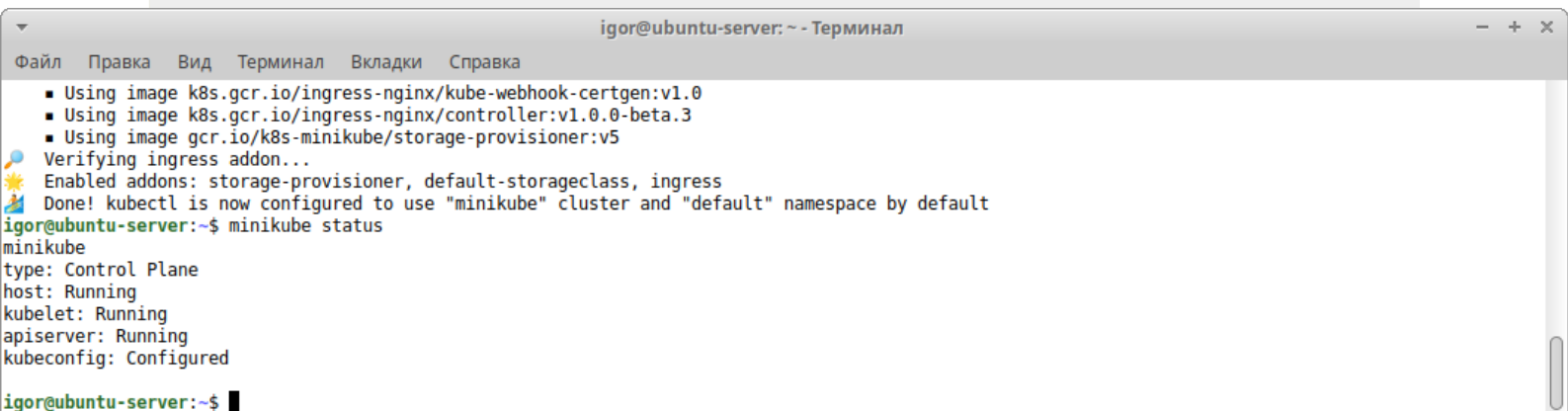


```
minikube start
```

```
minikube status
```

A terminal window titled "igor@ubuntu-server: ~ - Терминал" with a menu bar containing "Файл", "Правка", "Вид", "Терминал", "Вкладки", and "Справка". The terminal output shows the execution of "minikube start" and "minikube status". The "start" command lists three images being used, verifies the ingress addon, and enables addons: storage-provisioner, default-storageclass, and ingress. It also shows a message that kubectl is now configured to use the "minikube" cluster and "default" namespace. The "status" command shows the minikube type as Control Plane, host as Running, kubelet as Running, apiserver as Running, and kubeconfig as Configured.

```
igor@ubuntu-server: ~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
■ Using image k8s.gcr.io/ingress-nginx/kube-webhook-certgen:v1.0
■ Using image k8s.gcr.io/ingress-nginx/controller:v1.0.0-beta.3
■ Using image gcr.io/k8s-minikube/storage-provisioner:v5
🔍 Verifying ingress addon...
🌟 Enabled addons: storage-provisioner, default-storageclass, ingress
🎉 Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
igor@ubuntu-server:~$ minikube status
minikube
type: Control Plane
host: Running
kubelet: Running
apiserver: Running
kubeconfig: Configured
igor@ubuntu-server:~$
```

```
vi daemonset1.yaml
```

```
igor@ubuntu-server: ~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка

---
apiVersion: apps/v1
kind: DaemonSet
metadata:
  labels:
    app: node-exporter
  name: node-exporter
spec:
  updateStrategy:
    rollingUpdate:
      maxUnavailable: 1
    type: RollingUpdate
  selector:
    matchLabels:
      app: node-exporter
  template:
    metadata:
      labels:
        app: node-exporter
    spec:
      containers:
      - args:
        - --web.listen-address=0.0.0.0:9101
        - --path.procfs=/host/proc
        - --path.sysfs=/host/sys
        - --collector.filesystem.ignored-mount-points=^/(dev|proc|sys|var/lib/docker/.+)($/|)
        - --collector.filesystem.ignored-fs-types=^(autofs|binfmt_misc|cgroup|configfs|debugfs|devpts|devtmpfs|fusectl|hugetlbfs|mqueue|overlay|proc|p
rocs|pstore|rpc_pipefs|securityfs|sysfs|tracefs)$
        image: quay.io/prometheus/node-exporter:v0.16.0
        imagePullPolicy: IfNotPresent
        name: node-exporter
        volumeMounts:
        - mountPath: /host/proc
          name: proc
        - mountPath: /host/sys
          name: sys
        - mountPath: /host/root
          name: root
          readOnly: true
      hostNetwork: true
      hostPID: true
      tolerations:
      - effect: NoSchedule
        operator: Exists
      nodeSelector:
        beta.kubernetes.io/os: linux
      volumes:
      - hostPath:
          path: /proc
          type: ""
          name: proc
      - hostPath:
          path: /sys
          type: ""
          name: sys
      - hostPath:
          path: /
          type: ""
          name: root
```

```
kubectl create -f daemonset1.yaml
```

```
Warning: spec.template.spec.nodeSelector[beta.kubernetes.io/os]: deprecated
since v1.14; use "kubernetes.io/os" instead
```

```
Error from server (AlreadyExists): error when creating "daemonset1.yaml":
daemonsets.apps "node-exporter" already exists
```

```
kubectl get daemonset
```

```
kubectl describe daemonset node-exporter
```

```
igor@ubuntu-server: ~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
igor@ubuntu-server:~$ kubectl create -f daemonset1.yaml
Warning: spec.template.spec.nodeSelector[beta.kubernetes.io/os]: deprecated since v1.14; use "kubernetes.io/os" instead
daemonset.apps/node-exporter created
igor@ubuntu-server:~$ 
igor@ubuntu-server:~$ kubectl describe daem
error: the server doesn't have a resource type "daem"
igor@ubuntu-server:~$ kubectl get daemonset
NAME           DESIRED   CURRENT   READY   UP-TO-DATE   AVAILABLE   NODE SELECTOR   AGE
node-exporter   1         1         1       1            1           beta.kubernetes.io/os=linux   37s
igor@ubuntu-server:~$ kubectl describe daemonset node-exporter
Name:          node-exporter
Selector:      app=node-exporter
Node-Selector: beta.kubernetes.io/os=linux
Labels:        app=node-exporter
Annotations:   deprecated.daemonset.template.generation: 1
Desired Number of Nodes Scheduled: 1
Current Number of Nodes Scheduled: 1
Number of Nodes Scheduled with Up-to-date Pods: 1
Number of Nodes Scheduled with Available Pods: 1
Number of Nodes Misscheduled: 0
Pods Status:  1 Running / 0 Waiting / 0 Succeeded / 0 Failed
Pod Template:
  Labels:  app=node-exporter
  Containers:
    node-exporter:
      Image:      quay.io/prometheus/node-exporter:v0.16.0
      Port:      <none>
      Host Port:  <none>
      Args:
        --web.listen-address=0.0.0.0:9101
        --path.procfs=/host/proc
        --path.sysfs=/host/sys
        --collector.filesystem.ignored-mount-points=^/(dev|proc|sys|var/lib/docker/.+)($/|)
        --collector.filesystem.ignored-fs-types=^(autofs|binfmt_misc|cgroup|configfs|debugfs|devpts|devtmpfs|fusectl|hugetlbfs|mqueue|overlay|proc|procfs|pstore|rpc_pipefs|securityfs|sysfs|tracefs)$
      Environment:  <none>
      Mounts:
        /host/proc from proc (rw)
        /host/root from root (ro)
        /host/sys from sys (rw)
  Volumes:
    proc:
      Type:      HostPath (bare host directory volume)
      Path:      /proc
      HostPathType:
    sys:
      Type:      HostPath (bare host directory volume)
      Path:      /sys
      HostPathType:
    root:
      Type:      HostPath (bare host directory volume)
      Path:      /
      HostPathType:
Events:
  Type    Reason          Age    From          Message
  ----    -
Normal SuccessfulCreate 54s    daemonset-controller  Created pod: node-exporter-vpzsp
igor@ubuntu-server:~$
```

vi statefulset.yaml

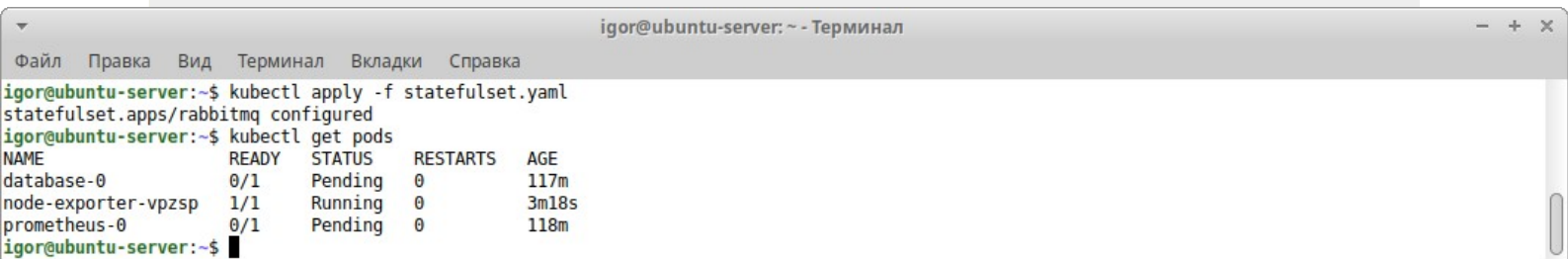
```

---
apiVersion: apps/v1
kind: StatefulSet
metadata:
  name: rabbitmq
spec:
  serviceName: rabbitmq
  replicas: 3
  selector:
    matchLabels:
      app: rabbitmq
  template:
    metadata:
      labels:
        app: rabbitmq
    spec:
      serviceAccount: rabbitmq
      terminationGracePeriodSeconds: 10
      containers:
        - name: rabbitmq-k8s
          image: rabbitmq:3.7-management
          env:
            - name: MY_POD_IP
              valueFrom:
                fieldRef:
                  fieldPath: status.podIP
            - name: RABBITMQ_USE_LONGNAME
              value: "true"
            - name: RABBITMQ_NODENAME
              value: "rabbit@$(MY_POD_IP)"
            - name: K8S_SERVICE_NAME
              value: "rabbitmq"
            - name: RABBITMQ_ERLANG_COOKIE
              value: "mycookie"
          ports:
            - name: amqp
              protocol: TCP
              containerPort: 5672
            - name: admin
              protocol: TCP
              containerPort: 15672
          livenessProbe:
            exec:
              command: ["rabbitmqctl", "status"]
            initialDelaySeconds: 60
            periodSeconds: 60
            timeoutSeconds: 15
          readinessProbe:
            exec:
              command: ["rabbitmqctl", "status"]
            initialDelaySeconds: 20
            periodSeconds: 60
            timeoutSeconds: 10
          imagePullPolicy: Always
          volumeMounts:
            - name: config-volume
              mountPath: /etc/rabbitmq
            - name: data
              mountPath: /var/lib/rabbitmq
      volumes:
        - name: config-volume
          configMap:
            name: rabbitmq-config
            items:
              - key: rabbitmq.conf
                path: rabbitmq.conf
              - key: enabled_plugins
                path: enabled_plugins
      affinity:
        podAntiAffinity:
          preferredDuringSchedulingIgnoredDuringExecution:
            - weight: 100
              podAffinityTerm:
                labelSelector:
                  matchExpressions:
                    - key: app
                      operator: In
                      values:
                        - rabbitmq
                topologyKey: kubernetes.io/hostname
  volumeClaimTemplates:
    - metadata:
        name: data
      spec:
        accessModes: ["ReadWriteOnce"]
        resources:
          requests:
            storage: 1Gi
        storageClassName: csi-ceph-hdd-dp1

```

```
kubectl apply -f statefulset.yaml
```

```
kubectl get pods
```



A terminal window titled "igor@ubuntu-server: ~ - Терминал" with a menu bar containing "Файл", "Правка", "Вид", "Терминал", "Вкладки", and "Справка". The terminal shows the following commands and output:

```
igor@ubuntu-server:~$ kubectl apply -f statefulset.yaml
statefulset.apps/rabbitmq configured
igor@ubuntu-server:~$ kubectl get pods
```

NAME	READY	STATUS	RESTARTS	AGE
database-0	0/1	Pending	0	117m
node-exporter-vpzsp	1/1	Running	0	3m18s
prometheus-0	0/1	Pending	0	118m

```
igor@ubuntu-server:~$
```

```
kubectl get statefulset
```

```
kubectl describe statefulset rabbitmq
```

```
igor@ubuntu-server: ~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка

igor@ubuntu-server:~$ kubectl get statefulset
NAME          READY   AGE
database      0/1     130d
prometheus    0/1     135d
rabbitmq      0/3     3h42m
igor@ubuntu-server:~$ kubectl describe statefulset rabbitmq
Name:          rabbitmq
Namespace:     default
CreationTimestamp: Sun, 27 Mar 2022 16:47:19 +0000
Selector:      app=rabbitmq
Labels:        <none>
Annotations:   <none>
Replicas:      3 desired | 0 total
Update Strategy: RollingUpdate
  Partition:    0
Pods Status:    0 Running / 0 Waiting / 0 Succeeded / 0 Failed
Pod Template:
  Labels:       app=rabbitmq
  Service Account: rabbitmq
  Containers:
    rabbitmq-k8s:
      Image:      rabbitmq:3.7-management
      Ports:      5672/TCP, 15672/TCP
      Host Ports: 0/TCP, 0/TCP
      Liveness:    exec [rabbitmqctl status] delay=60s timeout=15s period=60s #success=1 #failure=3
      Readiness:   exec [rabbitmqctl status] delay=20s timeout=10s period=60s #success=1 #failure=3
      Environment:
        MY_POD_IP: (v1:status.podIP)
        RABBITMQ_USE_LONGNAME: true
        RABBITMQ_NODENAME: rabbit@$(MY_POD_IP)
        K8S_SERVICE_NAME: rabbitmq
        RABBITMQ_ERLANG_COOKIE: mycookie
      Mounts:
        /etc/rabbitmq from config-volume (rw)
        /var/lib/rabbitmq from data (rw)
  Volumes:
    config-volume:
      Type:      ConfigMap (a volume populated by a ConfigMap)
      Name:      rabbitmq-config
      Optional:   false
Volume Claims:
  Name:          data
  StorageClass:  csi-ceph-hdd-dp1
  Labels:        <none>
  Annotations:   <none>
  Capacity:      1Gi
  Access Modes:  [ReadWriteOnce]
Events:
  Type      Reason      Age          From          Message
  ----      -
Warning    FailedCreate  84m (x26 over 3h43m)  statefulset-controller  create Pod rabbitmq-0 in StatefulSet rabbitmq failed error: pods "rabbitmq-0" is forbidden: error looking up service account default/rabbitmq: serviceaccount "rabbitmq" not found
Warning    FailedCreate  3m49s (x17 over 9m17s) statefulset-controller  create Pod rabbitmq-0 in StatefulSet rabbitmq failed error: pods "rabbitmq-0" is forbidden: error looking up service account default/rabbitmq: serviceaccount "rabbitmq" not found
igor@ubuntu-server:~$
```

vi job2.yaml

```
igor@ubuntu-server: ~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка

---
apiVersion: batch/v1
kind: Job
metadata:
  name: hello-job
spec:
  backoffLimit: 2
  activeDeadlineSeconds: 60
  template:
    spec:
      containers:
      - name: hello-job
        image: busybox
        args:
        - /bin/sh
        - -c
        - while true; do sleep 1; date; echo Hello from the Kubernetes cluster; done
      restartPolicy: Never
```

```
kubectl apply -f job2.yaml
```

```
kubectl get pods
```

```
igor@ubuntu-server: ~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
igor@ubuntu-server:~$ kubectl apply -f job2.yaml
job.batch/hello-job unchanged
igor@ubuntu-server:~$ kubectl get pods
NAME                READY   STATUS    RESTARTS   AGE
database-0          0/1     Pending   0           121m
node-exporter-vpzsp 1/1     Running   0           7m31s
prometheus-0        0/1     Pending   0           122m
igor@ubuntu-server:~$
```

```
kubectl get jobs
```

```
kubectl describe job hello-job
```

```
igor@ubuntu-server: ~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
igor@ubuntu-server:~$ kubectl get jobs
NAME            COMPLETIONS   DURATION   AGE
hello-job       0/1            157m       157m
igor@ubuntu-server:~$ kubectl describe job hello-job
Name:             hello-job
Namespace:        default
Selector:         controller-uid=7d8ac3c8-80b2-4534-83f5-bcc468d9e67f
Labels:           controller-uid=7d8ac3c8-80b2-4534-83f5-bcc468d9e67f
                  job-name=hello-job
Annotations:      <none>
Parallelism:      1
Completions:      1
Completion Mode:   NonIndexed
Start Time:       Sun, 27 Mar 2022 17:57:03 +0000
Active Deadline Seconds: 60s
Pods Statuses:    0 Active / 0 Succeeded / 1 Failed
Pod Template:
  Labels:  controller-uid=7d8ac3c8-80b2-4534-83f5-bcc468d9e67f
          job-name=hello-job
  Containers:
    hello-job:
      Image:      busybox
      Port:       <none>
      Host Port:  <none>
      Args:
        /bin/sh
        -c
        while true; do sleep 1; date; echo Hello from the Kubernetes cluster; done
      Environment:  <none>
      Mounts:       <none>
      Volumes:      <none>
  Events:          <none>
igor@ubuntu-server:~$
```

```
kubectl get pods
```

```
kubectl get pods -all
```

```
kubectl get pods --all-namespaces
```



```
igor@ubuntu-server: ~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка

igor@ubuntu-server:~$ kubectl get pods
NAME                READY   STATUS    RESTARTS   AGE
database-0          0/1     Pending   0           123m
node-exporter-vpzsp 1/1     Running   0           9m27s
prometheus-0        0/1     Pending   0           124m
igor@ubuntu-server:~$ kubectl get pods -all
error: unknown shorthand flag: 'a' in -all
See 'kubectl get --help' for usage.
igor@ubuntu-server:~$ kubectl get pods --all-namespaces
NAMESPACE   NAME                                                    READY   STATUS    RESTARTS   AGE
default     database-0                                              0/1     Pending   0           123m
default     node-exporter-vpzsp                                    1/1     Running   0           10m
default     prometheus-0                                           0/1     Pending   0           125m
gitlab       gitlab-runner-7467648488-2mdf6                        1/1     Running   103 (2m31s ago)  127d
ingress-nginx ingress-nginx-admission-create--1-5z22r                0/1     Completed   0           138d
ingress-nginx ingress-nginx-admission-patch--1-sgdmc                 0/1     Completed   1           138d
ingress-nginx ingress-nginx-controller-69bdbc4d57-z67vb          1/1     Running   15 (15m ago)    138d
kube-system  coredns-78fcd69978-7c8pw                             1/1     Running   18 (15m ago)    146d
kube-system  etcd-minikube                                          1/1     Running   18 (15m ago)    146d
kube-system  kube-apiserver-minikube                              1/1     Running   18 (15m ago)    146d
kube-system  kube-controller-manager-minikube                     1/1     Running   18 (15m ago)    146d
kube-system  kube-proxy-4zwp6                                      1/1     Running   18 (15m ago)    146d
kube-system  kube-scheduler-minikube                             1/1     Running   18 (15m ago)    146d
kube-system  storage-provisioner                                   1/1     Running   31 (15m ago)    146d
prod         database-0                                              0/1     Pending   0           127d
prod         geekbrains-55df48c88d-htjs5                          1/1     Running   9 (15m ago)     127d
prod         geekbrains-55df48c88d-mv8wk                          1/1     Running   9 (15m ago)     127d
redmine     pg-db-757b89cf9-p8dtq                                0/1     Pending   0           139d
redmine     redmine-app-58fc94cf66-kl4nw                         0/1     CrashLoopBackOff 473 (3m34s ago)  139d
stage       database-0                                              0/1     Pending   0           127d
stage       geekbrains-55df48c88d-7qgzp                          1/1     Running   9 (15m ago)     127d
stage       geekbrains-55df48c88d-gf5tm                          1/1     Running   9 (15m ago)     127d
igor@ubuntu-server:~$
```

kubectl describe job hello-job

```
igor@ubuntu-server: ~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка

igor@ubuntu-server:~$ kubectl describe job hello-job
Name:             hello-job
Namespace:        default
Selector:         controller-uid=7d8ac3c8-80b2-4534-83f5-bcc468d9e67f
Labels:           controller-uid=7d8ac3c8-80b2-4534-83f5-bcc468d9e67f
                  job-name=hello-job
Annotations:      <none>
Parallelism:      1
Completions:      1
Completion Mode:   NonIndexed
Start Time:       Sun, 27 Mar 2022 17:57:03 +0000
Active Deadline Seconds: 60s
Pods Statuses:    0 Active / 0 Succeeded / 1 Failed
Pod Template:
  Labels:  controller-uid=7d8ac3c8-80b2-4534-83f5-bcc468d9e67f
          job-name=hello-job
  Containers:
    hello-job:
      Image:      busybox
      Port:       <none>
      Host Port:  <none>
      Args:
        /bin/sh
        -c
        while true; do sleep 1; date; echo Hello from the Kubernetes cluster; done
      Environment:  <none>
      Mounts:       <none>
      Volumes:      <none>
  Events:          <none>
igor@ubuntu-server:~$
```

vi cronjob.yaml



```
igor@ubuntu-server: ~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка

apiVersion: batch/v1beta1
kind: CronJob
metadata:
  name: hello-cron-job
spec:
  schedule: "*/1 * * * *"
  concurrencyPolicy: Allow
  jobTemplate:
    spec:
      backoffLimit: 2
      activeDeadlineSeconds: 100
      template:
        spec:
          containers:
            - name: hello-cron-job
              image: busybox
              args:
                - /bin/sh
                - -c
                - date; echo Hello from the Kubernetes cluster
          restartPolicy: Never
```

```
kubectl apply -f cronjob.yaml
```

Warning: batch/v1beta1 CronJob is deprecated in v1.21+, unavailable in v1.25+; use batch/v1 CronJob

```
cronjob.batch/hello-cron-job created
```

```
kubectl get pods
```

```
igor@ubuntu-server: ~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка

igor@ubuntu-server:~$ kubectl apply -f cronjob.yaml
Warning: batch/v1beta1 CronJob is deprecated in v1.21+, unavailable in v1.25+; use batch/v1 CronJob
cronjob.batch/hello-cron-job created
igor@ubuntu-server:~$ kubectl get pods
NAME                READY   STATUS    RESTARTS   AGE
database-0          0/1     Pending   0           127m
node-exporter-vpzsp 1/1     Running   0           13m
prometheus-0        0/1     Pending   0           128m
igor@ubuntu-server:~$
```

```
kubectl get cronjobs
```

```
kubectl describe job hello-cron-job
```

```
igor@ubuntu-server: ~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка

igor@ubuntu-server:~$ kubectl get cronjobs
NAME                SCHEDULE    SUSPEND   ACTIVE   LAST SCHEDULE   AGE
hello-cron-job      */1 * * * *   False     0        47s             90s

igor@ubuntu-server:~$ kubectl describe job hello-cron-job
Name:                hello-cron-job-27473560
Namespace:           default
Selector:             controller-uid=57f23dfc-0dd2-4f56-a129-3adf2dd7a86d
Labels:              controller-uid=57f23dfc-0dd2-4f56-a129-3adf2dd7a86d
                    job-name=hello-cron-job-27473560
Annotations:         <none>
Controlled By:       CronJob/hello-cron-job
Parallelism:         1
Completions:         1
Completion Mode:     NonIndexed
Start Time:          Sun, 27 Mar 2022 20:40:00 +0000
Completed At:        Sun, 27 Mar 2022 20:40:03 +0000
Duration:            3s
Active Deadline Seconds: 100s
Pods Statuses:       0 Active / 1 Succeeded / 0 Failed
Pod Template:
  Labels:  controller-uid=57f23dfc-0dd2-4f56-a129-3adf2dd7a86d
          job-name=hello-cron-job-27473560
  Containers:
    hello-cron-job:
      Image:      busybox
      Port:       <none>
      Host Port:  <none>
      Args:
        /bin/sh
        -c
        date; echo Hello from the Kubernetes cluster
      Environment: <none>
      Mounts:      <none>
      Volumes:     <none>
Events:
  Type      Reason              Age   From          Message
  ----      -
  Normal    SuccessfulCreate    58s   job-controller Created pod: hello-cron-job-27473560--1-hr2jv
  Normal    Completed           55s   job-controller Job completed

igor@ubuntu-server:~$
```

kubectl get pods

kubectl get jobs

kubectl get cronjobs

kubectl delete cronjob hello-cron-job

```
igor@ubuntu-server: ~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка

igor@ubuntu-server:~$ kubectl get pods
NAME                READY   STATUS    RESTARTS   AGE
database-0          0/1     Pending   0           129m
hello-cron-job-27473560--1-hr2jv  0/1     Completed  0           107s
hello-cron-job-27473561--1-8dh4b  0/1     Completed  0           47s
node-exporter-vpzsp  1/1     Running   0           15m
prometheus-0        0/1     Pending   0           131m

igor@ubuntu-server:~$ kubectl get jobs
NAME                COMPLETIONS   DURATION   AGE
hello-cron-job-27473560  1/1           3s         2m17s
hello-cron-job-27473561  1/1           2s         77s
hello-cron-job-27473562  1/1           3s         17s
hello-job              0/1           165m       165m

igor@ubuntu-server:~$ kubectl get cronjobs
NAME                SCHEDULE    SUSPEND   ACTIVE   LAST SCHEDULE   AGE
hello-cron-job      */1 * * * *   False     0        27s             3m10s

igor@ubuntu-server:~$ kubectl delete cronjob hello-cron-job
cronjob.batch "hello-cron-job" deleted

igor@ubuntu-server:~$
```

```
vi php-apache.yaml
```

```
igor@ubuntu-server: ~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка

apiVersion: apps/v1
kind: Deployment
metadata:
  name: php-apache-hps
spec:
  selector:
    matchLabels:
      run: php-apache-hps
  replicas: 1
  template:
    metadata:
      labels:
        run: php-apache-hps
    spec:
      containers:
        - name: php-apache-hps
          image: k8s.gcr.io/hpa-example
          ports:
            - containerPort: 80
          resources:
            limits:
              cpu: 500m
            requests:
              cpu: 200m
---
apiVersion: v1
kind: Service
metadata:
  name: php-apache-hps
  labels:
    run: php-apache-hps
spec:
  ports:
    - port: 80
  selector:
    run: php-apache-hps
~
:█
```

```
kubectl apply -f php-apache.yaml
```

```
kubectl get pods
```

```
igor@ubuntu-server: ~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка

igor@ubuntu-server:~$ kubectl apply -f php-apache.yaml
deployment.apps/php-apache-hps created
service/php-apache-hps unchanged
igor@ubuntu-server:~$ kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
database-0                          0/1     Pending   0           136m
node-exporter-vpzsp                 1/1     Running   0           22m
php-apache-hps-f7855487b-jt99x     1/1     Running   0           10s
prometheus-0                        0/1     Pending   0           137m
igor@ubuntu-server:~$ █
```

```
kubectl describe pod php-apache-hps-f7855487b-jt99x
```

```
igor@ubuntu-server: ~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
igor@ubuntu-server:~$ kubectl describe pod php-apache-hps-f7855487b-jt99x
Name:          php-apache-hps-f7855487b-jt99x
Namespace:     default
Priority:       0
Node:          minikube/192.168.49.2
Start Time:    Sun, 27 Mar 2022 20:47:57 +0000
Labels:        pod-template-hash=f7855487b
               run=php-apache-hps
Annotations:   <none>
Status:        Running
IP:            172.17.0.10
IPs:
  IP:          172.17.0.10
Controlled By: ReplicaSet/php-apache-hps-f7855487b
Containers:
  php-apache-hps:
    Container ID:  docker://9d07272851953b1a2c3279acb54502bbd61c12a0f5d0207e68fe453d8fd7f2e5
    Image:         k8s.gcr.io/hpa-example
    Image ID:      docker-pullable://k8s.gcr.io/hpa-example@sha256:581697a37f0e136db86d6b30392f0db40ce99c8248a7044c770012f4e8491544
    Port:         80/TCP
    Host Port:    0/TCP
    State:        Running
      Started:    Sun, 27 Mar 2022 20:47:59 +0000
    Ready:        True
    Restart Count: 0
    Limits:
      cpu: 500m
    Requests:
      cpu: 200m
    Environment:  <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-bfw47 (ro)
Conditions:
  Type            Status
  Initialized      True
  Ready            True
  ContainersReady  True
  PodScheduled     True
Volumes:
  kube-api-access-bfw47:
    Type:          Projected (a volume that contains injected data from multiple sources)
    TokenExpirationSeconds: 3607
    ConfigMapName:  kube-root-ca.crt
    ConfigMapOptional: <nil>
    DownwardAPI:    true
  QoS Class:       Burstable
  Node-Selectors:  <none>
  Tolerations:     node.kubernetes.io/not-ready:NoExecute op=Exists for 300s
                   node.kubernetes.io/unreachable:NoExecute op=Exists for 300s
Events:
  Type     Reason      Age   From          Message
  ----     -
  Normal   Scheduled   117s  default-scheduler  Successfully assigned default/php-apache-hps-f7855487b-jt99x to minikube
  Normal   Pulling     117s  kubelet        Pulling image "k8s.gcr.io/hpa-example"
  Normal   Pulled      116s  kubelet        Successfully pulled image "k8s.gcr.io/hpa-example" in 636.586134ms
  Normal   Created     116s  kubelet        Created container php-apache-hps
  Normal   Started     116s  kubelet        Started container php-apache-hps
igor@ubuntu-server:~$
```

kubectl get pods

kubectl get replicaset

kubectl get deployments

kubectl get daemonset

```
igor@ubuntu-server: ~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка

igor@ubuntu-server:~$ kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
database-0                          0/1     Pending   0           141m
node-exporter-vpzsp                  1/1     Running   0           27m
php-apache-hps-f7855487b-jt99x       1/1     Running   0           5m23s
prometheus-0                         0/1     Pending   0           142m
igor@ubuntu-server:~$ kubectl get replicaset
NAME                                DESIRED   CURRENT   READY   AGE
php-apache-hps-f7855487b            1         1         1       6m58s
igor@ubuntu-server:~$ kubectl get deployments
NAME                                READY   UP-TO-DATE   AVAILABLE   AGE
php-apache-hps                      1/1     1             1           7m39s
igor@ubuntu-server:~$ kubectl get daemonset
NAME                                DESIRED   CURRENT   READY   UP-TO-DATE   AVAILABLE   NODE SELECTOR   AGE
node-exporter                      1         1         1         1             1           beta.kubernetes.io/os=linux  30m
igor@ubuntu-server:~$
```

```
kubectl delete daemonset node-exporter
```

```
kubectl get pods
```

```
kubectl top pod
```

```
kubectl autoscale deployment php-apache-hps --cpu-percent=50 --min=1 --max=5
```

```
igor@ubuntu-server: ~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка

igor@ubuntu-server:~$ kubectl delete daemonset node-exporter
daemonset.apps "node-exporter" deleted
igor@ubuntu-server:~$ kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
database-0                          0/1     Pending   0           146m
php-apache-hps-f7855487b-jt99x       1/1     Running   0           10m
prometheus-0                         0/1     Pending   0           147m
igor@ubuntu-server:~$ kubectl top pod
error: Metrics API not available
igor@ubuntu-server:~$ kubectl autoscale deployment php-apache-hps --cpu-percent=50 --min=1 --max=5
horizontalpodautoscaler.autoscaling/php-apache-hps autoscaled
igor@ubuntu-server:~$
```

```
kubectl get hpa
```

```
kubectl get pods
```

```
igor@ubuntu-server: ~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка

igor@ubuntu-server:~$ kubectl get hpa
NAME                                REFERENCE            TARGETS      MINPODS   MAXPODS   REPLICAS   AGE
php-apache-hps                     Deployment/php-apache-hps  <unknown>/50%  1         5         1          68s
igor@ubuntu-server:~$ kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
database-0                          0/1     Pending   0           162m
php-apache-hps-f7855487b-jt99x       1/1     Running   0           26m
prometheus-0                         0/1     Pending   0           164m
igor@ubuntu-server:~$
```

```
kubectl run load-generator --image=busybox -- /bin/sh -c "while true; do wget -g -O- http://php-apache-hps; done"
```

```
kubectl get pods
```

kubectl top pod

```
igor@ubuntu-server: ~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
igor@ubuntu-server:~$ kubectl run load-generator --image=busybox -- /bin/sh -c "while true; do wget -g -O- http://php-apache-hps; done"
pod/load-generator created
igor@ubuntu-server:~$ kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
database-0                          0/1     Pending   0           175m
load-generator                      1/1     Running   0           62s
php-apache-hps-f7855487b-jt99x     1/1     Running   0           39m
prometheus-0                       0/1     Pending   0           176m
igor@ubuntu-server:~$ kubectl top pod
error: Metrics API not available
igor@ubuntu-server:~$
```

kubectl get hpa

kubectl get pods

```
igor@ubuntu-server: ~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
igor@ubuntu-server:~$ kubectl get hpa
NAME           REFERENCE                TARGETS   MINPODS   MAXPODS   REPLICAS   AGE
php-apache-hps Deployment/php-apache-hps <unknown>/50%  1         5         1          15m
igor@ubuntu-server:~$ kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
database-0                          0/1     Pending   0           177m
load-generator                      1/1     Running   0           3m22s
php-apache-hps-f7855487b-jt99x     1/1     Running   0           41m
prometheus-0                       0/1     Pending   0           178m
igor@ubuntu-server:~$
```

kubectl describe pod php-apache-hps-f7855487b-jt99x



```
igor@ubuntu-server: ~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
igor@ubuntu-server:~$ kubectl describe pod php-apache-hps-f7855487b-jt99x
Name:         php-apache-hps-f7855487b-jt99x
Namespace:    default
Priority:      0
Node:         minikube/192.168.49.2
Start Time:   Sun, 27 Mar 2022 20:47:57 +0000
Labels:       pod-template-hash=f7855487b
              run=php-apache-hps
Annotations:   <none>
Status:       Running
IP:           172.17.0.10
IPs:
  IP:         172.17.0.10
Controlled By: ReplicaSet/php-apache-hps-f7855487b
Containers:
  php-apache-hps:
    Container ID:  docker://9d07272851953b1a2c3279acb54502bbd61c12a0f5d0207e68fe453d8fd7f2e5
    Image:         k8s.gcr.io/hpa-example
    Image ID:      docker-pullable://k8s.gcr.io/hpa-example@sha256:581697a37f0e136db86d6b30392f0db40ce99c8248a7044c770012f4e8491544
    Port:         80/TCP
    Host Port:     0/TCP
    State:         Running
      Started:     Sun, 27 Mar 2022 20:47:59 +0000
    Ready:         True
    Restart Count: 0
    Limits:
      cpu: 500m
    Requests:
      cpu: 200m
    Environment:   <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-bfw47 (ro)
Conditions:
  Type            Status
  Initialized      True
  Ready           True
  ContainersReady  True
  PodScheduled     True
Volumes:
  kube-api-access-bfw47:
    Type:              Projected (a volume that contains injected data from multiple sources)
    TokenExpirationSeconds: 3607
    ConfigMapName:       kube-root-ca.crt
    ConfigMapOptional:   <nil>
    DownwardAPI:         true
  QoS Class:           Burstable
  Node-Selectors:      <none>
  Tolerations:         node.kubernetes.io/not-ready:NoExecute op=Exists for 300s
                      node.kubernetes.io/unreachable:NoExecute op=Exists for 300s
Events:
  Type     Reason      Age   From          Message
  ----     -
  Normal   Scheduled   43m   default-scheduler   Successfully assigned default/php-apache-hps-f7855487b-jt99x to minikube
  Normal   Pulling     43m   kubelet          Pulling image "k8s.gcr.io/hpa-example"
  Normal   Pulled      43m   kubelet          Successfully pulled image "k8s.gcr.io/hpa-example" in 636.586134ms
  Normal   Created     43m   kubelet          Created container php-apache-hps
  Normal   Started     43m   kubelet          Started container php-apache-hps
igor@ubuntu-server:~$
```

kubectl get pods

kubectl get hpa

kubectl delete pod load-generator

kubectl get hpa

```
igor@ubuntu-server: ~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка

igor@ubuntu-server:~$ kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
database-0                          0/1     Pending   0           3h2m
load-generator                       1/1     Running   0           8m25s
php-apache-hps-f7855487b-jt99x     1/1     Running   0           46m
prometheus-0                        0/1     Pending   0           3h3m
igor@ubuntu-server:~$ kubectl get hpa
NAME                                REFERENCE          TARGETS      MINPODS   MAXPODS   REPLICAS   AGE
php-apache-hps  Deployment/php-apache-hps  <unknown>/50%  1         5         1          21m
igor@ubuntu-server:~$ kubectl delete pod load-generator
pod "load-generator" deleted
igor@ubuntu-server:~$ kubectl get hpa
NAME                                REFERENCE          TARGETS      MINPODS   MAXPODS   REPLICAS   AGE
php-apache-hps  Deployment/php-apache-hps  <unknown>/50%  1         5         1          25m
igor@ubuntu-server:~$
```

kubectl top pod

error: Metrics API not available

kubectl get pod -w

kubectl get hpa

```
igor@ubuntu-server: ~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка

igor@ubuntu-server:~$ kubectl top pod
error: Metrics API not available
igor@ubuntu-server:~$ kubectl get pod -w
NAME                                READY   STATUS    RESTARTS   AGE
database-0                          0/1     Pending   0           3h11m
php-apache-hps-f7855487b-jt99x     1/1     Running   0           55m
prometheus-0                        0/1     Pending   0           3h12m
^Z
[1]+  Stopped                  kubectl get pod -w
igor@ubuntu-server:~$ kubectl get hpa
NAME                                REFERENCE          TARGETS      MINPODS   MAXPODS   REPLICAS   AGE
php-apache-hps  Deployment/php-apache-hps  <unknown>/50%  1         5         1          41m
igor@ubuntu-server:~$
```

vi hpa.yaml

80%