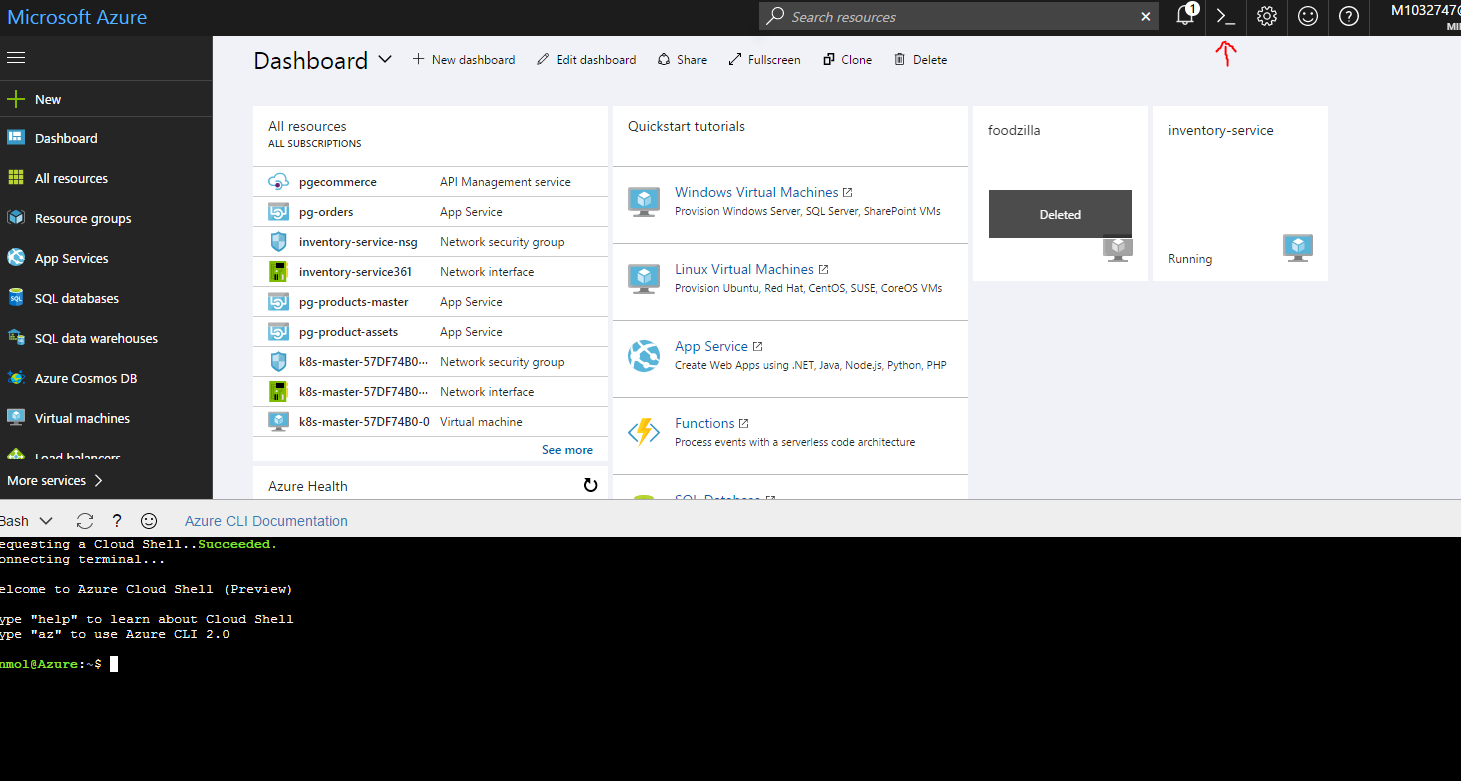
**Kubernetes Cluster Setup using Azure Container Service**

**Prerequisites:**

1. Azure Account
2. Azure CLI 2.0 (in your local system, if possible). If you don’t want to you can use the one provided by Azure Portal to do most of the things
3. Kubectl.exe file in your local machine (a package used to run kubernetes command). This is needed to see Kubernetes Dashboard once deployed on local as they don’t provide direct access to dashboard. Again its optional
4. SSH Keys – You can create them initially and store them in ~/.ssh/id\_rsa.pub file or you can create them later during cluster creation.

**Steps:**

1. If you are not planning to create a cluster only and want to join our existing cluster, kindly ignore the steps 2- 8 and directly go on step 9.
2. Login to Azure Portal with your credentials and click on Azure CLI. The CLI will appear as shown in figure



1. Creation of **Resource Group**

Write these commands in Azure CLI

**RESOURCE\_GROUP= <my-resource-group>(any name you want to give)**

**LOCATION=westus (or any other upto you)**

**az group create --name=$RESOURCE\_GROUP --location=$LOCATION**

(this command creates a resource group, same can be done from UI also)

1. Write these commands then to add a DNS\_PREFIX and CLUSTER\_NAME

**DNS\_PREFIX=<some-unique-value>** (it will appear in DNS of Master)

**CLUSTER\_NAME=any-acs-cluster-name** (give a proper name as it will be used later on)

1. Starting the Cluster through ACS

Write this command next in the CLI-

**az acs create**

**--orchestrator-type=kubernetes**

**--resource-group <my-resource-group>**

**--name=$CLUSTER\_NAME**

**--dns-prefix=$DNS\_PREFIX**

**--generate-ssh-keys**

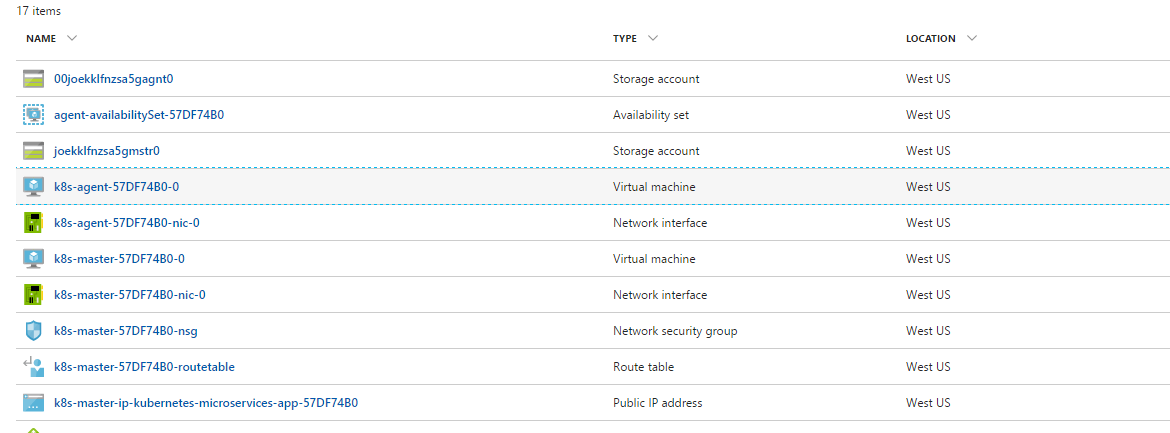
**--admin-username=<username>**

**--admin-password=<password>**

**--master-count=1**

**--agent-count=1**

After few mins when this command completes, you will see a cluster successfully being setup under the resource group you created. Once you click on that resource group, you will see few items like the ones given below with few of them having names starting with k8



1. After this much , write the following command

**az acs kubernetes get-credentials --resource-group=<my-resource-group> --name=$CLUSTER\_NAME**

This command will download the Kubernetes cluster Configuration to the **~/.kube/config** file. This is very important command and shouldn’t be missed otherwise cluster wont be setup properly

1. After this there are 2 ways to go about, first you can connect to cluster through CLI only in Azure Portal, but if you want to see the Kubernetes Dashboard after the setup, ill recommend to see the download steps for installing Kubectl library in your local machine and run it as a part of Command Prompt.

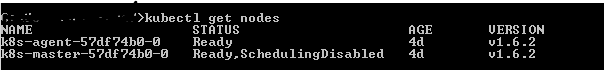
Refer to this link for installing kubectl: (<https://kubernetes.io/docs/tasks/tools/install-kubectl/> )

1. If you have installed Kubectl on local now, copy the file from **~/.kube/config** file in the AZURE CLI on azure portal and paste it inside **<C:\Users\<your username>\.kube\>**. Also add kubectl.exe directory path in your system path variables. After that open cmd, you are ready to use all the kubectl commands and connect to your cluster from your local also apart from Azure CLI inside Azure portal.
2. In order to connect to the existing cluster which is already deployed by us from your local or azure portal , kindly copy and paste this config file attached in this documentation to **<C:\Users\<your username>\.kube**> for windows or in **~/.kube/ folder for azure portal CLI** and run the kubectl commands from cmd/CLI.

Refer to this link for installing kubectl: (<https://kubernetes.io/docs/tasks/tools/install-kubectl/> )

1. Run **kubectl get nodes**

This will give you information about the master node and agent node running in the cluster



Once you this and see the status of both nodes as ready, that means your cluster is working fine

1. Use this command :

**kubectl proxy**

This will help us view the Kubernetes Dashboard by setting up a proxy tunnel to the Kubenetes

API Server. You can view the dashboard on <http://localhost:8001/ui>

It will look like this

