Cloud Development, 2021 fall.
Lab 5

Report

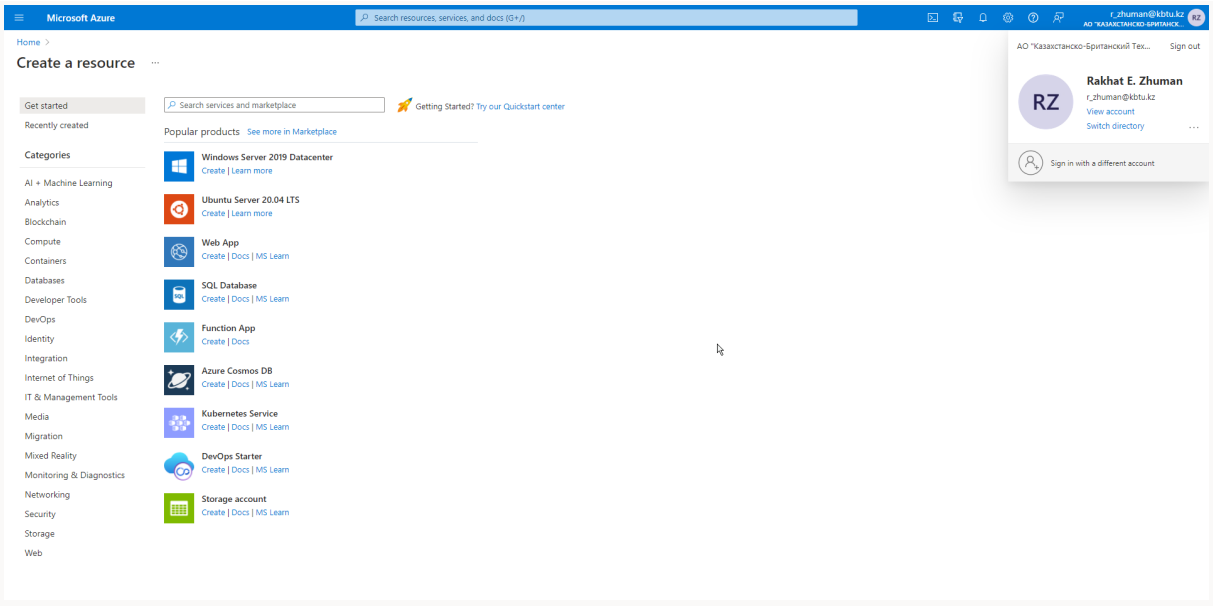
Deploying compute workloads by using images and containers

made by Zhuman Rakhat

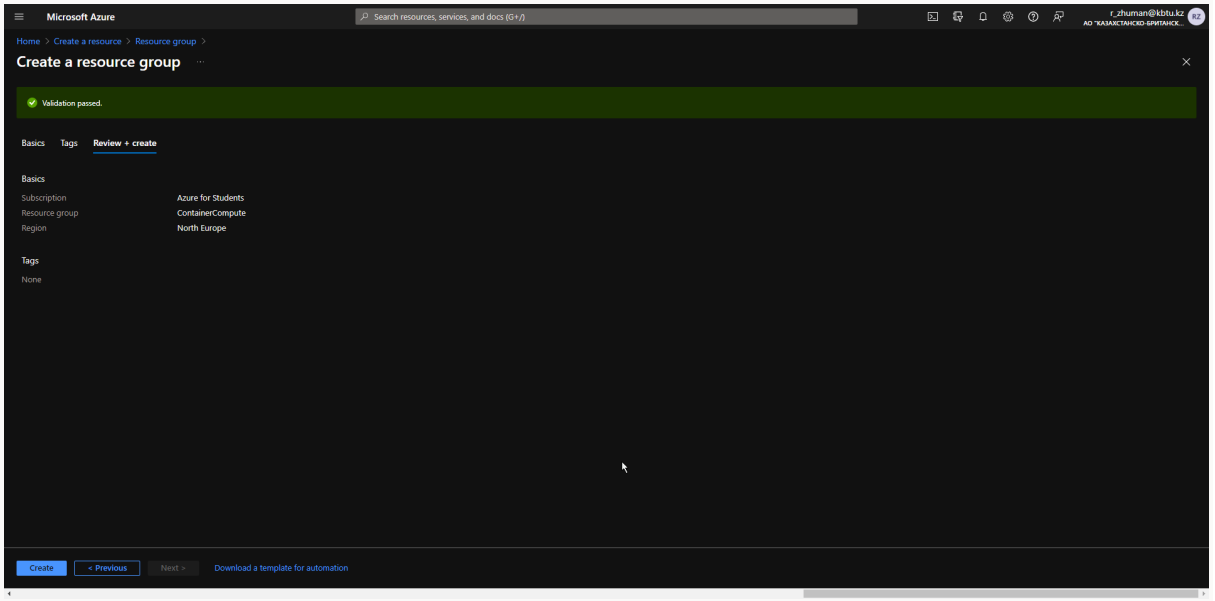
Almaty 2021

Exercise 1: Create a VM by using the Azure CLI

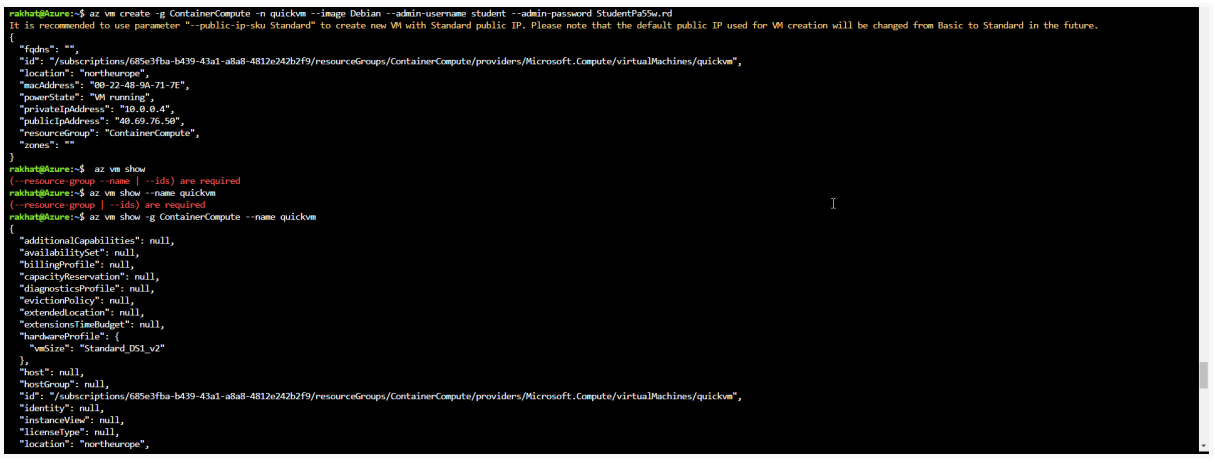
Task 1: Open the Azure portal



Task 2: Create a resource group



Task 4: Use the Azure CLI commands



```
rakhat@Azure:~$ az vm list-ip-addresses --resource-group ContainerCompute --name quickvm
[
  {
    "virtualMachine": {
      "name": "quickvm",
      "network": {
        "privateIpAddresses": [
          "10.0.0.4"
        ],
        "publicIpAddresses": [
          {
            "id": "/subscriptions/685e3fba-b439-43a1-a8a8-4812e242b2f9/resourceGroups/ContainerCompute/providers/Microsoft.Network/publicIPAddresses/quickvmPublicIP",
            "ipAddress": "40.69.76.50",
            "ipAllocationMethod": "Dynamic",
            "name": "quickvmPublicIP",
            "resourceGroup": "ContainerCompute"
          }
        ]
      },
      "resourceGroup": "ContainerCompute"
    }
  }
]
rakhat@Azure:~$
```

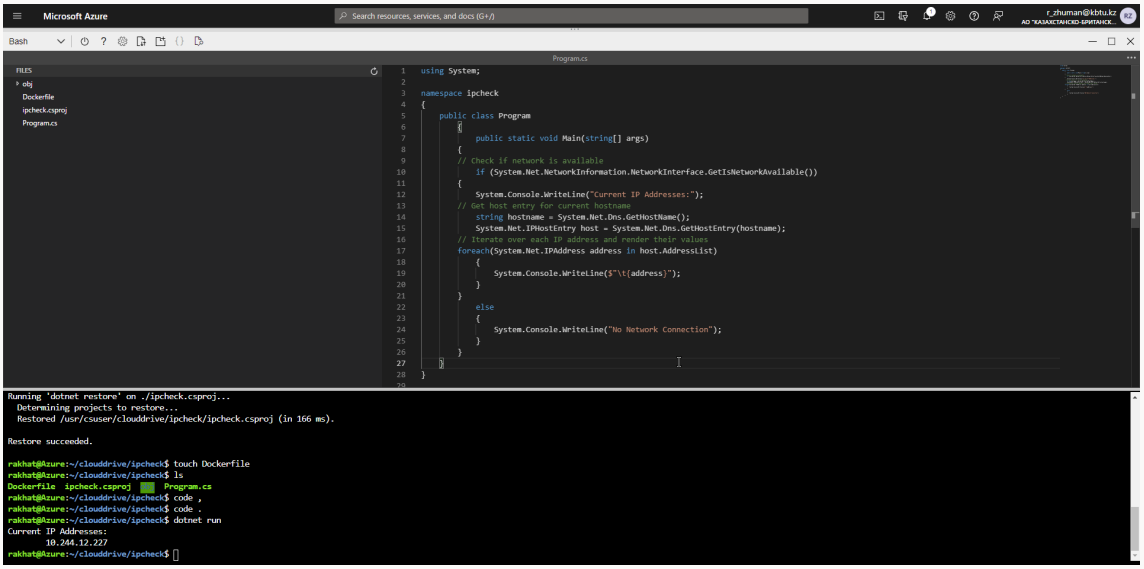
```
rakhat@Azure:~$ az vm list-ip-addresses --resource-group ContainerCompute --name quickvm --query '[].{ip:virtualMachine.network.publicIpAddresses[0].ipAddress}' --output tsv
40.69.76.50
rakhat@Azure:~$ echo tsv
tsv
rakhat@Azure:~$ ls
clouddrive
rakhat@Azure:~$ ipAddress=$(az vm list-ip-addresses --resource-group ContainerCompute --name quickvm --query '[].{ip:virtualMachine.network.publicIpAddresses[0].ipAddress}' --output tsv)
rakhat@Azure:~$ echo $ipAddress
40.69.76.50
rakhat@Azure:~$ ssh student@$ipAddress
The authenticity of host '40.69.76.50 (40.69.76.50)' can't be established.
ECDSA key fingerprint is SHA256:pbSoraxhVQ76s/h+yexOh2HIj8N8IX3z3MsMyrUE6Eo.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '40.69.76.50' (ECDSA) to the list of known hosts.
student@40.69.76.50's password:
Permission denied, please try again.
student@40.69.76.50's password:
Linux quickvm 4.19.0-18-cloud-amd64 #1 SMP Debian 4.19.208-1 (2021-09-29) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
student@quickvm:~$ uname -a
Linux quickvm 4.19.0-18-cloud-amd64 #1 SMP Debian 4.19.208-1 (2021-09-29) x86_64 GNU/Linux
student@quickvm:~$ exit
logout
Connection to 40.69.76.50 closed.
rakhat@Azure:~$
```

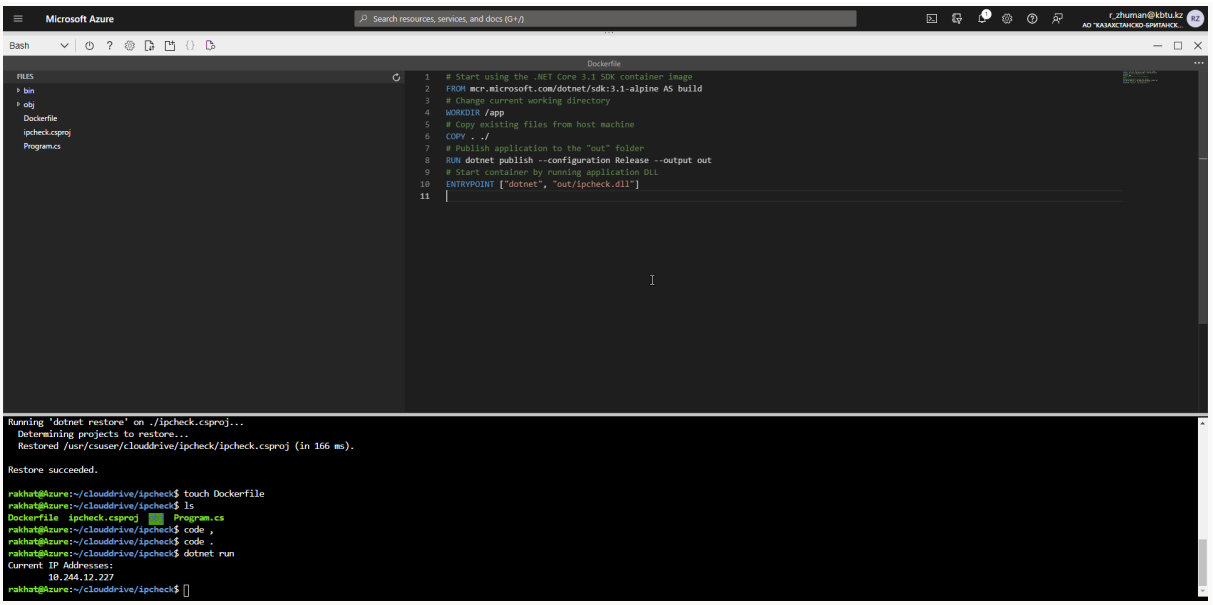
Exercise 2: Create a Docker container image and deploy it to Container Registry

Task 1: Open the Cloud Shell and editor

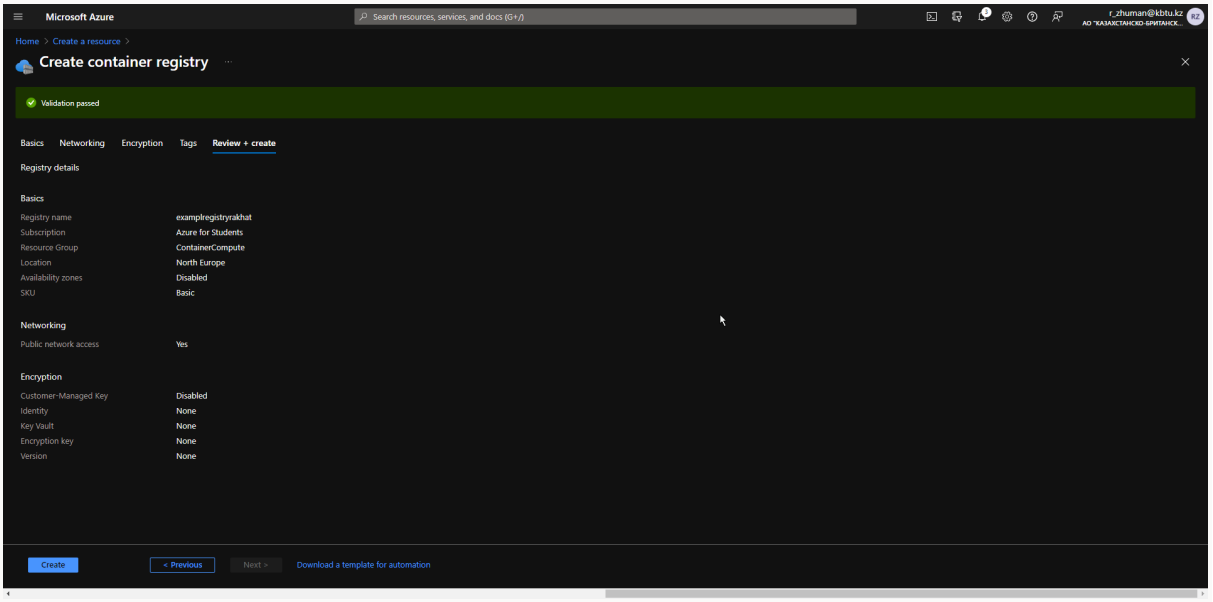
```
rakhat@Azure:~$ cd clouddrive
rakhat@Azure:~/clouddrive$ ls
rakhat@Azure:~/clouddrive$ mkdir ipcheck
rakhat@Azure:~/clouddrive$ cd ipcheck/
rakhat@Azure:~/clouddrive/ipcheck$ dotnet new console --output . --name ipcheck

Welcome to .NET Core 3.1!
---------------------
SDK Version: 3.1.403

Telemetry
---------
The .NET Core tools collect usage data in order to help us improve your experience. The data is anonymous. It is collected by Microsoft and shared with the community. You can opt-out of telemetry by setting the DOTNET_CLI_TELEMETRY_OPTOU
T environment variable to '1' or 'true' using your favorite shell.

Read more about .NET Core CLI Tools telemetry: https://aka.ms/dotnet-cli-telemetry

----------------
Explore documentation: https://aka.ms/dotnet-docs
Report issues and find source on GitHub: https://github.com/dotnet/core
Find out what's new: https://aka.ms/dotnet-whats-new
Learn about the installed HTTPS developer cert: https://aka.ms/aspnet-core-https
Use 'dotnet --help' to see available commands or visit: https://aka.ms/dotnet-cli-docs
Write your first app: https://aka.ms/first-net-core-app
--------------------------------------------------------------------------------------
Getting ready...
The template "Console Application" was created successfully.

Processing post-creation actions...
Running 'dotnet restore' on ./ipcheck.csproj...
  Determining projects to restore...
  Restored /usr/csuser/clouddrive/ipcheck/ipcheck.csproj (in 166 ms).

Restore succeeded.

rakhat@Azure:~/clouddrive/ipcheck$ touch Dockerfile
rakhat@Azure:~/clouddrive/ipcheck$ ls
Dockerfile  ipcheck.csproj  obj  Program.cs
rakhat@Azure:~/clouddrive/ipcheck$
```

Task 2: Create and test a .NET application



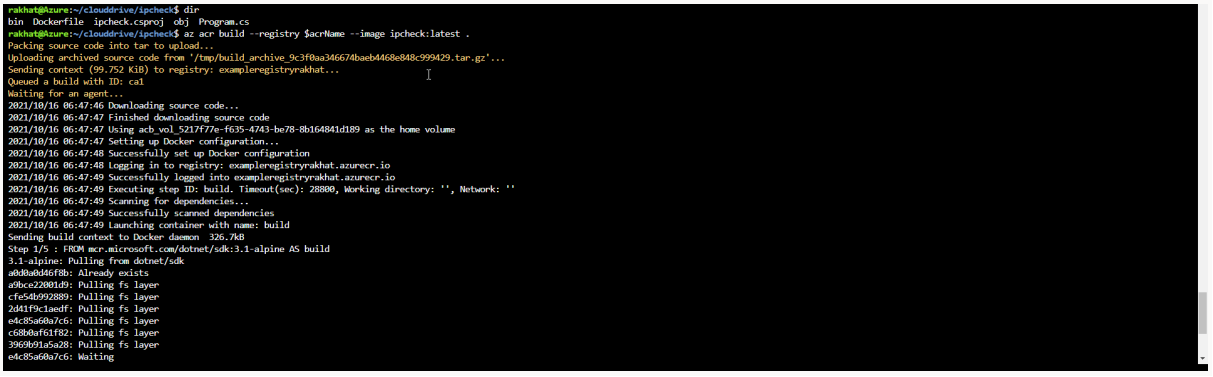


Task 3: Create a Container Registry resource

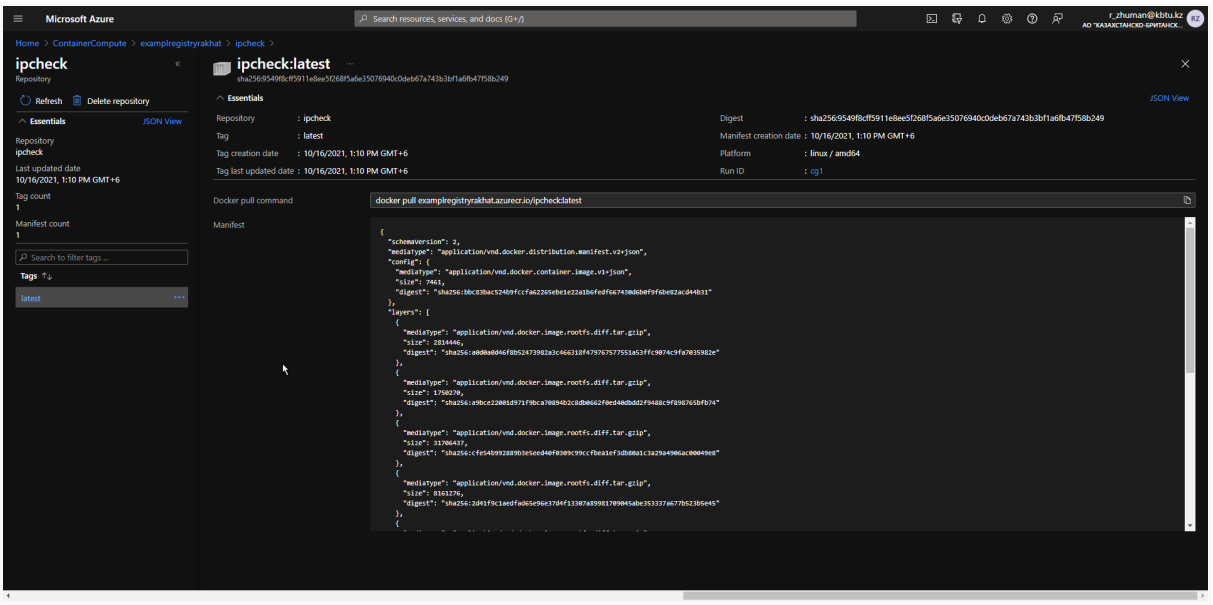


Task 4: Open Azure Cloud Shell and store Container Registry metadata

Task 5: Deploy a Docker container image to Container Registry

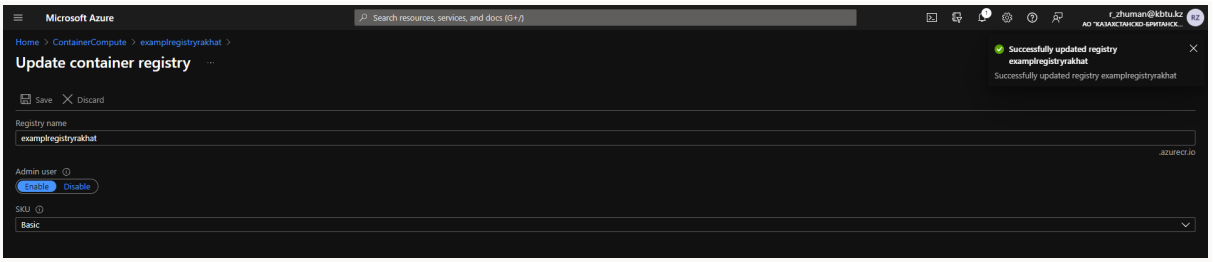


Task 6: Validate your container image in Container Registry

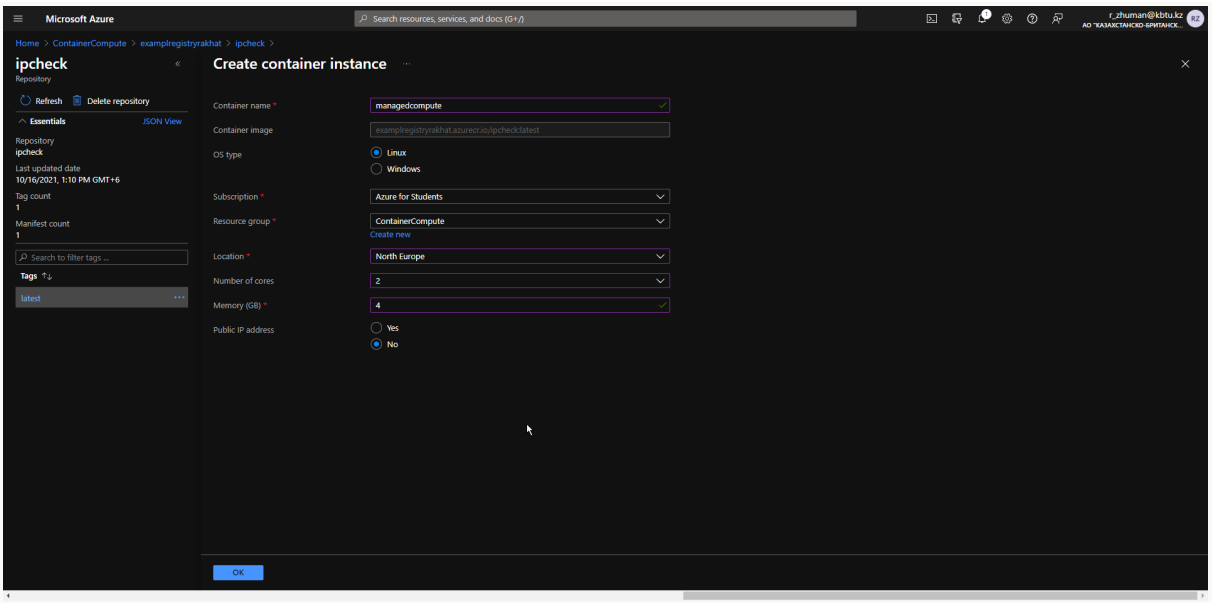


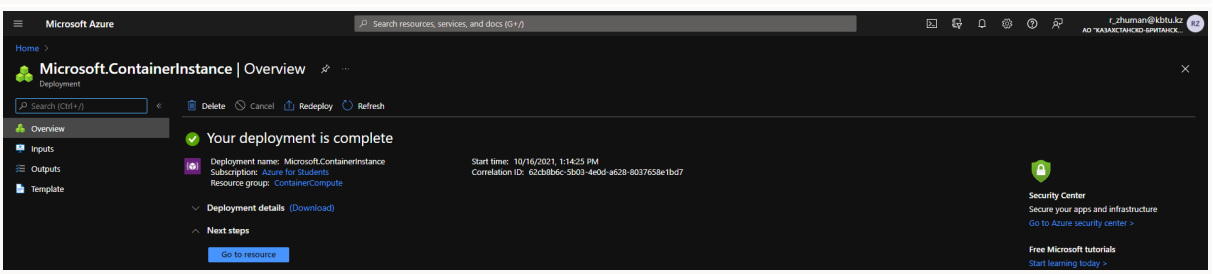
Exercise 3: Deploy an Azure container instance

Task 1: Enable the admin user in Container Registry

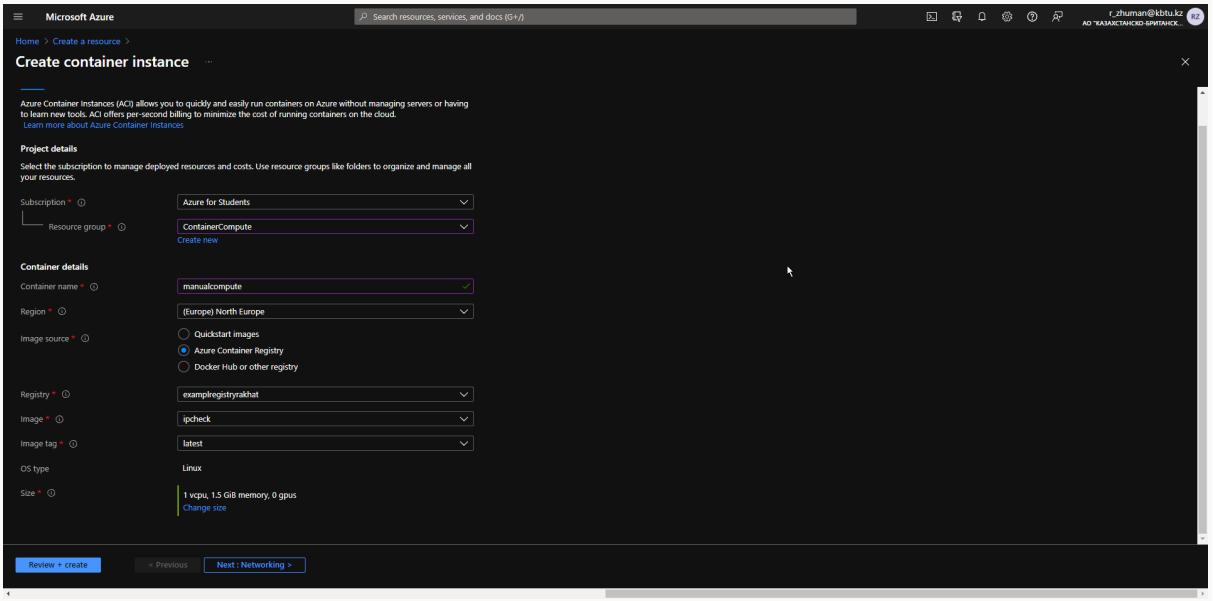


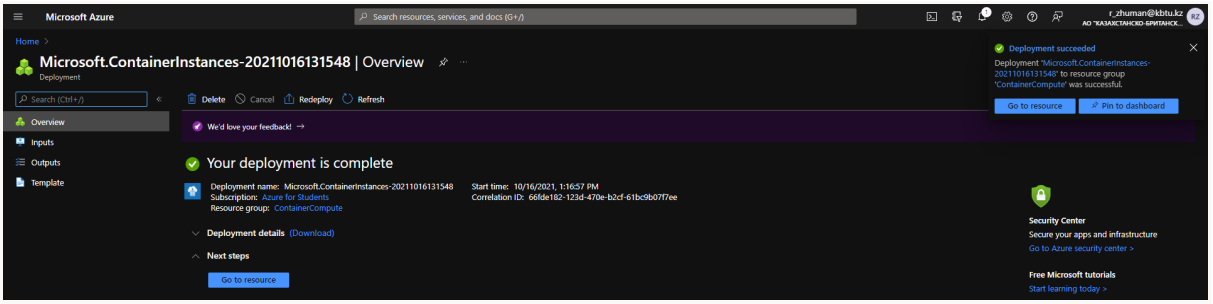
Task 2: Automatically deploy a container image to an Azure container instance



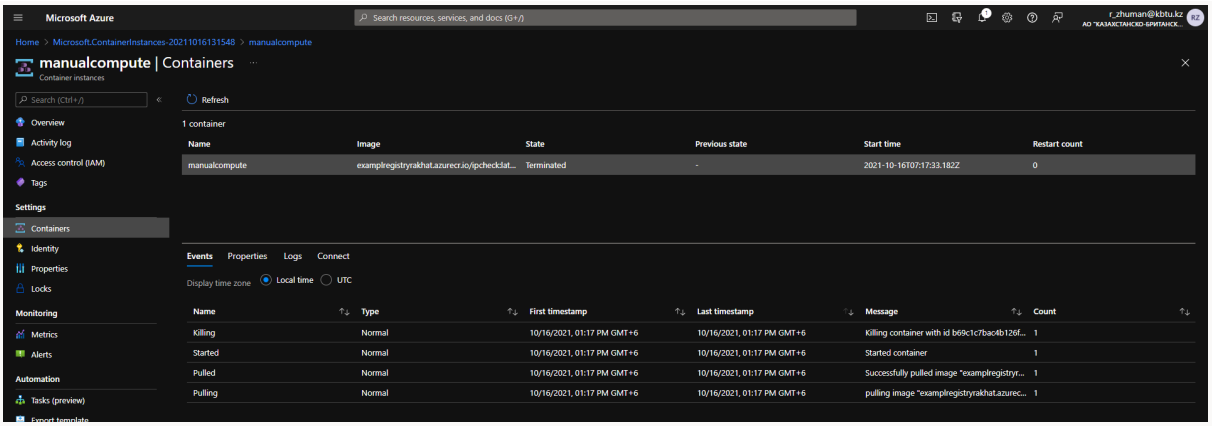


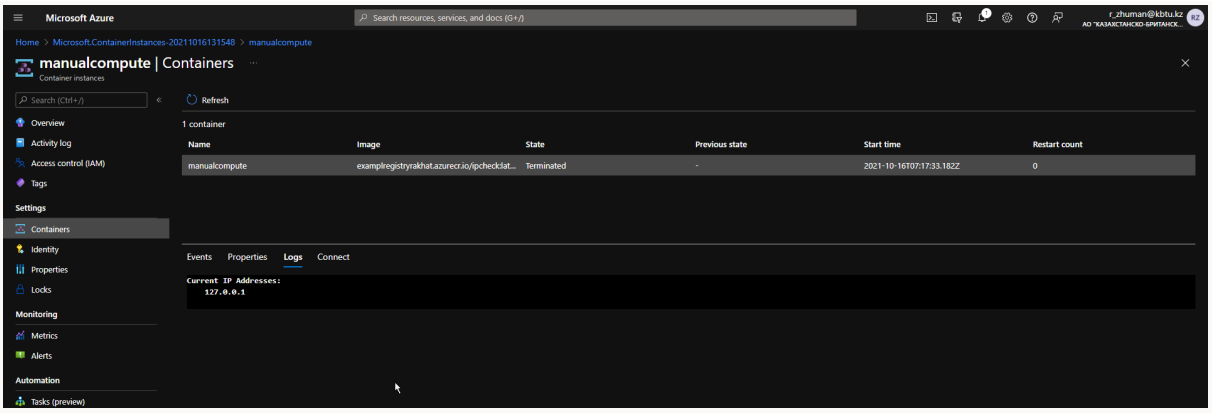
Task 3: Manually deploy a container image to Container Instance





Task 4: Validate that the container instance ran successfully





Exercise 4: Clean up your subscription

Task 2: Delete resource groups