Урок 4 - Хранение данных и ресурсы

kubectl cluster-info

```
                                    Терминал - tolstikov@my3vm:~                              - + ×
 Файл   Правка   Вид   Терминал   Вкладки   Справка
 * Support:        https://ubuntu.com/advantage
Last login: Sat Nov  6 13:50:08 2021 from 109.252.20.121
tolstikov@my3vm:~$ kubectl cluster-info
Kubernetes control plane is running at https://192.168.49.2:8443
CoreDNS is running at https://192.168.49.2:8443/api/v1/namespaces/kube-system/services/kube-dns:dns/proxy

To further debug and diagnose cluster problems, use 'kubectl cluster-info dump'.
```
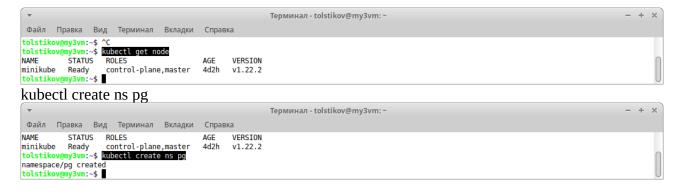
kubectl get node

```
                                    Терминал - tolstikov@my3vm:~                              - + ×
 Файл   Правка   Вид   Терминал   Вкладки   Справка
tolstikov@my3vm:~$ ^C
tolstikov@my3vm:~$ kubectl get node
NAME       STATUS   ROLES                  AGE    VERSION
minikube   Ready    control-plane,master   4d2h   v1.22.2
tolstikov@my3vm:~$
```

kubectl create ns pg

```
                                    Терминал - tolstikov@my3vm:~                              - + ×
 Файл   Правка   Вид   Терминал   Вкладки   Справка
NAME       STATUS   ROLES                  AGE    VERSION
minikube   Ready    control-plane,master   4d2h   v1.22.2
tolstikov@my3vm:~$ kubectl create ns pg
namespace/pg created
tolstikov@my3vm:~$
```

kubectl config set-context --current --namespace=pg

```
                                    Терминал - tolstikov@my3vm:~                              - + ×
 Файл   Правка   Вид   Терминал   Вкладки   Справка
tolstikov@my3vm:~$ kubectl create ns pg
namespace/pg created
tolstikov@my3vm:~$ kubectl config set-context --current --namespace=pg
Context "minikube" modified.
tolstikov@my3vm:~$
```

kubectl get storageclasses.storage.k8s.io

```
                                    Терминал - tolstikov@my3vm:~                              - + ×
 Файл   Правка   Вид   Терминал   Вкладки   Справка
Context "minikube" modified.
tolstikov@my3vm:~$ kubectl get storageclasses.storage.k8s.io
NAME               PROVISIONER             RECLAIMPOLICY   VOLUMEBINDINGMODE   ALLOWVOLUMEEXPANSION   AGE
standard (default) k8s.io/minikube-hostpath Delete         Immediate           false                  4d2h
tolstikov@my3vm:~$
```

vi pvc.yaml

```
                        Терминал - tolstikov@my3vm:~                    - + ×
 Файл   Правка   Вид   Терминал   Вкладки   Справка
---
kind: PersistentVolumeClaim
apiVersion: v1
metadata:
  name: pg-storage
  namespace: pg
spec:
  accessModes:
  - ReadWriteMany
  resources:
    requests:
      storage: 10Gi
  storageClassName: "standard"
~

:wq
```
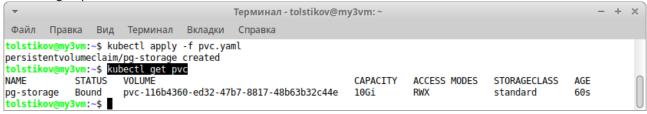
kubectl apply -f pvc.yaml

```
Терминал - tolstikov@my3vm: ~                    − + ×
Файл  Правка  Вид  Терминал  Вкладки  Справка

tolstikov@my3vm:~$ kubectl apply -f pvc.yaml
persistentvolumeclaim/pg-storage created
tolstikov@my3vm:~$
```

kubectl get pvc

```
Терминал - tolstikov@my3vm: ~                    − + ×
Файл  Правка  Вид  Терминал  Вкладки  Справка

tolstikov@my3vm:~$ kubectl apply -f pvc.yaml
persistentvolumeclaim/pg-storage created
tolstikov@my3vm:~$ kubectl get pvc
NAME         STATUS   VOLUME                                     CAPACITY   ACCESS MODES   STORAGECLASS   AGE
pg-storage   Bound    pvc-116b4360-ed32-47b7-8817-48b63b32c44e   10Gi       RWX            standard       60s
tolstikov@my3vm:~$
```

kubectl create secret generic pg-secret --from-literal=PASS=testpassword

```
Терминал - tolstikov@my3vm: ~                    − + ×
Файл  Правка  Вид  Терминал  Вкладки  Справка

tolstikov@my3vm:~$ kubectl get pvc
NAME         STATUS   VOLUME                                     CAPACITY   ACCESS MODES   STORAGECLASS   AGE
pg-storage   Bound    pvc-116b4360-ed32-47b7-8817-48b63b32c44e   10Gi       RWX            standard       60s
tolstikov@my3vm:~$ kubectl create secret generic pg-secret --from-literal=PASS=testpassword
secret/pg-secret created
tolstikov@my3vm:~$
```

kubectl get secret pg-secret

```
Терминал - tolstikov@my3vm: ~                    − + ×
Файл  Правка  Вид  Терминал  Вкладки  Справка

tolstikov@my3vm:~$ kubectl create secret generic pg-secret --from-literal=PASS=testpassword
secret/pg-secret created
tolstikov@my3vm:~$ kubectl get secret pg-secret
NAME        TYPE     DATA   AGE
pg-secret   Opaque   1      44s
tolstikov@my3vm:~$
```

kubectl get secret pg-secret -oyaml

```
Терминал - tolstikov@my3vm: ~                    − + ×
Файл  Правка  Вид  Терминал  Вкладки  Справка

tolstikov@my3vm:~$ kubectl get secret pg-secret -oyaml
apiVersion: v1
data:
  PASS: dGVzdHBhc3N3b3Jk
kind: Secret
metadata:
  creationTimestamp: "2021-11-06T22:20:50Z"
  name: pg-secret
  namespace: pg
  resourceVersion: "164964"
  uid: f50a3880-a331-4fa0-88ab-60278cf9fa91
type: Opaque
tolstikov@my3vm:~$
```

vi deployment.yaml

```yaml
---
apiVersion: apps/v1
kind: Deployment
metadata:
  name: pg-db
  namespace: pg
spec:
  replicas: 1
  selector:
    matchLabels:
      app: pg-db
  strategy:
    type: Recreate
  template:
    metadata:
      labels:
        app: pg-db
    spec:
      initContainers:
      - image: busybox
        name: mount-permissions-fix
        command: ["sh", "-c", "chmod 777 /var/lib/postgresql/data"]
        volumeMounts:
        - name: data
          mountPath: /var/lib/postgresql/data
      containers:
      - image: postgres:10.13
        name: postgres
        env:
        - name: POSTGRES_USER
          value: "testuser"
        - name: POSTGRES_DB
          value: "testdatabase"
        - name: PGDATA
          value: "/var/lib/postgresql/data/pgdata"
        - name: POSTGRES_PASSWORD
          valueFrom:
            secretKeyRef:
              name: pg-secret
              key: PASS
        ports:
        - containerPort: 5432
          protocol: TCP
        resources:
          requests:
            cpu: 100m
            memory: 100Mi
          limits:
            cpu: 100m
            memory: 100Mi
        volumeMounts:
        - name: data
          mountPath: /var/lib/postgresql/data
      volumes:
      - name: data
        persistentVolumeClaim:
          claimName: pg-storage
:wq
```

kubectl apply -f deployment.yaml

```
tolstikov@my3vm:~$ vi deployment.yaml
tolstikov@my3vm:~$ kubectl apply -f deployment.yaml
deployment.apps/pg-db created
tolstikov@my3vm:~$
```

kubectl get pod

```
tolstikov@my3vm:~$ kubectl get pod
NAME                   READY   STATUS    RESTARTS   AGE
pg-db-ff476d4cf-l46qf   1/1    Running   0          81s
tolstikov@my3vm:~$
```

kubectl describe pod pg-db-ff476d4cf-l46qf

```
                    sh
                    -c
                    chmod 777 /var/lib/postgresql/data
      State:        Terminated
        Reason:     Completed
        Exit Code:  0
        Started:    Sat, 06 Nov 2021 22:53:24 +0000
        Finished:   Sat, 06 Nov 2021 22:53:24 +0000
      Ready:        True
      Restart Count: 0
      Environment:  <none>
      Mounts:
        /var/lib/postgresql/data from data (rw)
        /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-2j8n9 (ro)
Containers:
  postgres:
      Container ID:  docker://2245399caf6c972df3585b7cf946dcc4fc73d6daff9c733ad1435469c083160d
      Image:         postgres:10.13
      Image ID:      docker-pullable://postgres@sha256:e3a02efdce3ec64cfdb76a8ff93ae14d3294e47a0203d8230c8853a3890fe34
0
      Port:          5432/TCP
      Host Port:     0/TCP
      State:         Running
        Started:     Sat, 06 Nov 2021 22:53:40 +0000
      Ready:         True
      Restart Count: 0
      Limits:
        cpu:      100m
        memory:   100Mi
      Requests:
        cpu:      100m
        memory:   100Mi
      Environment:
        POSTGRES_USER:      testuser
        POSTGRES_DB:        testdatabase
        PGDATA:             /var/lib/postgresql/data/pgdata
        POSTGRES_PASSWORD:  <set to the key 'PASS' in secret 'pg-secret'>  Optional: false
      Mounts:
        /var/lib/postgresql/data from data (rw)
        /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-2j8n9 (ro)
Conditions:
  Type              Status
  Initialized       True
  Ready             True
  ContainersReady   True
  PodScheduled      True
Volumes:
  data:
      Type:       PersistentVolumeClaim (a reference to a PersistentVolumeClaim in the same namespace)
      ClaimName:  pg-storage
      ReadOnly:   false
  kube-api-access-2j8n9:
      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  2m37s  default-scheduler  Successfully assigned pg/pg-db-ff476d4cf-l46qf to minikube
  Normal  Pulling    2m31s  kubelet            Pulling image "busybox"
  Normal  Pulled     2m25s  kubelet            Successfully pulled image "busybox" in 5.360549934s
  Normal  Created    2m25s  kubelet            Created container mount-permissions-fix
  Normal  Started    2m25s  kubelet            Started container mount-permissions-fix
  Normal  Pulling    2m25s  kubelet            Pulling image "postgres:10.13"
  Normal  Pulled     2m11s  kubelet            Successfully pulled image "postgres:10.13" in 13.570961304s
  Normal  Created    2m9s   kubelet            Created container postgres
  Normal  Started    2m9s   kubelet            Started container postgres
tolstikov@my3vm:~$
```

kubectl get pod -o wide

```
tolstikov@my3vm:~$ kubectl get pod -o wide
NAME                   READY   STATUS    RESTARTS   AGE    IP            NODE       NOMINATED NODE   READINESS GATES
pg-db-ff476d4cf-l46qf  1/1     Running   0          7m53s  172.17.0.12   minikube   <none>           <none>
tolstikov@my3vm:~$
```

kubectl port-forward pod/pg-db-ff476d4cf-l46qf 5433:5432


```
tolstikov@my3vm:~$ kubectl port-forward pod/pg-db-ff476d4cf-l46qf 5433:5432
Forwarding from 127.0.0.1:5433 -> 5432
Forwarding from [::1]:5433 -> 5432
```

kubectl run -t -i --rm --image postgres:10.13 test bash


```
igor@igor-MS-7808:~$ ssh tolstikov@62.84.121.110
Welcome to Ubuntu 20.04.3 LTS (GNU/Linux 5.4.0-89-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage
Last login: Sun Nov  7 16:56:03 2021 from 109.252.20.121
tolstikov@my3vm:~$ kubectl run -t -i --rm --image postgres:10.13 test bash
If you don't see a command prompt, try pressing enter.
root@test:/#
```

psql -h 172.17.0.12 -U testuser testdatabase


```
igor@igor-MS-7808:~$ ssh tolstikov@62.84.121.110
Welcome to Ubuntu 20.04.3 LTS (GNU/Linux 5.4.0-89-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage
Last login: Sun Nov  7 16:56:03 2021 from 109.252.20.121
tolstikov@my3vm:~$ kubectl run -t -i --rm --image postgres:10.13 test bash
If you don't see a command prompt, try pressing enter.
root@test:/# psql -h 172.17.0.12  -U testuser testdatabase
Password for user testuser:
```

пароль - testpassword


```
tolstikov@my3vm:~$ kubectl run -t -i --rm --image postgres:10.13 test bash
If you don't see a command prompt, try pressing enter.
root@test:/# psql -h 172.17.0.12  -U testuser testdatabase
Password for user testuser:
psql: fe_sendauth: no password supplied
root@test:/# psql -h 172.17.0.12  -U testuser testdatabase
Password for user testuser:
psql (10.13 (Debian 10.13-1.pgdg90+1))
Type "help" for help.

testdatabase=#
```

CREATE TABLE testtable (testcolumn VARCHAR (50));

```
root@test:/# psql -h 172.17.0.12  -U testuser testdatabase
Password for user testuser:
psql: fe_sendauth: no password supplied
root@test:/# psql -h 172.17.0.12  -U testuser testdatabase
Password for user testuser:
psql (10.13 (Debian 10.13-1.pgdg90+1))
Type "help" for help.

testdatabase=# CREATE TABLE testtable (testcolumn VARCHAR (50) );
CREATE TABLE
testdatabase=#
```

\dt

```
testdatabase=# CREATE TABLE testtable (testcolumn VARCHAR (50) );
CREATE TABLE
testdatabase=# \dt
          List of relations
 Schema |   Name    | Type  |  Owner
--------+-----------+-------+----------
 public | testtable | table | testuser
(1 row)

testdatabase=#
```

\q
exit

```
 Schema |   Name    | Type  |  Owner
--------+-----------+-------+----------
 public | testtable | table | testuser
(1 row)

testdatabase=# \q
root@test:/# exit
exit
Session ended, resume using 'kubectl attach test -c test -i -t' command when the pod is running
pod "test" deleted
tolstikov@my3vm:~$
```

kubectl delete pod pg-db-ff476d4cf-l46qf

```
 public | testtable | table | testuser
(1 row)

testdatabase=# \q
root@test:/# exit
exit
Session ended, resume using 'kubectl attach test -c test -i -t' command when the pod is running
pod "test" deleted
tolstikov@my3vm:~$ kubectl delete pod pg-db-ff476d4cf-l46qf
pod "pg-db-ff476d4cf-l46qf" deleted
tolstikov@my3vm:~$
```

kubectl get pod

```
testdatabase=# \q
root@test:/# exit
exit
Session ended, resume using 'kubectl attach test -c test -i -t' command when the pod is running
pod "test" deleted
tolstikov@my3vm:~$ kubectl delete pod pg-db-ff476d4cf-l46qf
pod "pg-db-ff476d4cf-l46qf" deleted
tolstikov@my3vm:~$ kubectl get pod
NAME                    READY   STATUS    RESTARTS   AGE
pg-db-ff476d4cf-xkxhn   1/1     Running   0          77s
tolstikov@my3vm:~$
```

kubectl get pod -o wide


```
Session ended, resume using 'kubectl attach test -c test -i -t' command when the pod is running
pod "test" deleted
tolstikov@my3vm:~$ kubectl delete pod pg-db-ff476d4cf-l46qf
pod "pg-db-ff476d4cf-l46qf" deleted
tolstikov@my3vm:~$ kubectl get pod
NAME                    READY   STATUS    RESTARTS   AGE
pg-db-ff476d4cf-xkxhn   1/1     Running   0          77s
tolstikov@my3vm:~$ kubectl get pod -o wide
NAME                    READY   STATUS    RESTARTS   AGE     IP           NODE       NOMINATED NODE   READINESS GATES
pg-db-ff476d4cf-xkxhn   1/1     Running   0          2m20s   172.17.0.12  minikube   <none>           <none>
tolstikov@my3vm:~$
```

kubectl run -t -i --rm --image postgres:10.13 test bash


```
Welcome to Ubuntu 20.04.3 LTS (GNU/Linux 5.4.0-89-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage
Last login: Sun Nov  7 17:39:26 2021 from 109.252.20.121
tolstikov@my3vm:~$ kubectl run -t -i --rm --image postgres:10.13 test bash
If you don't see a command prompt, try pressing enter.
root@test:/#
```

psql -h 172.17.0.12 -U testuser testdatabase


```
 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage
Last login: Sun Nov  7 17:39:26 2021 from 109.252.20.121
tolstikov@my3vm:~$ kubectl run -t -i --rm --image postgres:10.13 test bash
If you don't see a command prompt, try pressing enter.
root@test:/# psql -h 172.17.0.12  -U testuser testdatabase
Password for user testuser:
```

пароль — testpassword


```
Last login: Sun Nov  7 17:39:26 2021 from 109.252.20.121
tolstikov@my3vm:~$ kubectl run -t -i --rm --image postgres:10.13 test bash
If you don't see a command prompt, try pressing enter.
root@test:/# psql -h 172.17.0.12  -U testuser testdatabase
Password for user testuser:
psql (10.13 (Debian 10.13-1.pgdg90+1))
Type "help" for help.

testdatabase=#
```

\dt


```
testdatabase=# \dt
           List of relations
 Schema |    Name    | Type  |  Owner
--------+------------+-------+----------
 public | testtable  | table | testuser
(1 row)

testdatabase=# ▮
```

\q
exit


```
 public | testtable  | table | testuser
(1 row)

testdatabase=# \q
root@test:/# exit
exit
Session ended, resume using 'kubectl attach test -c test -i -t' command when the pod is running
pod "test" deleted
tolstikov@my3vm:~$ ▮
```