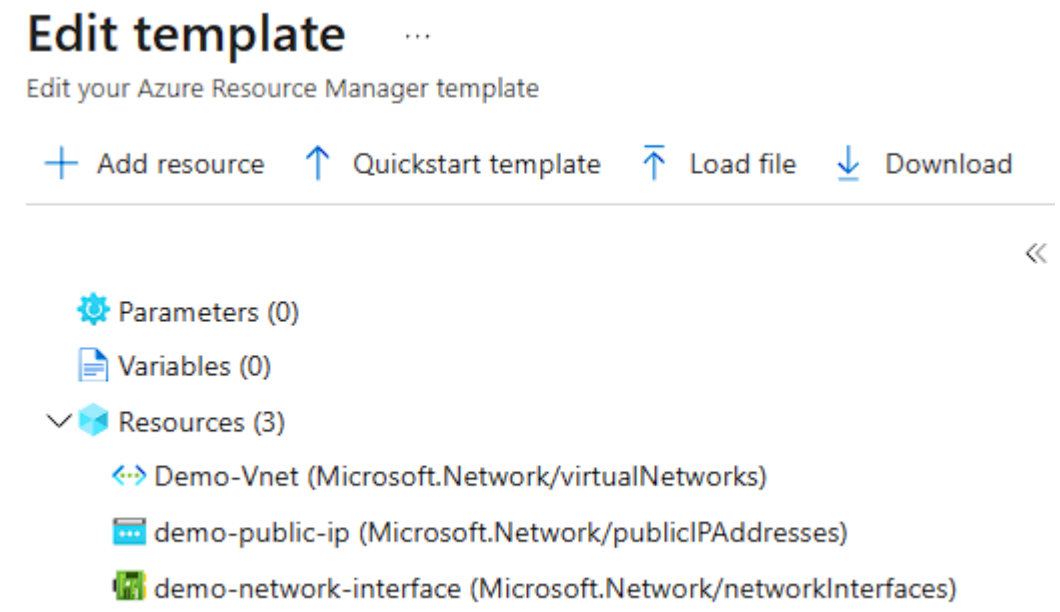
😊 Network Interface with a Virtual Network

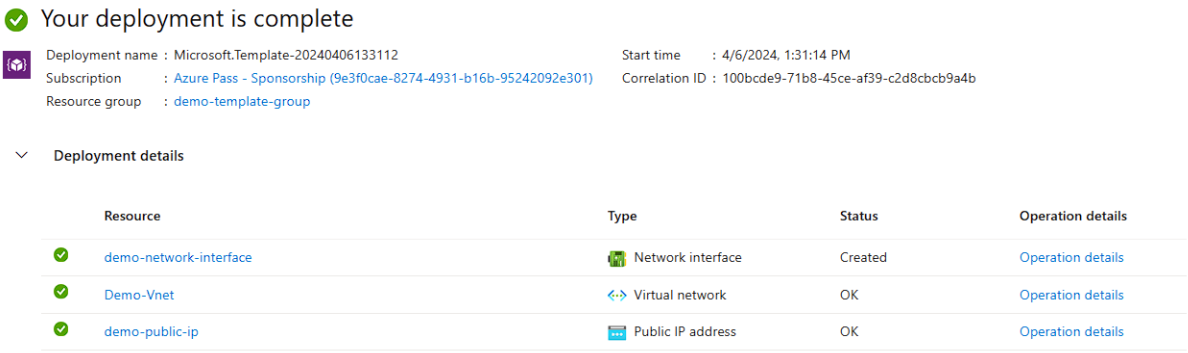
1. Now in this lab, I want to go through the deployment of a virtual network, a public IP address, and a network interface within one template itself.
2. What we have done in the previous lab is to deploy each resource separately but this time around we will create everything in a single template.
3. First if you have previous resources then go and delete them. Then start creating your template in VS Code.
4. Now to create this template we need to copy the code from our previous templates.
5. First, we need to put the code to create Vnet. Then we are going to create the Public IP after that in the last we are going to create the network interface.
6. Now the thing is, if while creating everything, Azure did not follow the order that we have given to it then we'll get an error.
7. Let's suppose, while deploying the template the deployment start with creating network interface then we will definitely get the error because we don't have the Vnet and the public IP then what will be attached with network interface.
8. So, to avoid such error we are going to use the Depends on clause.
9. In the "demo-network-interface" resource, the "dependsOn" property is added with the resource IDs of the virtual network and public IP address resources. This ensures that the network interface deployment waits for these resources to be created before proceeding.

```
{
    "name": "demo-network-interface",
    "type": "Microsoft.Network/networkInterfaces",
    "apiVersion": "2023-04-01",
    "location": "[resourceGroup().location]",
    "dependsOn": [
        "[resourceId('Microsoft.Network/virtualNetworks', 'Demo-Vnet')]",
        "[resourceId('Microsoft.Network/publicIPAddresses', 'demo-public-ip')]"
    ],
```

10. Now once you are done with the code then save it in VS Code and paste it in the template deployment.
11. Here you can see the order of creation. First, it will create the Vnet then public IP and then it will create the network interface.



12. Below you can see that our resources have been created. Now you can visit them and see for yourself.



13. Now go directly to the network interface and it will show if our Vnet and public IP have been attached or not.
14. And below you can see that they are attached successfully.