

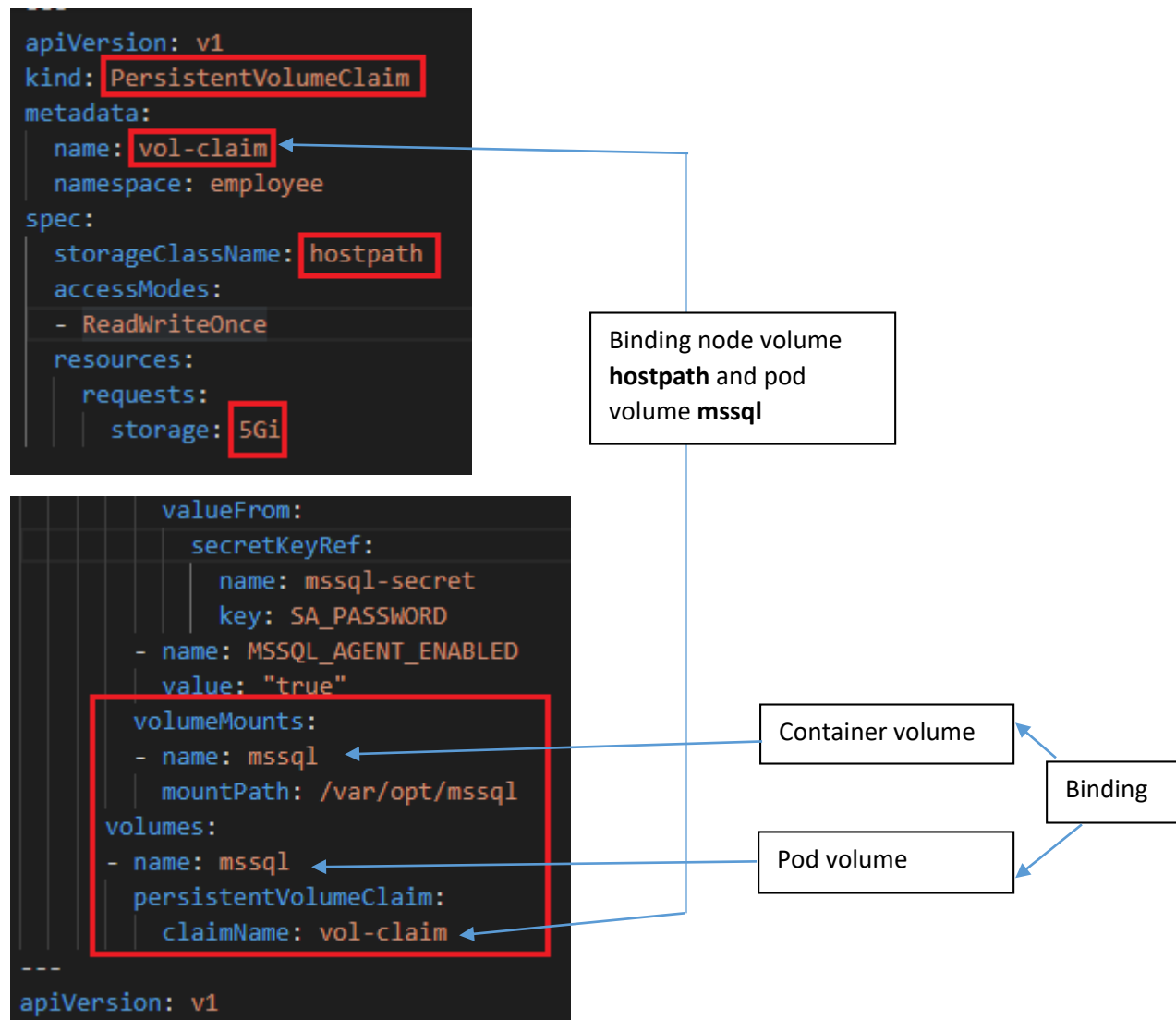
## Lesson5 - Create persistent volume

In this lesson we will deploy MSSQL with persistent volume for high availability.

**kubectl get storageclass**

```
PS C:\Kubernetes\EmployeesManagement\Employees> kubectl get storageclass
NAME             PROVISIONER      RECLAIMPOLICY   VOLUMEBINDINGMODE   ALLOWVOLUMEEXPANSION   AGE
hostpath (default)  docker.io/hostpath  Delete          Immediate            false                  5d4h
PS C:\Kubernetes\EmployeesManagement\Employees>
```

We added to **mssql-deploy-with-secret-and-PV.yml** the section below:



**kubectl create namespace employee**

```
kubectl create secret generic mssql-secret --namespace=employee --from-
literal='ConnectionString="server=mssql-service;Initial Catalog=EmployeeDB;Persist Security
Info=False;User ID=sa;Password=MyDemoPwd2021!;MultipleActiveResultSets=true"' --from-
literal='SA_PASSWORD=MyDemoPwd2021!'
```

```
kubectl apply -f .\mssql-deploy-with-secret-and-PV.yml
```

```
PS C:\Kubernetes\EmployeesManagement\manifests> kubectl apply -f .\mssql-deploy-with-secret-and-PV.yml
deployment.apps/mssql-deployment created
service/mssql-service created
persistentvolumeclaim/vol-claim created
PS C:\Kubernetes\EmployeesManagement\manifests>
```

```
kubectl get pv -n employee
```

```
PS C:\Kubernetes\EmployeesManagement\manifests> kubectl get pv -n employee
```

NAME	CAPACITY	ACCESS MODES	RECLAIM POLICY	STATUS	CLAIM	STORAGECLASS	REASON	AGE
pvc-1a247447-48eb-46c3-ad7f-add854c1151f	5Gi	RWO	Delete	Bound	employee/vol-claim	hostpath		118s

```
PS C:\Kubernetes\EmployeesManagement\manifests>
```

```
kubectl get pvc -n employee
```

```
PS C:\Kubernetes\EmployeesManagement\manifests> kubectl get pvc -n employee
```

NAME	STATUS	VOLUME	CAPACITY	ACCESS MODES	STORAGECLASS	AGE
vol-claim	Bound	pvc-1a247447-48eb-46c3-ad7f-add854c1151f	5Gi	RWO	hostpath	3m3s

```
PS C:\Kubernetes\EmployeesManagement\manifests>
```

```
kubectl get all -n employee
```

```
PS C:\Kubernetes\EmployeesManagement\manifests> kubectl get all -n employee
```

NAME	READY	STATUS	RESTARTS	AGE
pod/mssql-deployment-6bcb97764c-2tpl4	1/1	Running	0	4m18s

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
service/mssql-service	LoadBalancer	10.102.148.234	localhost	1433:30123/TCP	4m18s

NAME	READY	UP-TO-DATE	AVAILABLE	AGE
deployment.apps/mssql-deployment	1/1	1	1	4m18s

NAME	DESIRED	CURRENT	READY	AGE
replicaset.apps/mssql-deployment-6bcb97764c	1	1	1	4m18s

```
PS C:\Kubernetes\EmployeesManagement\manifests>
```

```
cd ..
```

```
cd Employees
```

**dotnet ef database update** (creates **EmployeeDB** database in our MSSQL server with initial data.)

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL
PS C:\employee\employeemanagement\manifests> cd ..
PS C:\employee\employeemanagement> cd .\Employees\
PS C:\employee\employeemanagement\Employees> dotnet ef database update
Build started...
Build succeeded.
Done.
PS C:\employee\employeemanagement\Employees> |
```

**dotnet build & dotnet run**

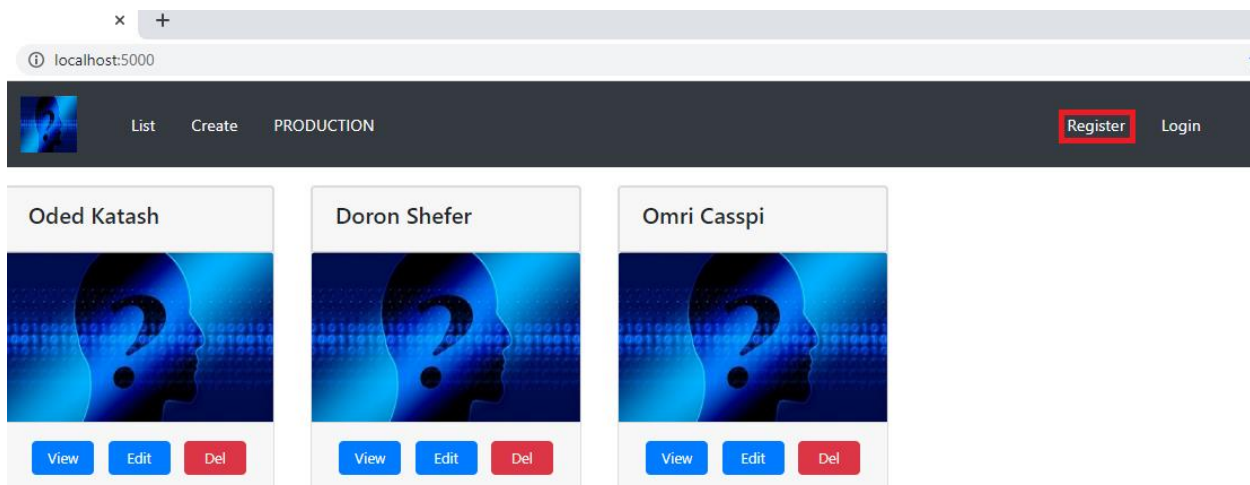
```
Try the new cross-platform PowerShell https://aka.ms/pscore6
PS C:\employee> cd .\employeemanagement\
PS C:\employee\employeemanagement> cd .\Employees\
PS C:\employee\employeemanagement\Employees> dotnet build
Microsoft (R) Build Engine version 16.7.2+b60ddb6f4 for .NET
Copyright (C) Microsoft Corporation. All rights reserved.

Determining projects to restore...
Restored C:\employee\employeemanagement\Employees\Employees.csproj (in 1.56 sec).
Employees -> C:\employee\employeemanagement\Employees\bin\Debug\netcoreapp3.1\Employees.dll
Employees -> C:\employee\employeemanagement\Employees\bin\Debug\netcoreapp3.1\Employees.Views.dll

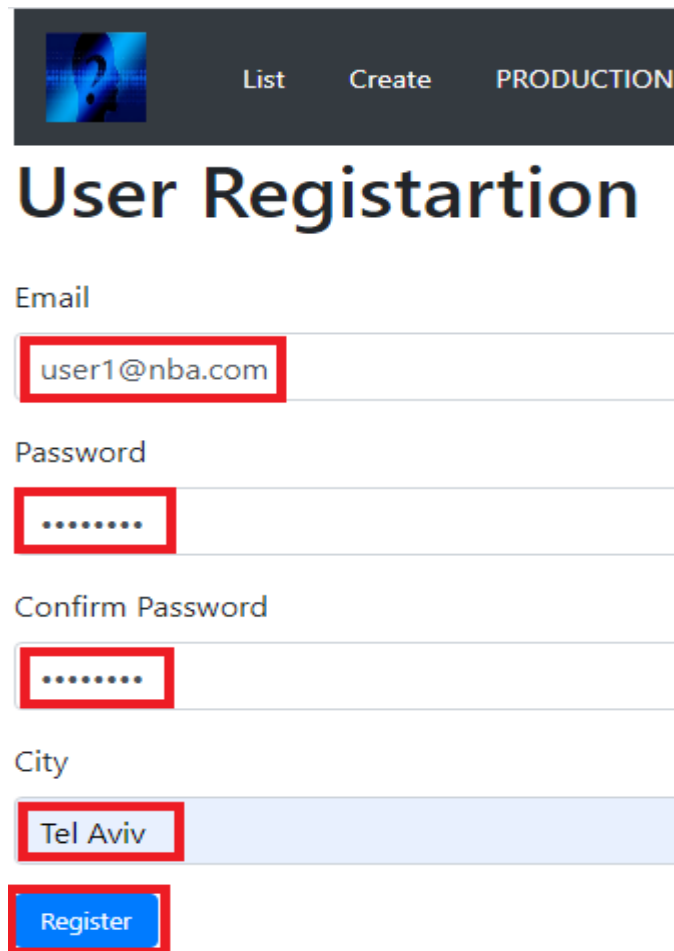
Build succeeded.
    0 Warning(s)
    0 Error(s)

Time Elapsed 00:00:30.76
PS C:\employee\employeemanagement\Employees> dotnet run
|
```

Open chrome – <http://localhost:5000> and click **Register**.

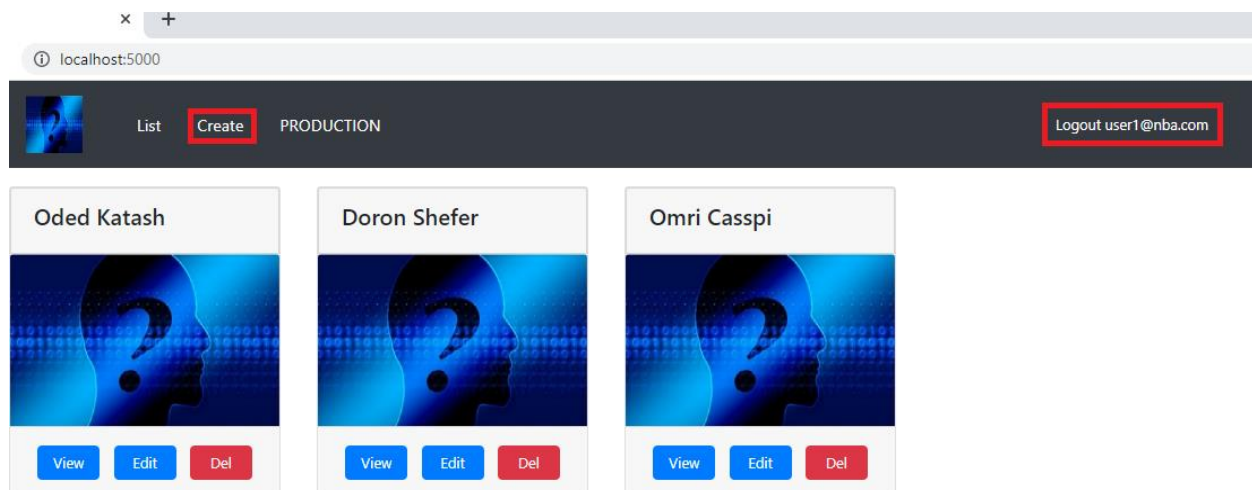


Enter the user details in the screen below and click **Register**.



The form is titled "User Registration" and is part of a web application. At the top, there is a navigation bar with a logo (a blue silhouette of a head with a question mark) and three links: "List", "Create", and "PRODUCTION". The form fields are: "Email" with the value "user1@nba.com", "Password" with masked characters ".....", "Confirm Password" with masked characters ".....", and "City" with the value "Tel Aviv". A blue "Register" button is at the bottom.

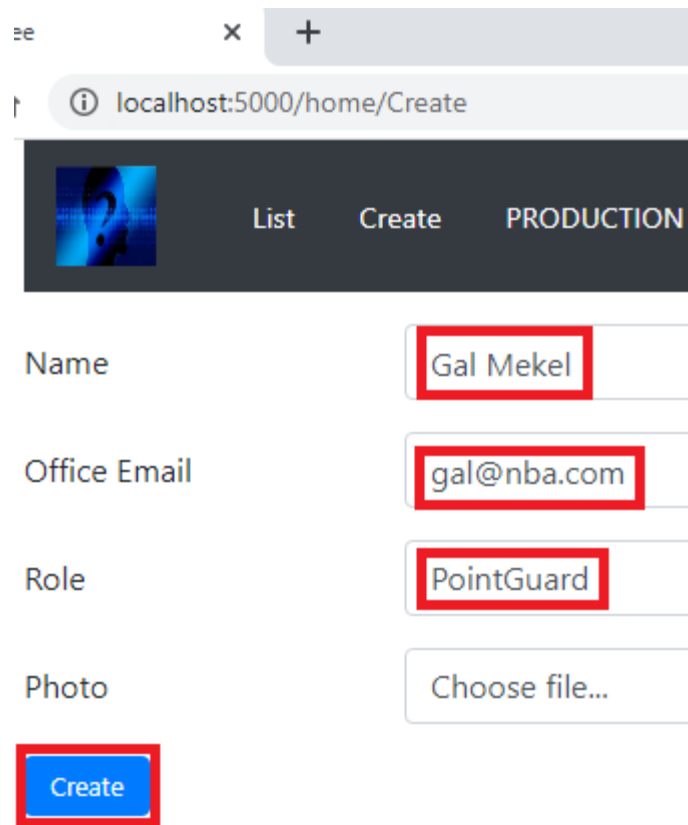
The user is logged in. Click **Create** in order to create new employee.



The dashboard shows a list of users. At the top, there is a navigation bar with the same logo and links as the registration form, but the "Create" link is highlighted. A "Logout user1@nba.com" button is in the top right. Below the navigation bar, there are three user cards. Each card has a header with the user's name, a profile picture (the same blue silhouette logo), and three buttons: "View", "Edit", and "Del".

Name	View	Edit	Del
Oded Katash	View	Edit	Del
Doron Shefer	View	Edit	Del
Omri Casspi	View	Edit	Del

Enter the employee details and click **Create**.

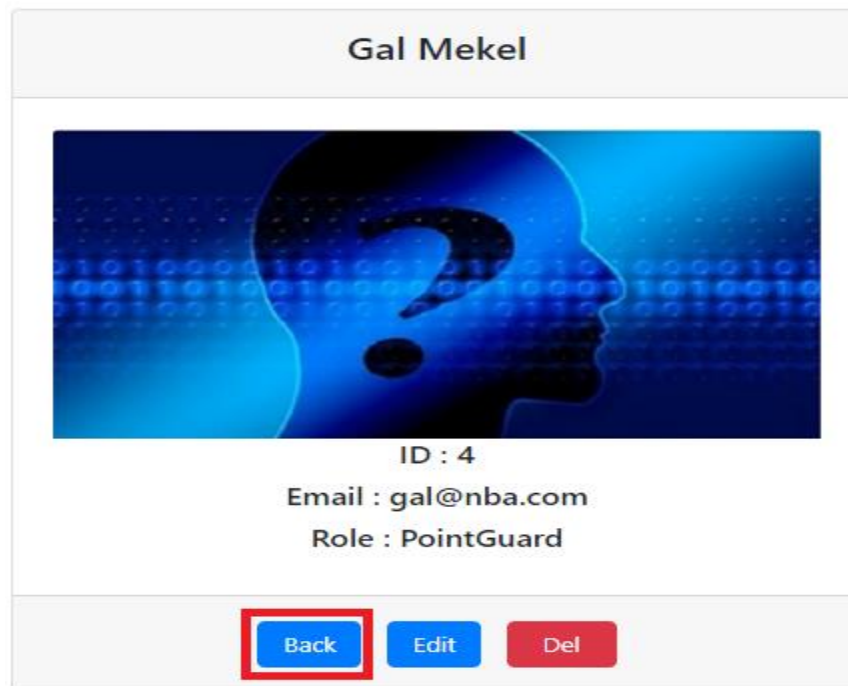


The screenshot shows a web browser window with the address bar displaying 'localhost:5000/home/Create'. The page has a dark header with a logo and navigation links: 'List', 'Create', and 'PRODUCTION'. The main content area contains a form with the following fields:

- Name: Gal Mekel
- Office Email: gal@nba.com
- Role: PointGuard
- Photo: Choose file...

A blue 'Create' button is located at the bottom of the form, highlighted with a red box.

Click **Back**.

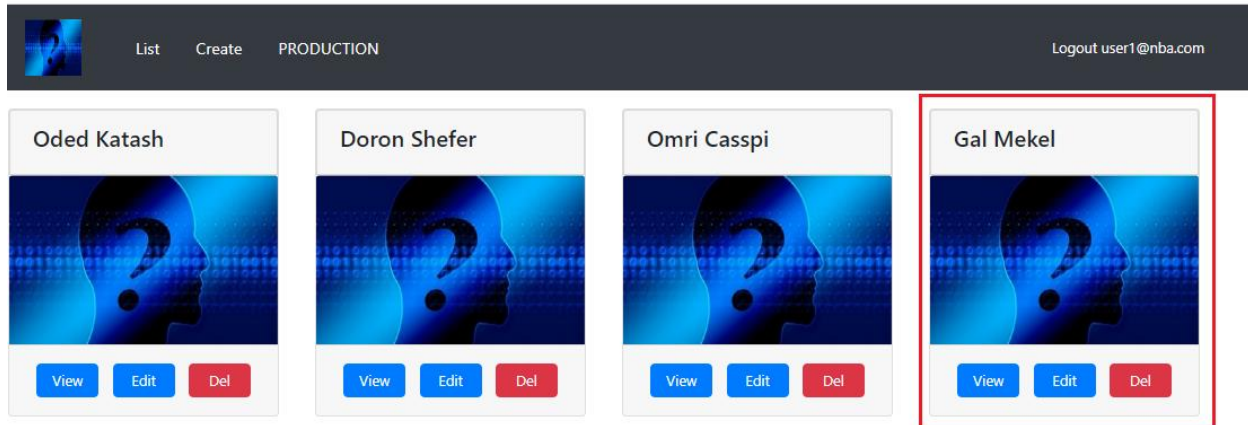


The screenshot shows a web application displaying the details of an employee named Gal Mekel. The page has a light gray header with the name 'Gal Mekel'. Below the header is a large placeholder image of a blue silhouette of a head with a question mark. The details are listed below the image:

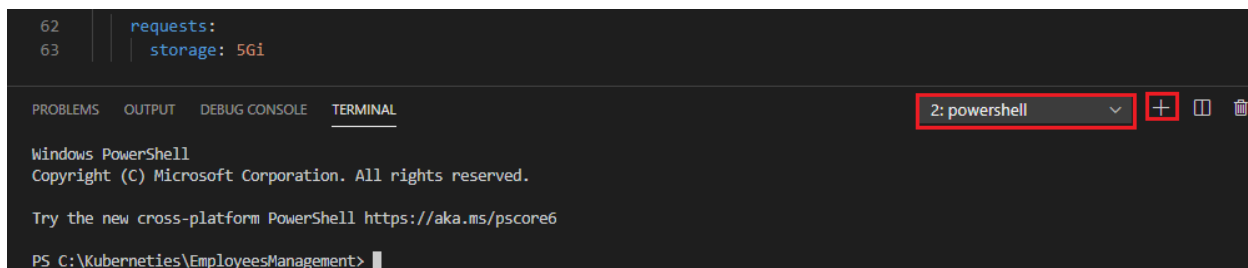
- ID : 4
- Email : gal@nba.com
- Role : PointGuard

At the bottom of the page, there are three buttons: 'Back' (blue), 'Edit' (blue), and 'Del' (red). The 'Back' button is highlighted with a red box.

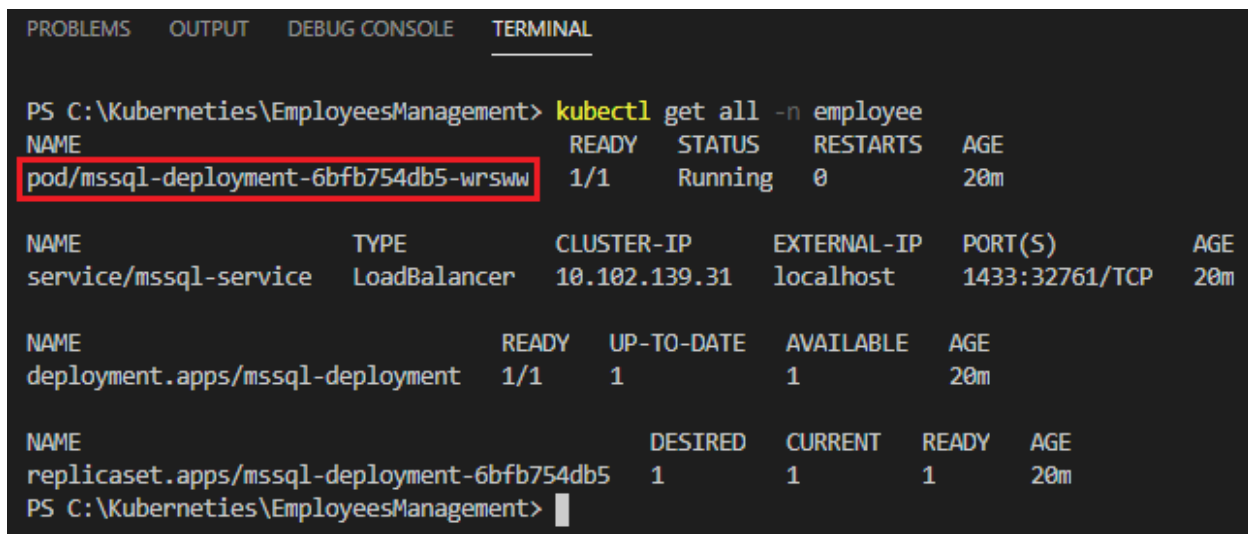
The user is created.



Click “+” in the terminal in order to open another PowerShell command prompt



**kubectl get all -n employee**



**kubectl delete pod/mssql-deployment-6bfb754db5-wrsww -n employee**

```
PS C:\Kubernetes\EmployeesManagement> kubectl delete pod/mssql-deployment-6bfb754db5-wrsww -n employee
pod "mssql-deployment-6bfb754db5-wrsww" deleted
PS C:\Kubernetes\EmployeesManagement>
```

**kubectl get all -n employee** (a new mssql pod is created based on the required state of the deployment replicaset)


```
PS C:\Kubernetes\EmployeesManagement> kubectl get all -n employee
NAME                                READY   STATUS    RESTARTS   AGE
pod/mssql-deployment-6bcb97764c-dfphx 1/1     Running   0           6m7s


NAME                                TYPE          CLUSTER-IP    EXTERNAL-IP  PORT(S)          AGE
service/mssql-service               LoadBalancer  10.102.148.234 localhost     1433:30123/TCP   4d19h

NAME                                READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/mssql-deployment    1/1     1             1           4d19h


NAME                                DESIRED   CURRENT   READY   AGE
replicaset.apps/mssql-deployment-6bcb97764c 1         1         1       4d19h
PS C:\Kubernetes\EmployeesManagement>
```

Refresh the page and see that the database and its data remains.



List   Create   PRODUCTION
Logout user1@nba.com

Oded Katash



View Edit Del

Doron Shefer


View Edit Del

Omri Casspi


View Edit Del

Gal Mekel


View Edit Del

Connect to mssql pod database file:

```
kubectl -n employee exec -it mssql-deployment-6bcb97764c-dfphx -- /bin/sh
```

```
ls -ltr /var/opt/mssql/data
```

```
PS C:\Kubernetes\EmployeesManagement> kubectl -n employee exec -it mssql-deployment-6bcb97764c-dfphx -- /bin/sh
# ls -ltr /var/opt/mssql/data
total 71176
-rw-r----- 1 root root 4194304 Feb  2 10:29 master.mdf
-rw-r----- 1 root root 8388608 Feb  2 10:30 modellog.ldf
-rw-r----- 1 root root 8388608 Feb  2 10:30 model.mdf
-rw-r----- 1 root root 8388608 Feb  2 10:30 tempdb.mdf
-rw-r----- 1 root root 8388608 Feb  2 10:30 EmployeeDB.mdf
-rw-r----- 1 root root 15466496 Feb  2 10:30 msdbdata.mdf
-rw-r----- 1 root root 786432 Feb  2 10:30 msdblog.ldf
-rw-r----- 1 root root 8388608 Feb  2 10:30 templog.ldf
-rw-r----- 1 root root 2097152 Feb  2 10:35 mastlog.ldf
-rw-r----- 1 root root 8388608 Feb  2 10:37 EmployeeDB_log.ldf
#
```

Exit

Connect to docker-desktop persistent volume with database files that bound to mssql pod:

```
docker run -it --rm --privileged --pid=host justincormack/nsenter1
```

```
cd /var/lib/k8s-pvs/vol-claim
```

```
ls (see the name of the persistent volume pvc-1a247447-48eb-46c3-ad7f-add854c1151f)
```

```
cd pvc-1a247447-48eb-46c3-ad7f-add854c1151f
```

```
ls -ltr (see the database files are the same as in mssql pod)
```

```
PS C:\Kubernetes\EmployeesManagement> docker run -it --rm --privileged --pid=host justincormack/nsenter1
/ # cd /var/lib/k8s-pvs/vol-claim
/var/lib/k8s-pvs/vol-claim # ls
pvc-1a247447-48eb-46c3-ad7f-add854c1151f
/var/lib/k8s-pvs/vol-claim # cd pvc-1a247447-48eb-46c3-ad7f-add854c1151f/
/var/lib/k8s-pvs/vol-claim/pvc-1a247447-48eb-46c3-ad7f-add854c1151f # ls -ltr
total 12
drwxr-xr-x  2 root  root    4096 Jan 28 14:50 secrets
drwxr-xr-x  2 root  root    4096 Feb  2 10:22 data
drwxr-xr-x  2 root  root    4096 Feb  2 10:43 log
/var/lib/k8s-pvs/vol-claim/pvc-1a247447-48eb-46c3-ad7f-add854c1151f # cd data
/var/lib/k8s-pvs/vol-claim/pvc-1a247447-48eb-46c3-ad7f-add854c1151f/data # ls -ltr
total 71176
-rw-r----- 1 root  root    4194304 Feb  2 10:29 master.mdf
-rw-r----- 1 root  root    8388608 Feb  2 10:30 modellog.ldf
-rw-r----- 1 root  root    8388608 Feb  2 10:30 model.mdf
-rw-r----- 1 root  root    8388608 Feb  2 10:30 tempdb.mdf
-rw-r----- 1 root  root    8388608 Feb  2 10:30 EmployeeDB.mdf
-rw-r----- 1 root  root   15466496 Feb  2 10:30 msdbdata.mdf
-rw-r----- 1 root  root    786432 Feb  2 10:30 msdblog.ldf
-rw-r----- 1 root  root    8388608 Feb  2 10:30 templog.ldf
-rw-r----- 1 root  root    2097152 Feb  2 10:37 mastlog.ldf
-rw-r----- 1 root  root    8388608 Feb  2 10:43 EmployeeDB_log.ldf
/var/lib/k8s-pvs/vol-claim/pvc-1a247447-48eb-46c3-ad7f-add854c1151f/data #
```

Done!