    0/TCP
      Environment:  <none>
      Mounts:       <none>
      Volumes:      <none>
Events:
  Type      Reason              Age   From                      Message
  ----      -
  Normal    SuccessfulCreate    5m48s replication-controller     Created pod: kubia-xt5jg
  Normal    SuccessfulCreate    5m48s replication-controller     Created pod: kubia-x4bcm
  Normal    SuccessfulCreate    5m48s replication-controller     Created pod: kubia-wznch
  Normal    SuccessfulCreate    3m14s replication-controller     Created pod: kubia-ptr25
cs571@ubuntu:~/app_js$
```

## Moving pods in and out of the scope of a Replication Controller

```
cs571@ubuntu:~/app_js$ kubectl get pods
NAME          READY   STATUS    RESTARTS   AGE
kubia-ptr25   1/1     Running   0           33m
kubia-x4bcm   1/1     Running   0           36m
kubia-xt5jg   1/1     Running   0           36m
cs571@ubuntu:~/app_js$ kubectl label pod kubia-xt5jg type=special
pod/kubia-xt5jg labeled
cs571@ubuntu:~/app_js$ kubectl get pods --show-labels
NAME          READY   STATUS    RESTARTS   AGE   LABELS
kubia-ptr25   1/1     Running   0           33m   app=kubia
kubia-x4bcm   1/1     Running   0           36m   app=kubia
kubia-xt5jg   1/1     Running   0           36m   app=kubia,type=special
cs571@ubuntu:~/app_js$ kubectl label pod kubia-xt5jg app=foo --overwrite
pod/kubia-xt5jg labeled
cs571@ubuntu:~/app_js$ kubectl get pods -L app
NAME          READY   STATUS    RESTARTS   AGE   APP
kubia-ptr25   1/1     Running   0           34m   kubia
kubia-x4bcm   1/1     Running   0           37m   kubia
kubia-xt5jg   1/1     Running   0           37m   foo
kubia-z2pb4   1/1     Running   0           14s   kubia
cs571@ubuntu:~/app_js$
```

### Changing the pod template

```
cs571@ubuntu:~/app_js$ kubectl edit rc kubia
replicationcontroller/kubia edited
```

### Horizontally scaling pods

```
cs571@ubuntu:~/app_js$ kubectl scale rc kubia --replicas=10
replicationcontroller/kubia scaled
```

```
cs571@ubuntu:~/app_js$ kubectl edit rc kubia
replicationcontroller/kubia edited
cs571@ubuntu:~/app_js$ kubectl get rc
NAME      DESIRED   CURRENT   READY   AGE
kubia     10        10        3       50m
```

### Scaling down with the kubectl scale command

```
cs571@ubuntu:~/app_js$ kubectl scale rc kubia --replicas=3
replicationcontroller/kubia scaled
cs571@ubuntu:~/app_js$ kubectl get rc
NAME      DESIRED   CURRENT   READY   AGE
kubia     3         3         3       51m
```

### Deleting a Replication Controller

```
cs571@ubuntu:~/app_js$ kubectl delete rc kubia --cascade=false
replicationcontroller "kubia" deleted
cs571@ubuntu:~/app_js$ kubectl get rc
No resources found in default namespace.
```



## Creating and examining a Replica Set

```
cs571@ubuntu:~/app_js$ cat kubia-replicaset.yaml
apiVersion: apps/v1beta2
kind: ReplicaSet
metadata:
  name: kubia
spec:
  replicas: 3
  selector:
    matchLabels:
      app: kubia
  template:
    metadata:
      labels:
        app: kubia
    spec:
      containers:
      - name: kubia
        image: mansi2210/shahm888:kubia
```

```
cs571@ubuntu:~/app_js$ kubectl create -f kubia-replicaset.yaml
replicaset.apps/kubia created
cs571@ubuntu:~/app_js$ kubectl get rs
```

NAME	DESIRED	CURRENT	READY	AGE
kubia	3	3	3	10s

```
cs571@ubuntu:~/app_js$ kubectl describe rs
```

Name: kubia  
Namespace: default  
Selector: app=kubia  
Labels: <none>  
Annotations: <none>  
Replicas: 3 current / 3 desired  
Pods Status: 3 Running / 0 Waiting / 0 Succeeded / 0 Failed  
Pod Template:  
 Labels: app=kubia  
 Containers:  
 kubia:  
 Image: mansi2210/shahm888:kubia  
 Port: <none>  
 Host Port: <none>  
 Environment: <none>  
 Mounts: <none>  
 Volumes: <none>

Events:

Type	Reason	Age	From	Message
Normal	SuccessfulCreate	37s	replicaset-controller	Created pod: kubia-jkq9h
Normal	SuccessfulCreate	37s	replicaset-controller	Created pod: kubia-wgtg4
Normal	SuccessfulCreate	37s	replicaset-controller	Created pod: kubia-bkz9x

### Using the Replicase's more expressive label selectors

```
cs571@ubuntu:~/app_js$ cat kubia-replicaset-matchexpressions.yaml
apiVersion: apps/v1
kind: ReplicaSet
metadata:
  name: kubia
spec:
  replicas: 3
  selector:
    matchExpressions:
      - key: app
        operator: In
        values:
          - kubia
  template:
    metadata:
      labels:
        app: kubia
    spec:
      containers:
        - name: kubia
          image: mansi2210/shahm888:kubia
```

### Wrapping up Replica Sets

```
cs571@ubuntu:~/app_js$ kubectl delete rs kubia
replicaset.apps "kubia" deleted
```

## Daemon Set:

```
cs571@ubuntu:~$ mkdir busybox
cs571@ubuntu:~$ cd busybox
cs571@ubuntu:~/busybox$ vim Dockerfile
cs571@ubuntu:~/busybox$ docker build -t busybox .
Sending build context to Docker daemon 2.048kB
Step 1/2 : FROM busybox
latest: Pulling from library/busybox
bdbbaa22dec6: Pull complete
Digest: sha256:6915be4043561d64e0ab0f8f098dc2ac48e077fe23f488ac24b665166898115a
Status: Downloaded newer image for busybox:latest
--> 6d5fcfe5ff17
Step 2/2 : ENTRYPOINT while true; do echo 'SSD OK'; sleep 5; done
--> Running in 48efdfcc1849
Removing intermediate container 48efdfcc1849
--> 86b6e2f6226e
Successfully built 86b6e2f6226e
Successfully tagged busybox:latest
cs571@ubuntu:~/busybox$ docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
busybox              latest              86b6e2f6226e       7 seconds ago      1.22MB
kubia-unhealthy      latest              6783c694a02e       2 hours ago        660MB
mansi2210/shahm888   kubia-unhealthy    6783c694a02e       2 hours ago        660MB
kubia                latest             fac6faaace7c       4 weeks ago        660MB
mansi2210/shahm888   kubia              fac6faaace7c       4 weeks ago        660MB
ubuntu              latest             ccc6e87d482b       6 weeks ago        64.2MB
busybox              <none>             6d5fcfe5ff17       2 months ago       1.22MB
hello-world          latest             fce289e99eb9       14 months ago      1.84kB
node                 7                  d9aed20b68a4       2 years ago        660MB
cs571@ubuntu:~/busybox$ docker tag busybox mansi2210/shahm888:busybox
cs571@ubuntu:~/busybox$ docker push mansi2210/shahm888:busybox
The push refers to repository [docker.io/mansi2210/shahm888]
195be5f8beid: Mounted from library/busybox
busybox: digest: sha256:897e594cbbeca50f56bc878212caf9d620bb2ddd0201035dbf20dea244d1b380 size: 527
cs571@ubuntu:~/busybox$ vim ssd-monitor-daemonset.yaml
cs571@ubuntu:~/busybox$ kubectl create -f ssd-monitor-daemonset.yaml
error: unable to recognize "ssd-monitor-daemonset.yaml": no matches for kind "DaemonSet" in version "apps/v1beta2"
cs571@ubuntu:~/busybox$ vim ssd-monitor-daemonset.yaml
cs571@ubuntu:~/busybox$ kubectl create -f ssd-monitor-daemonset.yaml
daemonset.apps/ssd-monitor created
```

## Adding the required label to your node (s)

```
cs571@ubuntu:~/busybox$ kubectl get ds
NAME          DESIRED   CURRENT   READY   UP-TO-DATE   AVAILABLE   NODE SELECTOR   AGE
ssd-monitor   0         0         0       0             0           disk=ssd        65s
cs571@ubuntu:~/busybox$ kubectl get po
No resources found in default namespace.
cs571@ubuntu:~/busybox$ kubectl get node
NAME          STATUS    ROLES    AGE   VERSION
minikube      Ready     master   18d   v1.17.2
cs571@ubuntu:~/busybox$ kubectl label node minikube disk=ssd
node/minikube labeled
cs571@ubuntu:~/busybox$ kubectl get po
NAME          READY   STATUS    RESTARTS   AGE
ssd-monitor-zq6l7  1/1     Running   0           10s
cs571@ubuntu:~/busybox$
```

## Removing the required label from the node

```
cs571@ubuntu:~/busybox$ kubectl get po
NAME          READY   STATUS    RESTARTS   AGE
ssd-monitor-zq6l7  1/1     Running   0           10s
cs571@ubuntu:~/busybox$ kubectl label node minikube disk=hdd --overwrite
node/minikube labeled
cs571@ubuntu:~/busybox$ kubectl get po
NAME          READY   STATUS    RESTARTS   AGE
ssd-monitor-zq6l7  1/1     Terminating   0           51s
cs571@ubuntu:~/busybox$
```



## Defining a Job resource

```
cs571@ubuntu:~$ mkdir batchjob
cs571@ubuntu:~$ cd batchjob/
cs571@ubuntu:~/batchjob$ vim Dockerfile
cs571@ubuntu:~/batchjob$ docker build -t batchjob .
Sending build context to Docker daemon 2.048kB
Step 1/2 : FROM busybox
--> 86b6e2f6226e
Step 2/2 : ENTRYPOINT echo "$(date) Batch job starting"; sleep 120; echo "$(date) Finished successfully"
--> Running in 45bd3263f350
Removing intermediate container 45bd3263f350
--> 75ba5741b4b0
Successfully built 75ba5741b4b0
Successfully tagged batchjob:latest
cs571@ubuntu:~/batchjob$ docker tag batchjob mansi2210/shahm888:batchjob
cs571@ubuntu:~/batchjob$ docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
batchjob	latest	75ba5741b4b0	29 seconds ago	1.22MB
mansi2210/shahm888	batchjob	75ba5741b4b0	29 seconds ago	1.22MB
busybox	latest	86b6e2f6226e	9 minutes ago	1.22MB
mansi2210/shahm888	busybox	86b6e2f6226e	9 minutes ago	1.22MB
kubia-unhealthy	latest	6783c694a02e	2 hours ago	660MB
mansi2210/shahm888	kubia-unhealthy	6783c694a02e	2 hours ago	660MB
mansi2210/shahm888	kubia	fac6faaace7c	4 weeks ago	660MB
kubia	latest	fac6faaace7c	4 weeks ago	660MB
ubuntu	latest	ccc6e87d482b	6 weeks ago	64.2MB
busybox	<none>	6d5fcfe5ff17	2 months ago	1.22MB
hello-world	latest	fce289e99eb9	14 months ago	1.84kB
node	7	d9aed20b68a4	2 years ago	660MB

```
cs571@ubuntu:~/batchjob$ docker push mansi2210/shahm888:batchjob
The push refers to repository [docker.io/mansi2210/shahm888]
195be5f8be1d: Layer already exists
batchjob: digest: sha256:42816b8d836dd1b024ea1cffe1a89310c0c68486a620ee23f9f7b6f86300cc34 size: 527
cs571@ubuntu:~/batchjob$ vim time-limited-batch-job.yaml
cs571@ubuntu:~/batchjob$ kubectl create -f time-limited-batch-job.yaml
job.batch/time-limited-batch-job created
cs571@ubuntu:~/batchjob$ kubectl get jobs
```

NAME	COMPLETIONS	DURATION	AGE
time-limited-batch-job	0/1	18s	18s

```
cs571@ubuntu:~/batchjob$ kubectl get po
```

NAME	READY	STATUS	RESTARTS	AGE
time-limited-batch-job-2dftj	1/1	Running	0	26s

```
cs571@ubuntu:~/batchjob$
```

```
cs571@ubuntu:~/batchjob$ kubectl get po
```

NAME	READY	STATUS	RESTARTS	AGE
time-limited-batch-job-jvcfc	1/1	Terminating	0	47s

```
cs571@ubuntu:~/batchjob$ kubectl logs time-limited-batch-job-jvcfc
Tue Mar 3 10:08:31 UTC 2020 Batch job starting
cs571@ubuntu:~/batchjob$ kubectl get po
```

NAME	READY	STATUS	RESTARTS	AGE
time-limited-batch-job-jvcfc	1/1	Terminating	0	56s

```
cs571@ubuntu:~/batchjob$
```

```
cs571@ubuntu:~/batchjob$ kubectl get job
```

NAME	COMPLETIONS	DURATION	AGE
time-limited-batch-job	0/1	2m56s	2m56s

```
cs571@ubuntu:~/batchjob$
```

## Running multiple pod instances in a Job

```
cs571@ubuntu:~/batchjob$ cat multi-completion-batch-job.yaml
apiVersion: batch/v1
kind: Job
metadata:
  name: multi-completion-batch-job
spec:
  completions: 5
  template:
    metadata:
      labels:
        app: batch-job
    spec:
      restartPolicy: OnFailure
      containers:
      - name: main
        image: mansi2210/shahm888:batchjob

cs571@ubuntu:~/batchjob$ cat multi-completion-parallel-batch-job.yaml
apiVersion: batch/v1
kind: Job
metadata:
  name: multi-completion-batch-job
spec:
  completions: 5
  parallelism: 2
  template:
    metadata:
      labels:
        app: batch-job
    spec:
      restartPolicy: OnFailure
      containers:
      - name: main
        image: mansi2210/shahm888:batchjob
```

```
cs571@ubuntu:~/batchjob$ kubectl create -f multi-completion-batch-job.yaml
job.batch/multi-completion-batch-job created
cs571@ubuntu:~/batchjob$ kubectl get jobs
NAME                                COMPLETIONS  DURATION  AGE
multi-completion-batch-job         0/5           4s        4s
cs571@ubuntu:~/batchjob$ kubectl get po
NAME                                READY  STATUS   RESTARTS  AGE
multi-completion-batch-job-9sslc    1/1    Running  0         7s
cs571@ubuntu:~/batchjob$ kubectl scale job multi-completion-batch-job --replicas 3
Error from server (NotFound): the server could not find the requested resource
cs571@ubuntu:~/batchjob$ kubectl get jobs
NAME                                COMPLETIONS  DURATION  AGE
multi-completion-batch-job         1/5           2m24s     2m24s
cs571@ubuntu:~/batchjob$ kubectl get po
NAME                                READY  STATUS   RESTARTS  AGE
multi-completion-batch-job-9sslc    0/1    Completed  0         2m29s
multi-completion-batch-job-rrmgw    1/1    Running   0         26s
cs571@ubuntu:~/batchjob$ kubectl scale job multi-completion-batch-job --replicas 3
Error from server (NotFound): the server could not find the requested resource
cs571@ubuntu:~/batchjob$ kubectl get jobs
NAME                                COMPLETIONS  DURATION  AGE
multi-completion-batch-job         1/5           2m38s     2m38s
cs571@ubuntu:~/batchjob$
```



## Cronjob

```
cs571@ubuntu:~/batchjob$ cat cronjob.yaml
apiVersion: batch/v1beta1
kind: CronJob
metadata:
  name: batch-job-every-fifteen-minutes
spec:
  schedule: "0,15,30,45 * * * *"
  jobTemplate:
    spec:
      template:
        metadata:
          labels:
            app: periodic-batch-job
        spec:
          restartPolicy: OnFailure
          containers:
            - name: main
              image: mansi2210/shahm888:batchjob
```

```
cs571@ubuntu:~/batchjob$ kubectl create -f cronjob.yaml
cronjob.batch/batch-job-every-fifteen-minutes created
```

```
cs571@ubuntu:~/batchjob$ kubectl get cronjob
```

NAME	SCHEDULE	SUSPEND	ACTIVE	LAST SCHEDULE	AGE
batch-job-every-fifteen-minutes	0,15,30,45 * * * *	False	0	<none>	14s

## Understanding how scheduled jobs are run

```
cs571@ubuntu:~/batchjob$ kubectl delete cronjobs --all
cronjob.batch "batch-job-every-fifteen-minutes" deleted
```

```
cs571@ubuntu:~/batchjob$ cat cronjob.yaml
apiVersion: batch/v1beta1
kind: CronJob
metadata:
  name: batch-job-every-fifteen-minutes
spec:
  schedule: "0,15,30,45 * * * *"
  startingDeadlineSeconds: 15
  jobTemplate:
    spec:
      template:
        metadata:
          labels:
            app: periodic-batch-job
        spec:
          restartPolicy: OnFailure
          containers:
            - name: main
              image: mansi2210/shahm888:bat
```

```
cs571@ubuntu:~/batchjob$ kubectl create -f cronjob.yaml
cronjob.batch/batch-job-every-fifteen-minutes created
```

```
cs571@ubuntu:~/batchjob$ kubectl get cronjob
```

NAME	SCHEDULE	SUSPEND	ACTIVE	LAST SCHEDULE	AGE
batch-job-every-fifteen-minutes	0 15,30,45 * * * *	False	0	<none>	7s

```
cs571@ubuntu:~/batchjob$ kubectl get po
```

NAME	READY	STATUS	RESTARTS	AGE
batch-job-every-fifteen-minutes-1583231400-lrq9w	1/1	Terminating	0	37s
multi-completion-batch-job-5zspj	1/1	Running	0	61s
multi-completion-batch-job-9jbvq	0/1	Completed	0	5m10s
multi-completion-batch-job-9sslc	0/1	Completed	0	9m18s
multi-completion-batch-job-c7wj9	0/1	Completed	0	3m5s
multi-completion-batch-job-rrmgw	0/1	Completed	0	7m15s