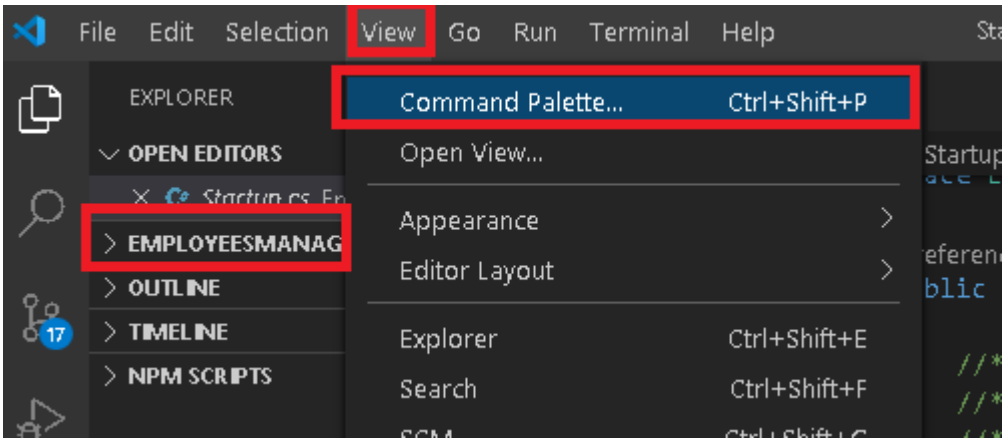
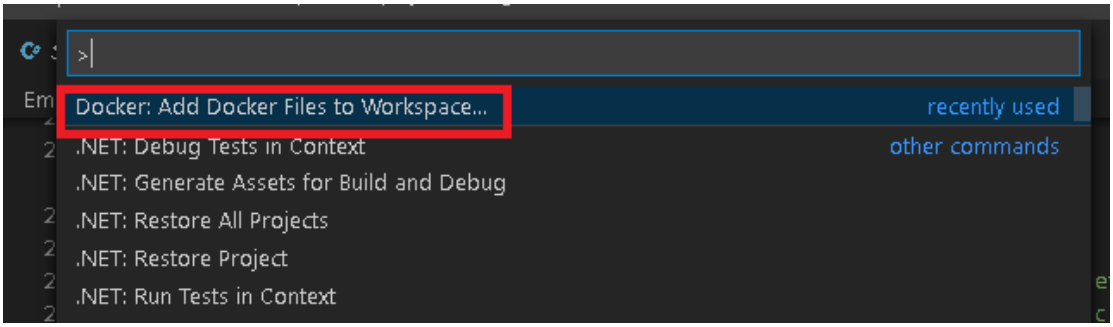
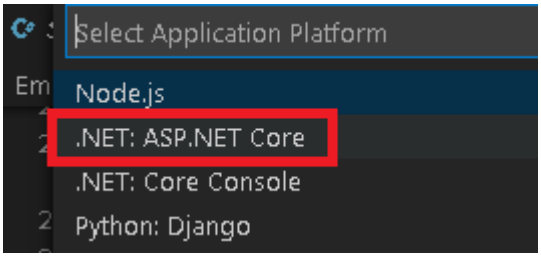
Lesson6 - Create docker image and push to DockerHub

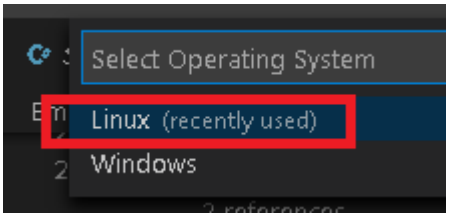
In the previous lesson we deployed MSSQL to Kuberneties by using a public docker image that was published by Microsoft and was ready for deployment. In this lesson we will create docker image for the .NET core web application.

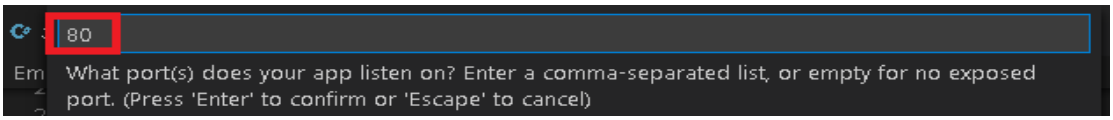
Create Dockerfile:

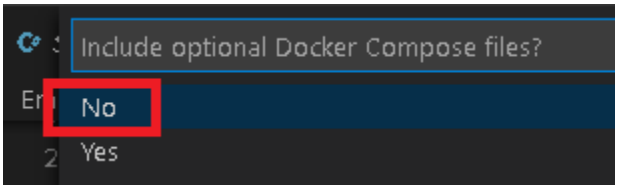












Click ENTER > Dockerfile is created.

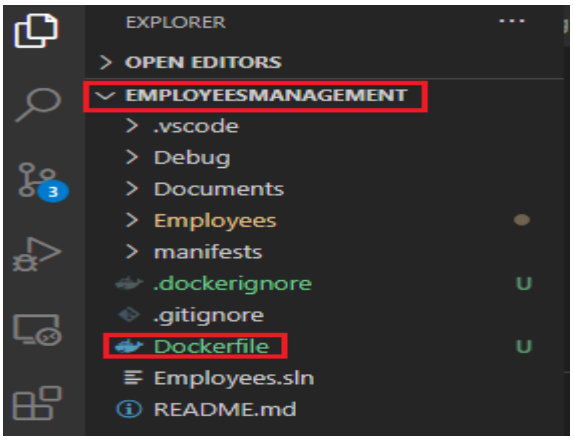
```
FROM mcr.microsoft.com/dotnet/core/aspnet:3.1 AS base
WORKDIR /app
EXPOSE 80

FROM mcr.microsoft.com/dotnet/core/sdk:3.1 AS build
WORKDIR /src
COPY ["Employees/Employees.csproj", "Employees/"]
RUN dotnet restore "Employees/Employees.csproj"
COPY . .
WORKDIR "/src/Employees"
RUN dotnet build "Employees.csproj" -c Release -o /app/build

FROM build AS publish
RUN dotnet publish "Employees.csproj" -c Release -o /app/publish

FROM base AS final
WORKDIR /app
COPY --from=publish /app/publish .
ENTRYPOINT ["dotnet", "Employees.dll"]
```

Move the Dockerfile to EmployeeManagement folder.



Create Docker image with the command below:

docker build -t employees:v1 .

```
PS C:\Kuberneties\EmployeesManagement> docker build -t employees:v1 .
[+] Building 256.1s (18/18) FINISHED
 => [internal] load build definition from Dockerfile
 => => transferring dockerfile: 579B
 => [internal] load .dockerignore
 => => transferring context: 375B
 => [internal] load metadata for mcr.microsoft.com/dotnet/aspnet:3.1
 => [internal] load metadata for mcr.microsoft.com/dotnet/sdk:3.1
 => [build 1/7] FROM mcr.microsoft.com/dotnet/sdk:3.1@sha256:f7bc4a555fef18101b64bc7b954738489349d53655731ef8254f4d4619a9230c
 => => resolve mcr.microsoft.com/dotnet/sdk:3.1@sha256:f7bc4a555fef18101b64bc7b954738489349d53655731ef8254f4d4619a9230c
 => => sha256:f7bc4a555fef18101b64bc7b954738489349d53655731ef8254f4d4619a9230c 2.53kB / 2.53kB
 => => sha256:0fece15a102530aa2dad9d247bc0d05db6790917696377fc56a8465604ef1aff 1.80kB / 1.80kB
 => => sha256:52a3845cafb11eac25cf49744ad90e4e22086cef203de8c815c077af179bf987 6.33kB / 6.33kB
 => => sha256:b9a857cbf04d2c0d2f0f6b73e894b20a977a6d3b6edd9e27d080e03142954950 50.40MB / 50.40MB
 => => sha256:d557ee20540b597f518df05bc6888778cfc92bf46040c701d4a622389feb6807 7.81MB / 7.81MB
 => => sha256:3b9ca4f00c2e4896c65625d678544b764d7483dca9dcab92b62093db72f21d3e 10.00MB / 10.00MB
```

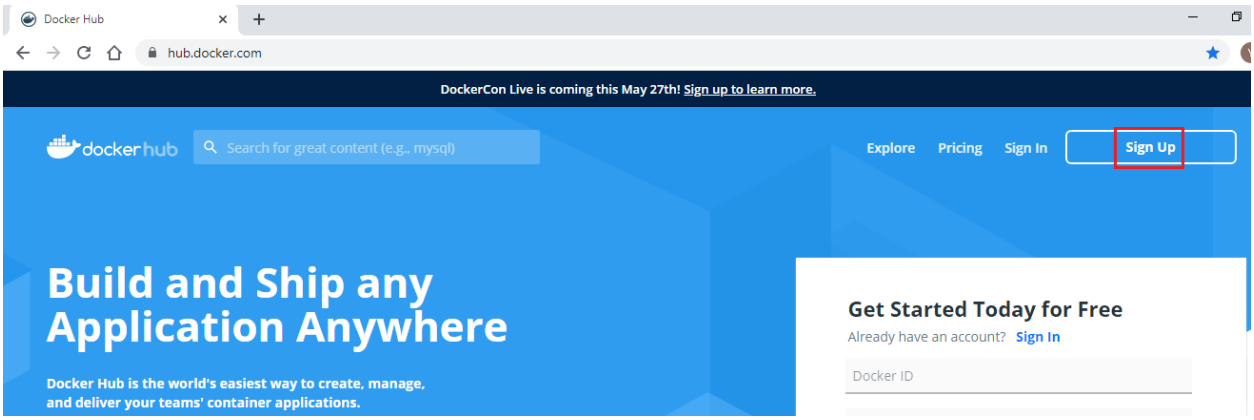
See the docker image is created3 with tag v1:

docker images

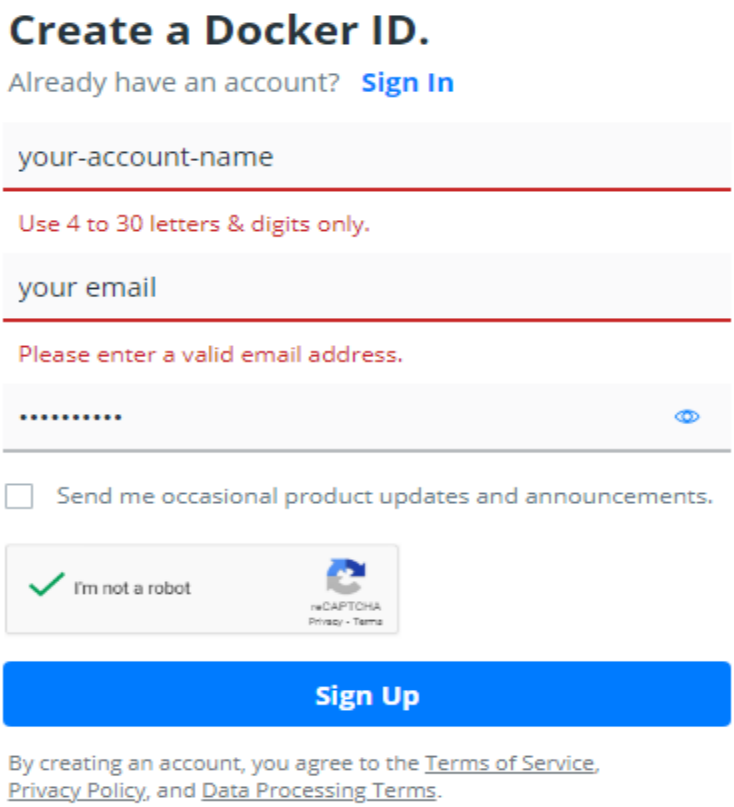
```
PS C:\Kuberneties\EmployeesManagement> docker images
REPOSITORY                    TAG                                        IMAGE ID       CREATED          SIZE
employees                     v1                                         3f4f232dabd6   3 minutes ago    223MB
docker/desktop-kubernetes     kubernetes-v1.19.3-cni-v0.8.5-critools-v1.17.0-debian   bb2106081d17   2 months ago     285MB
docker/desktop-kubernetes     kubernetes-v1.19.3-cni-v0.8.5-critools-v1.17.0          7f85afe431d8   3 months ago     285MB
k8s.gcr.io/kube-proxy         v1.19.3                                    cdef7632a242   3 months ago     118MB
k8s.gcr.io/kube-controller-manager   v1.19.3                             9b60aca1d818   3 months ago     111MB
k8s.gcr.io/kube-apiserver     v1.19.3                                    a301be0cd44b   3 months ago     119MB
```

Push docker image to DockerHub:

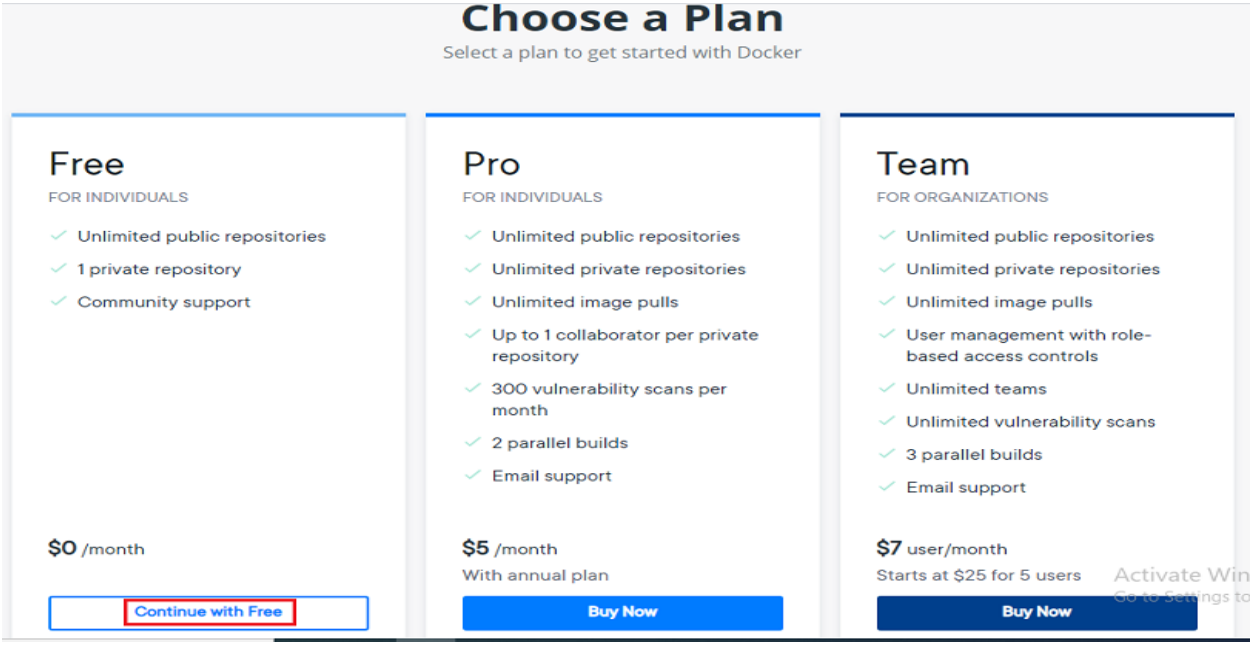
Open https://hub.docker.com/ and click Sign Up.

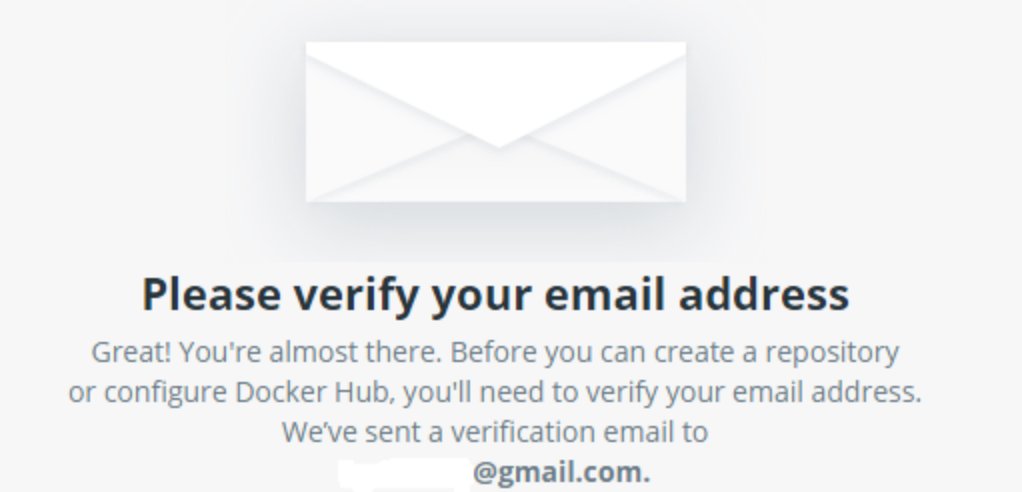


Enter Username, Email and password and click Sign Up:

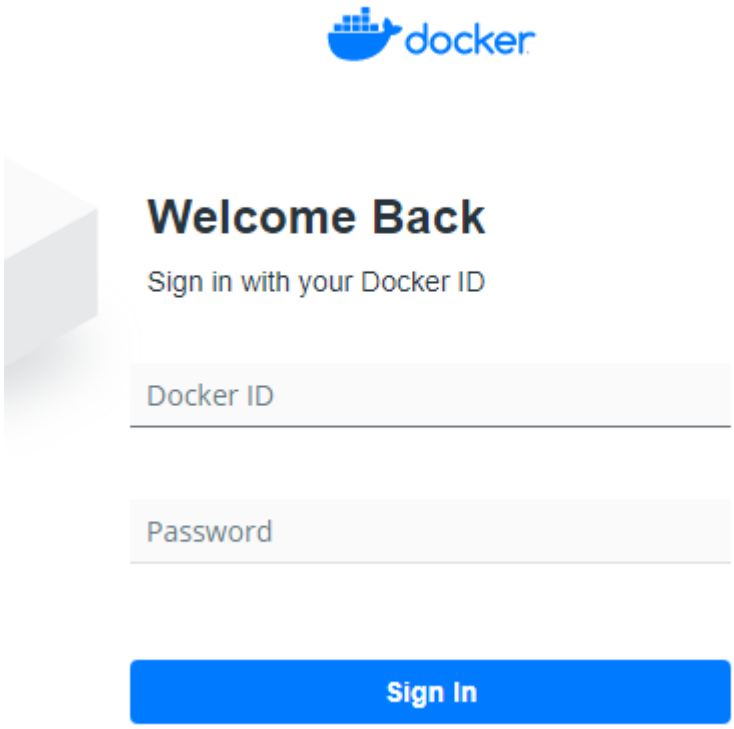


Select Free plan:

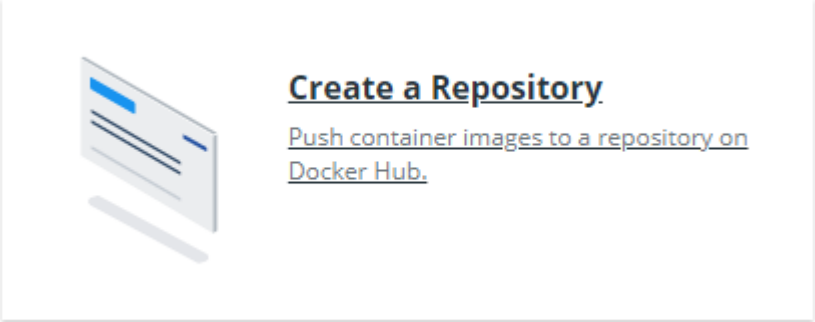



Please verify your email address
Great! You're almost there. Before you can create a repository or configure Docker Hub, you'll need to verify your email address. We've sent a verification email to
@gmail.com.

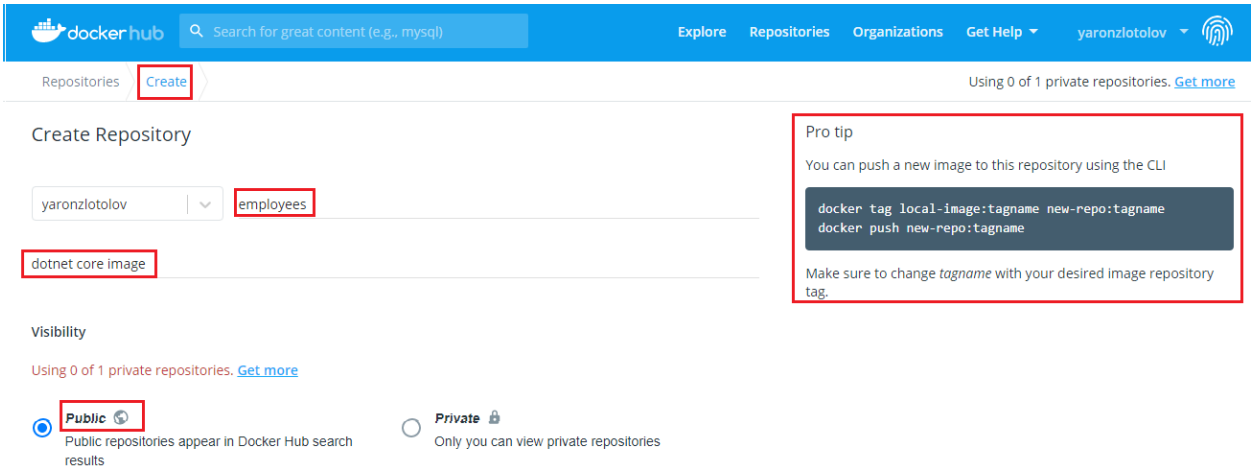
Sign in:


docker
Welcome Back
Sign in with your Docker ID
Docker ID
Password
Sign In

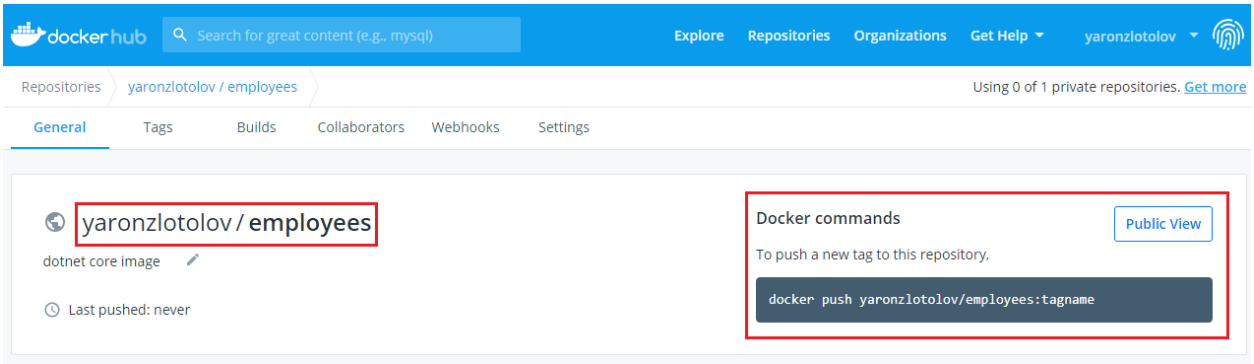
Click Create a Repository:



Write the repository name, description, choose Public and click Create in the bottom of the page:



Repository for images is created.



This command will be used to push the docker images from PC to DockerHub:

```
docker push yaronzlotolov/employees:tagname
```

Push employees image to DockerHub:

The command below creates a tag for yaronzlotolov/employees repository before pushing to DockerHub:

docker tag employees:v1 yaronzlotolov/employees:v1

docker images | more

```
PS C:\Kuberneties\EmployeesManagement> docker tag employees:v1 yaronzlotolov/employees:v1
PS C:\Kuberneties\EmployeesManagement> docker images | more
REPOSITORY                          TAG                                                IMAGE ID
yaronzlotolov/employees             v1                                                 3f4f232dabd6
employees                           v1                                                 3f4f232dabd6
docker/desktop-kubernetes           kubernetes-v1.19.3-cni-v0.8.5-critools-v1.17.0-debian   bb2106081d17
docker/desktop-kubernetes           kubernetes-v1.19.3-cni-v0.8.5-critools-v1.17.0      7f85afe431d8
k8s.gcr.io/kube-proxy               v1.19.3                                            cdef7632a242
```

In order to push images to DockerHub it is required to login:

docker login

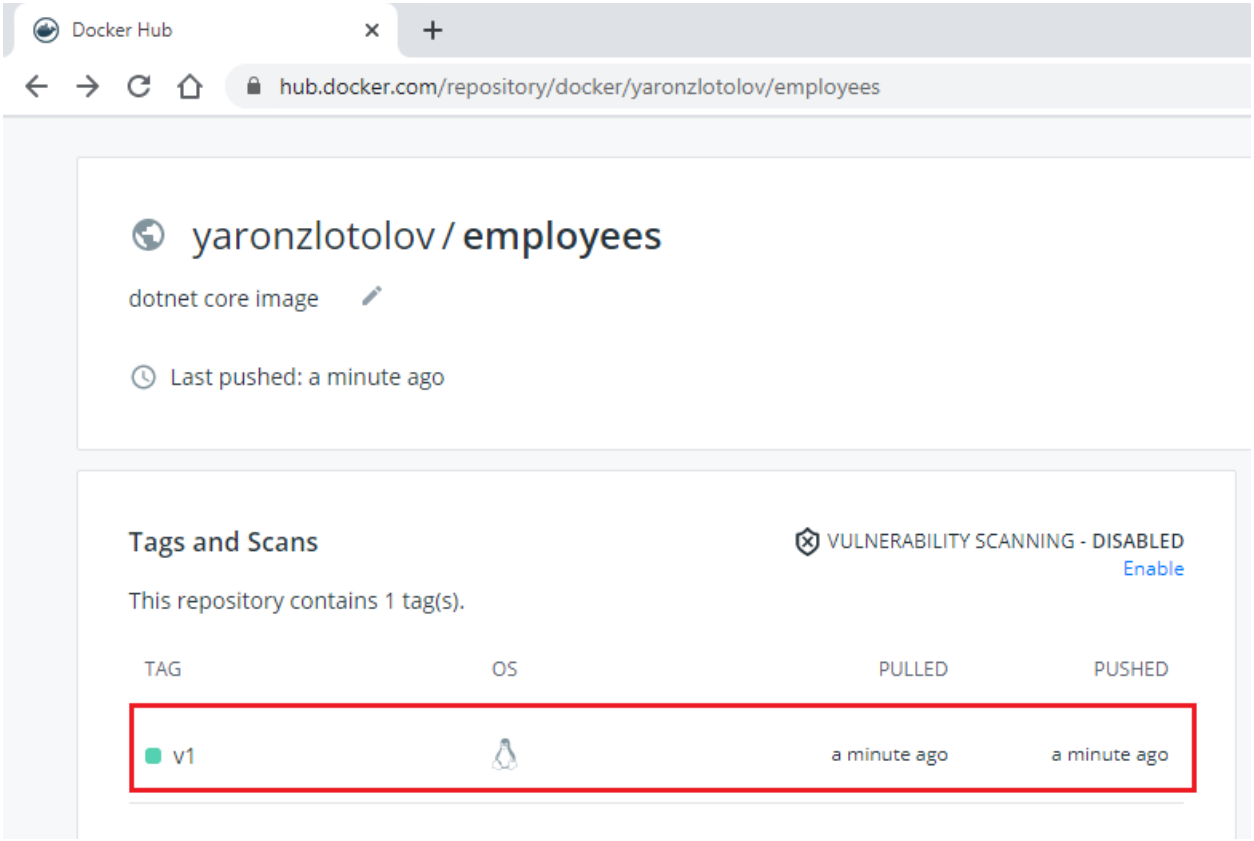
```
PS C:\Kuberneties\EmployeesManagement> docker login
Login with your Docker ID to push and pull images from Docker Hub
Username: yaronzlotolov
Password:
Login Succeeded
PS C:\Kuberneties\EmployeesManagement>
```

The command below will push the employees image to DockerHub:

docker push yaronzlotolov/employees:v1

```
PS C:\Kuberneties\EmployeesManagement> docker push yaronzlotolov/employees:v1
The push refers to repository [docker.io/yaronzlotolov/employees]
0265d76b3a65: Pushed
5f70bf18a086: Pushed
fdad21fa7fcd: Pushed
139aa1a59c8a: Pushed
f3bc884425c5: Pushed
6db0849b3e99: Pushed
800afb4d883b: Pushed
cb42413394c4: Pushed
v1: digest: sha256:582095abb05b04e3589af622cdc92ad2f0046c5f8fb5a18c53c67c835ef65abb size: 1999
PS C:\Kuberneties\EmployeesManagement>
```

Refresh the page and see the V1 employees image is pushed and ready for deployment:



Done!