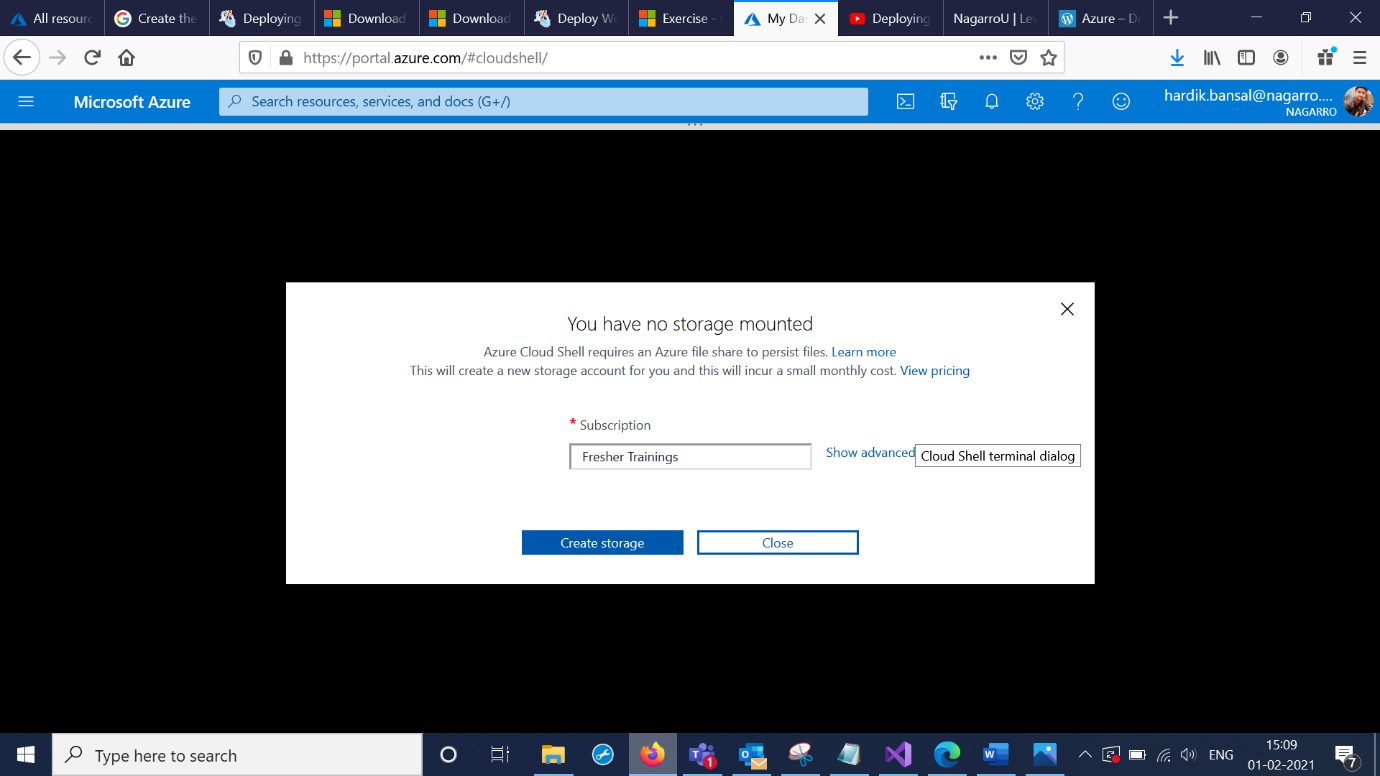
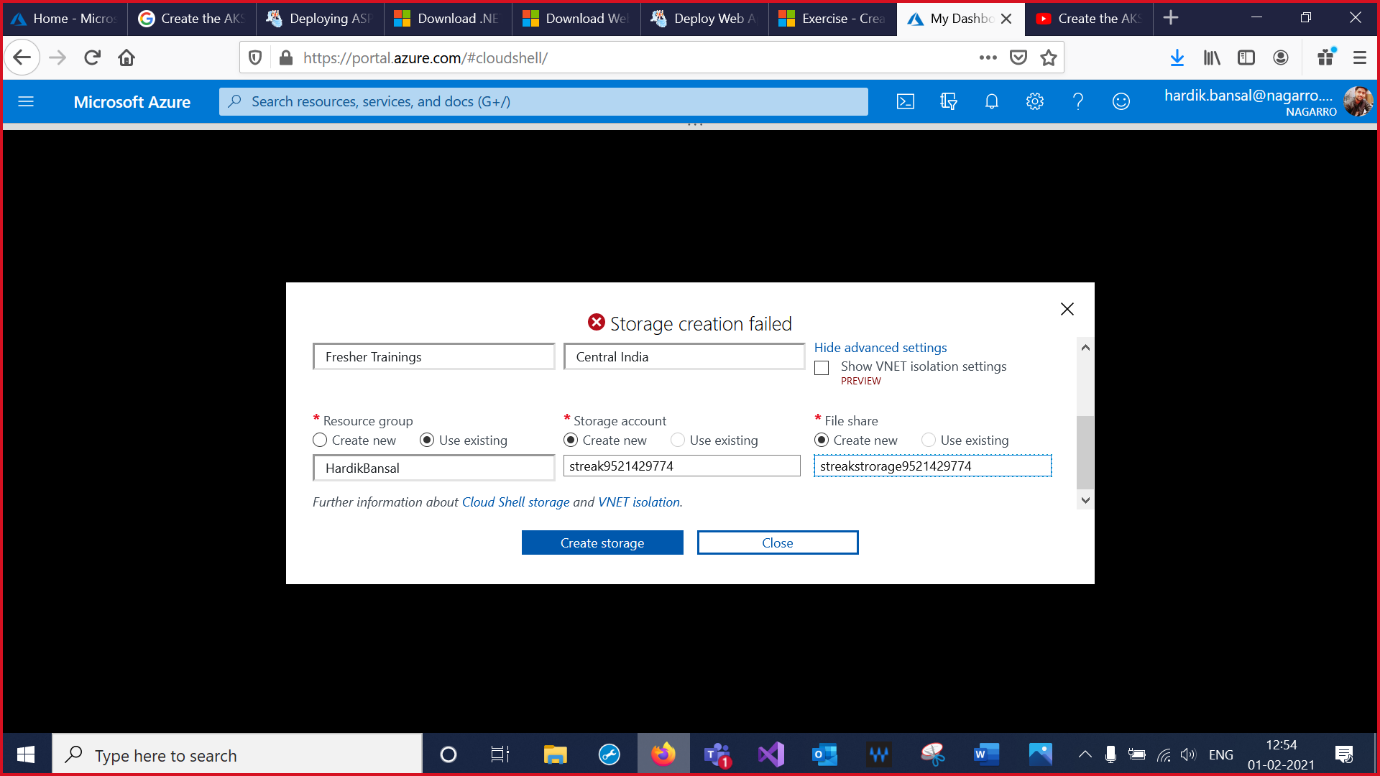
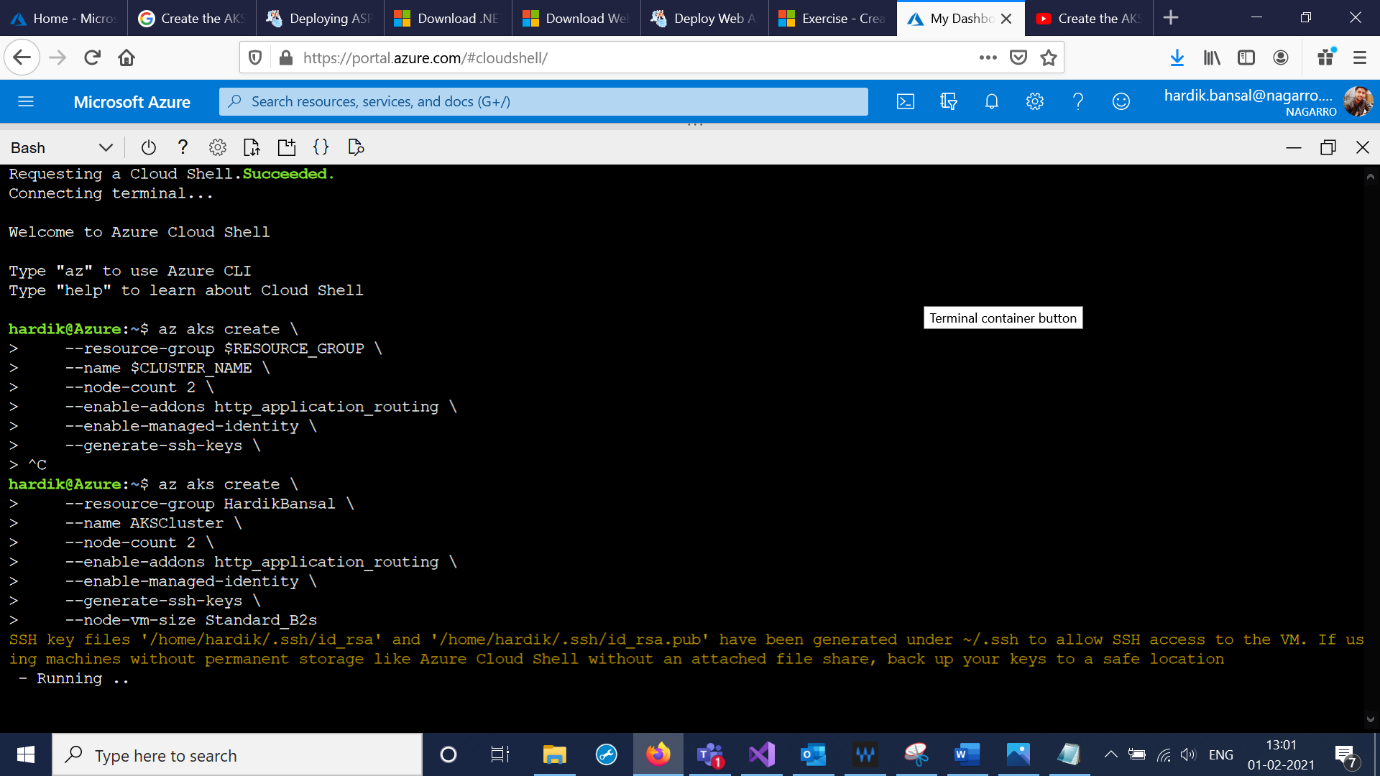
**Create the AKS cluster (2 nodes, smallest size VM) and deploy any two services on it. Services should be accessible from the internet.**

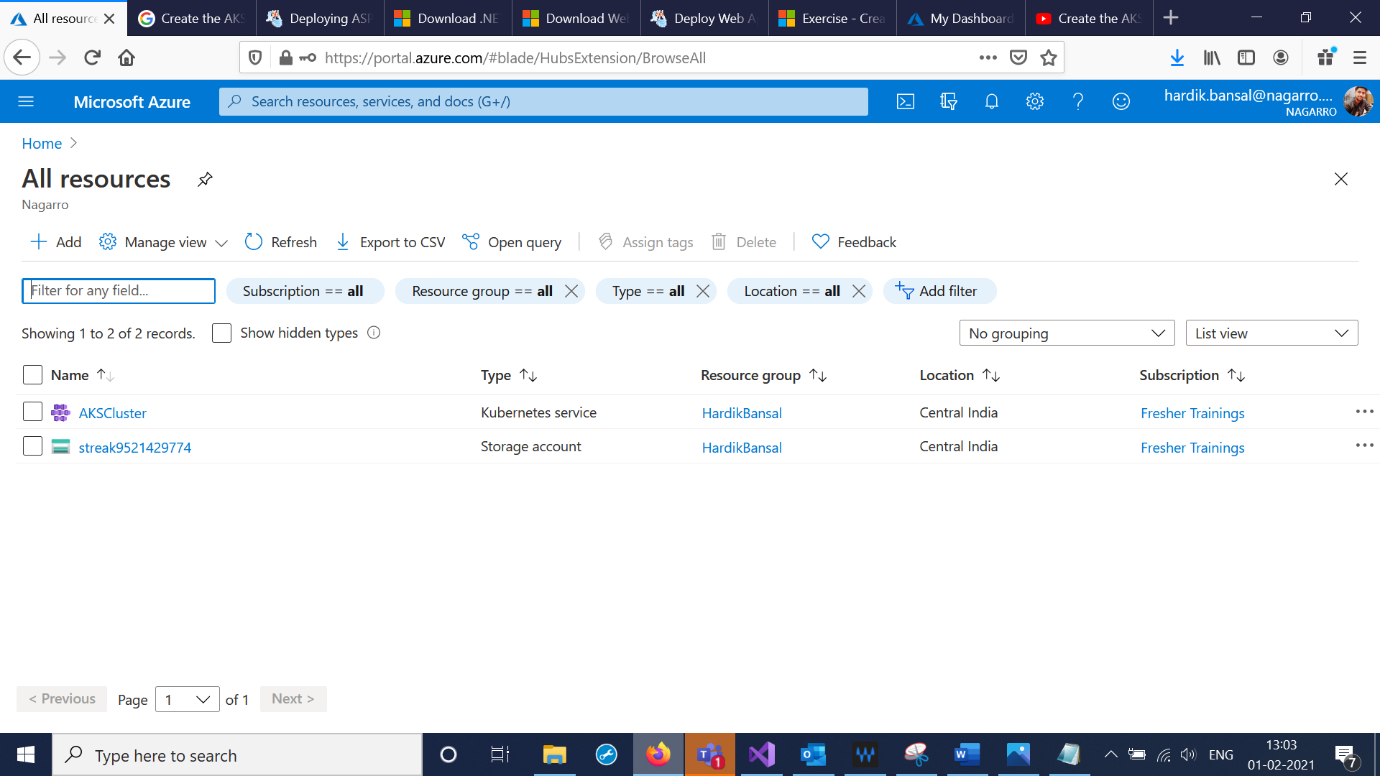
1.Select Azure CLI and select Bash ,Create Storage on Azure.



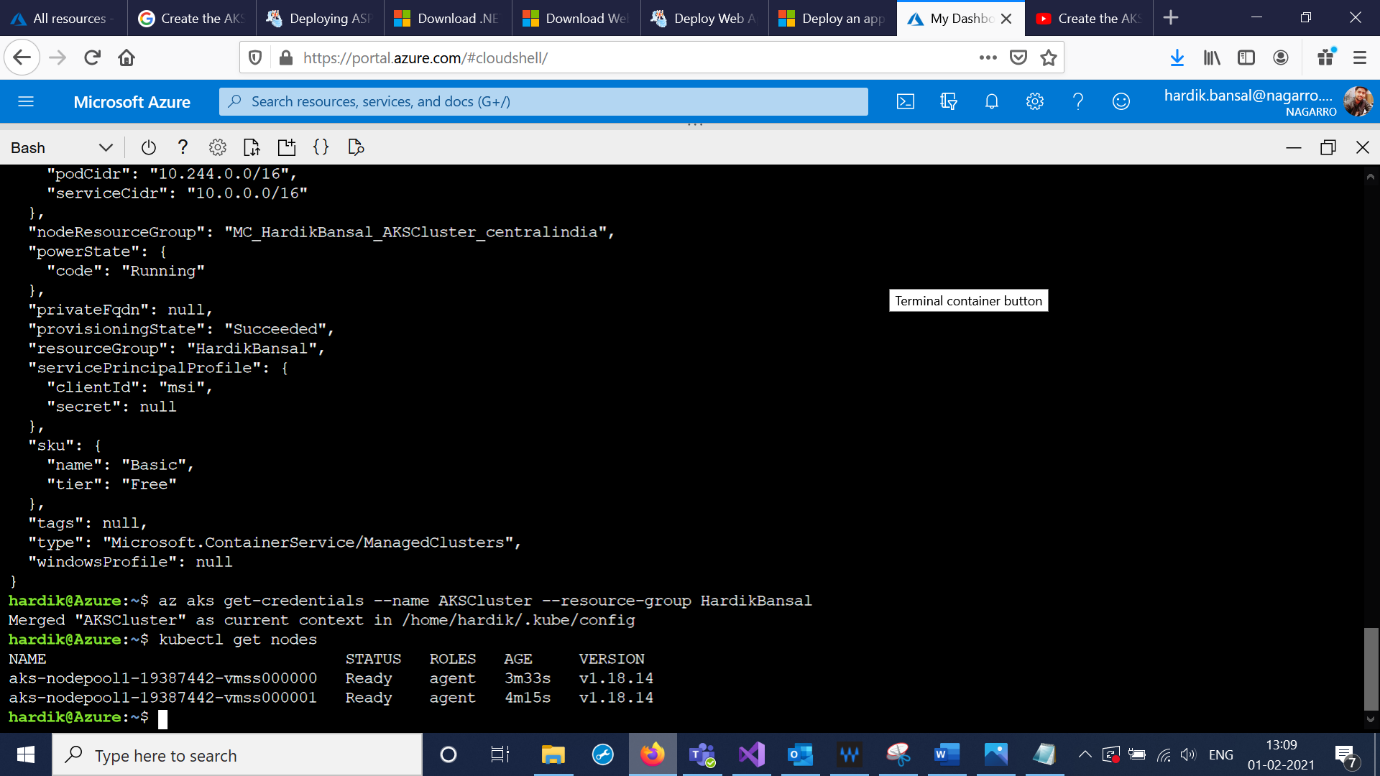


2. Run the az aks create command to create an AKS cluster.

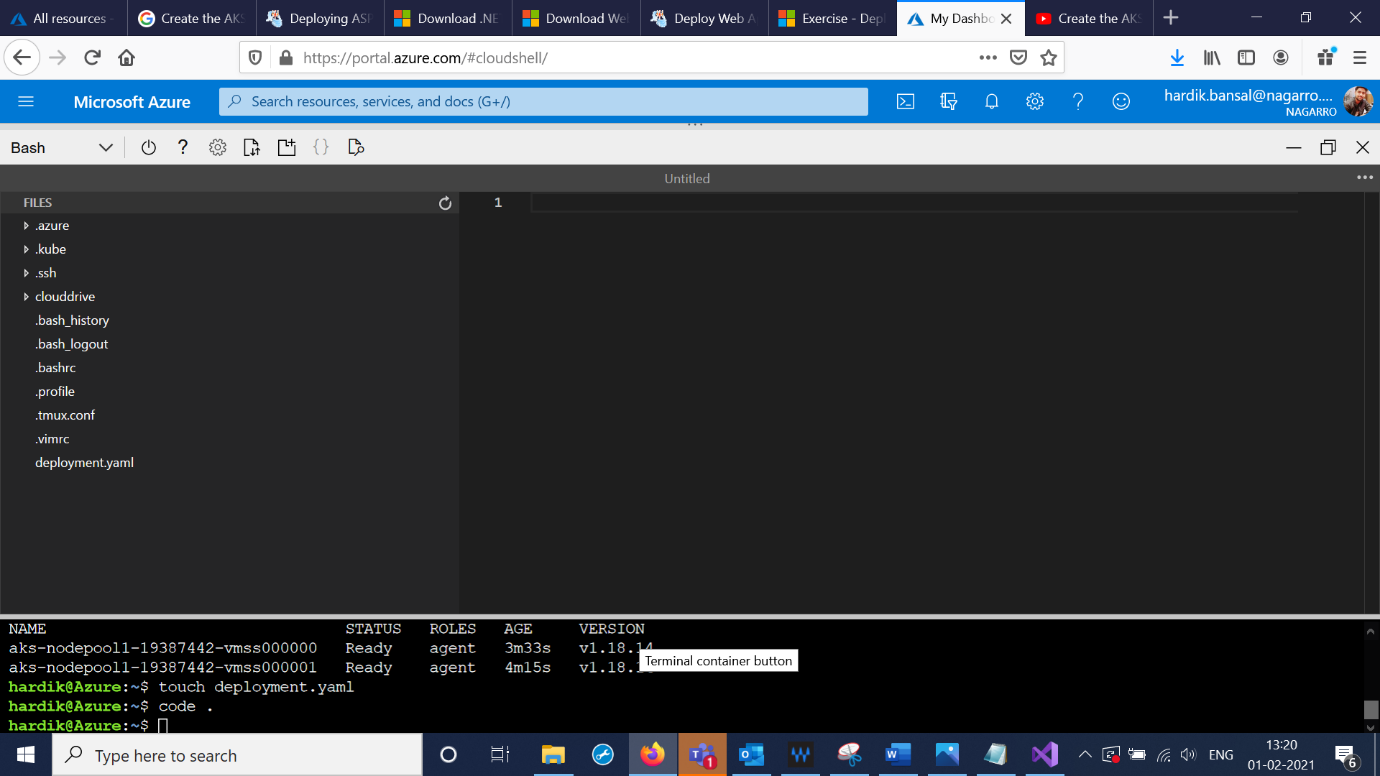




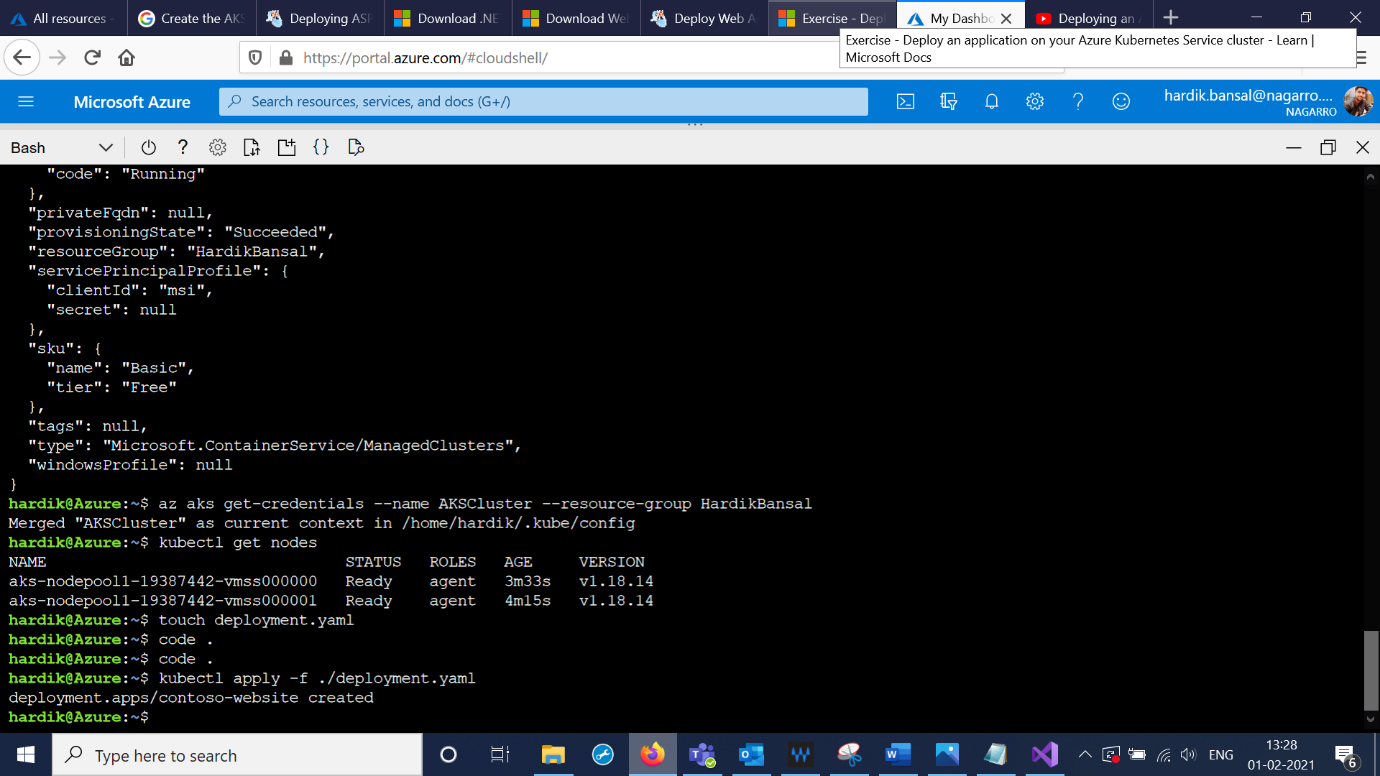
3. Link your Kubernetes cluster with kubectl by using the following command in Cloud Shell. Execute the kubectl get nodes command to check that you can connect to your cluster and confirm its configuration.

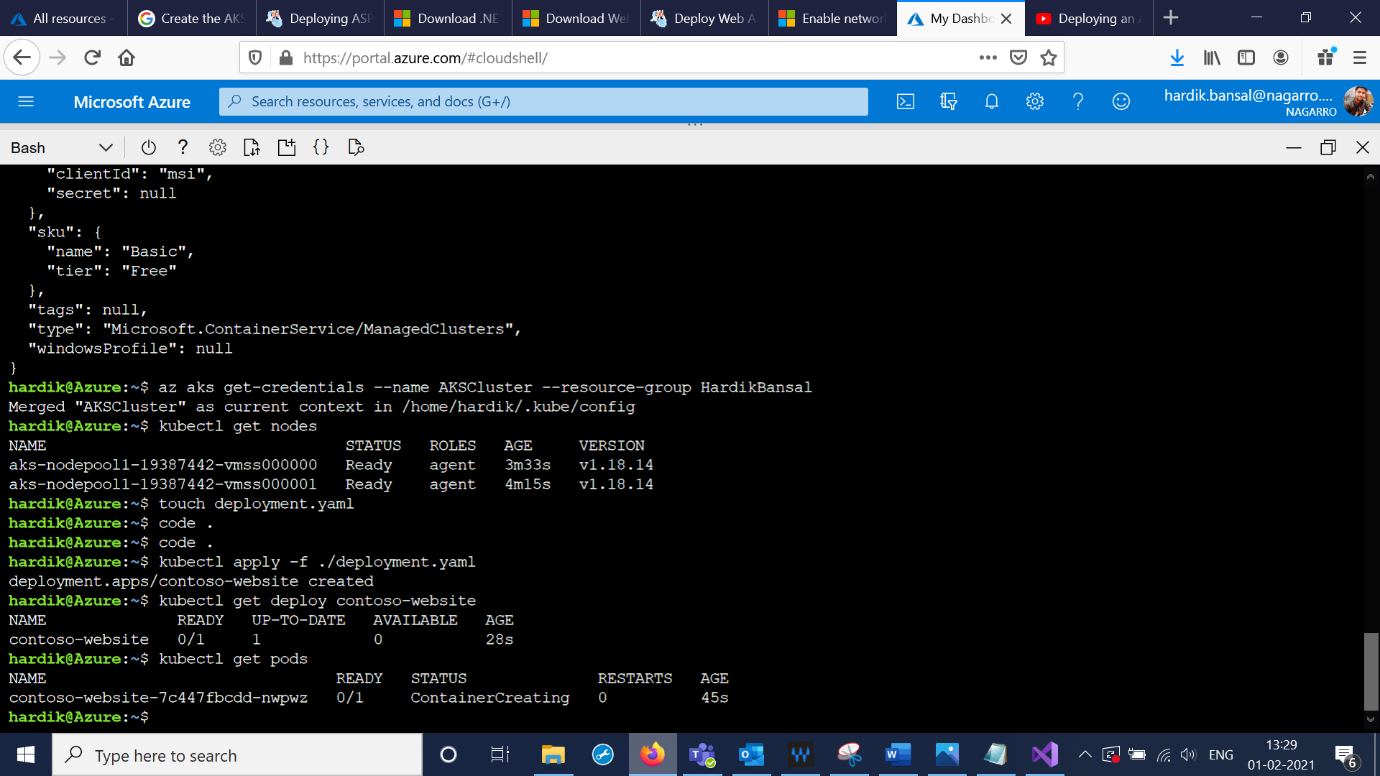


4. In Cloud Shell, create a manifest file for the Kubernetes deployment called deployment.yaml by using the integrated editor. Update the deployment.yaml file to match the following YAML.

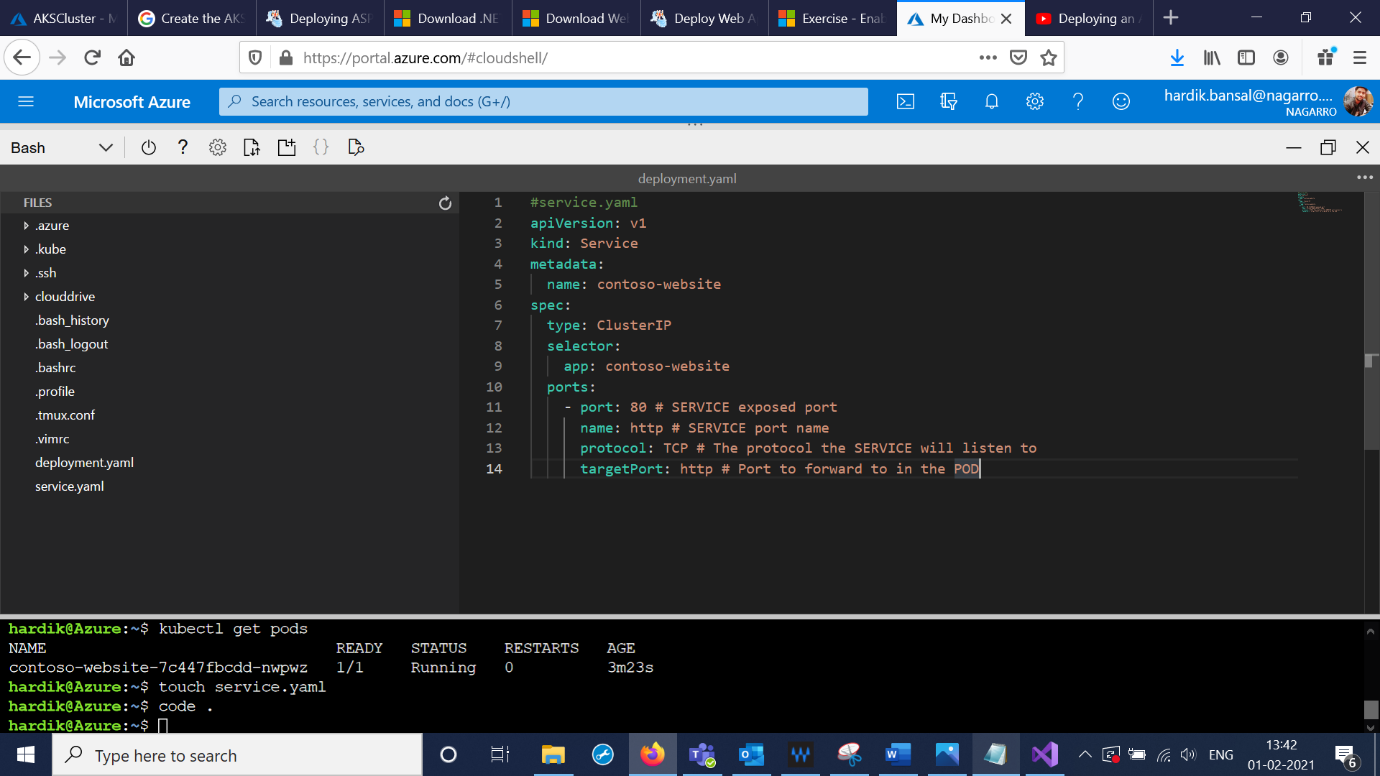


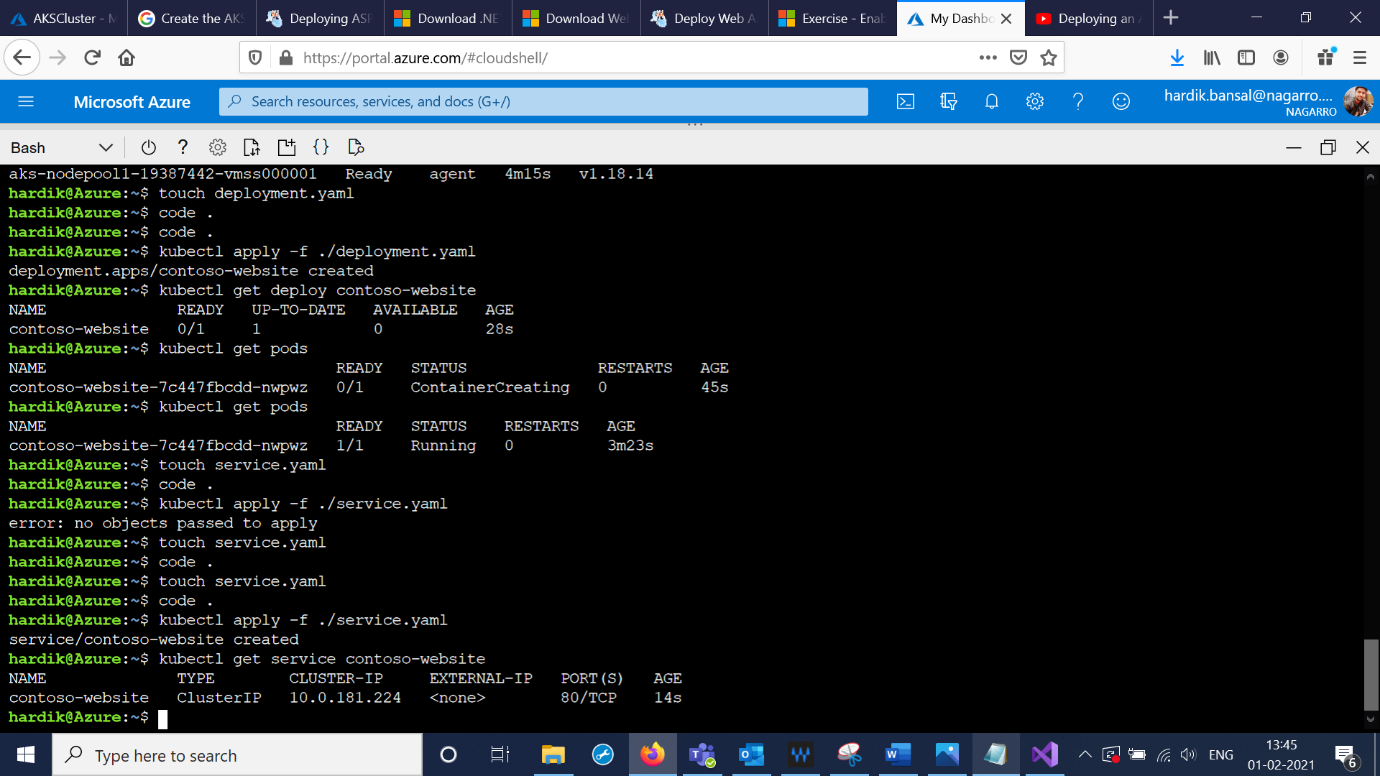
5.Apply the Manifest. In Cloud Shell, run the kubectl apply command to submit the deployment manifest to your cluster. Run the kubectl get deploy command to check if the deployment was successful. Run the kubectl get pods command to check if the pod is running.





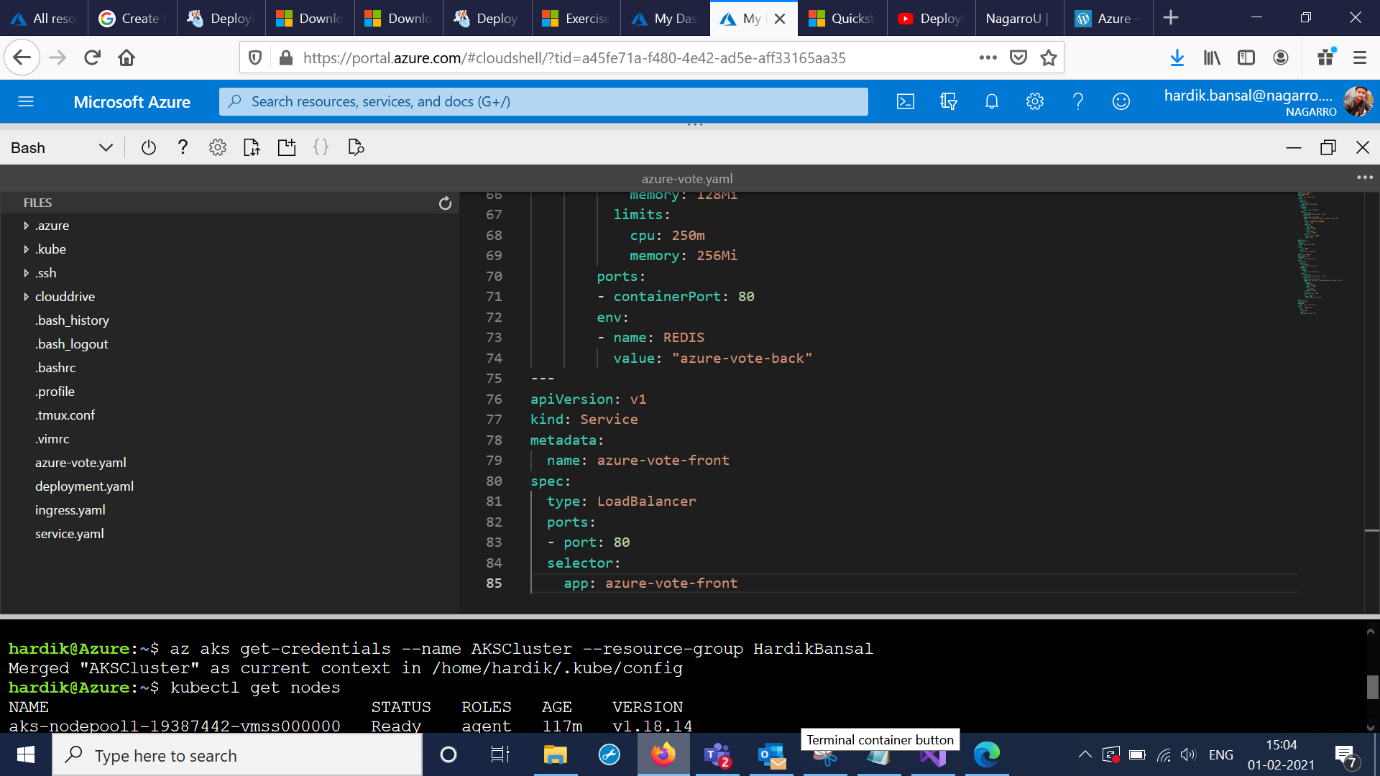
6. In Cloud Shell, create a manifest file for the Kubernetes service called service.yaml. Update the service.yaml file to match the following YAML.

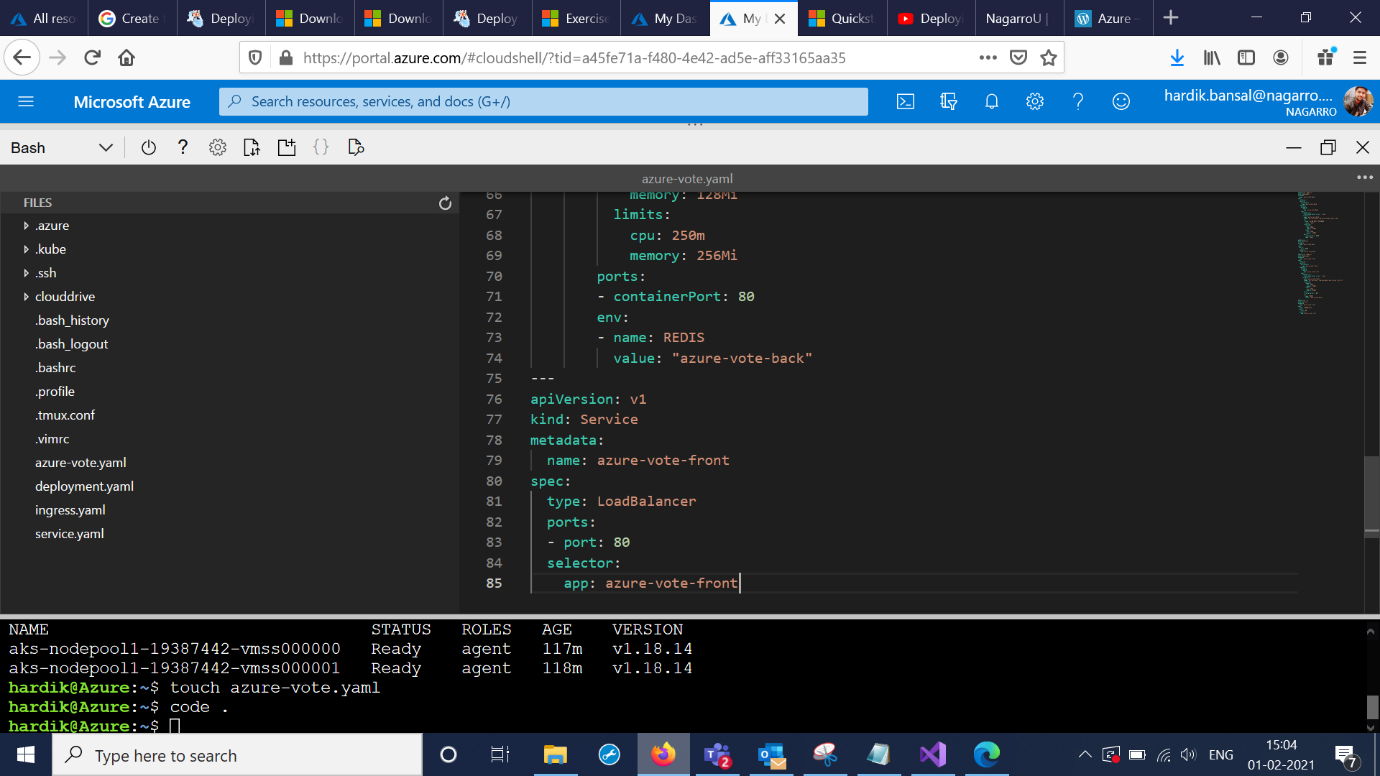




2nd App Service

7. In Cloud Shell, create a manifest file for the Kubernetes service called azure-vote.yaml. Update the azure-vote.yaml file to match the following YAML.





8.Deploy the application using the [kubectl apply](https://kubernetes.io/docs/reference/generated/kubectl/kubectl-commands#apply) command and specify the name of your YAML manifest, To monitor progress, use the [kubectl get service](https://kubernetes.io/docs/reference/generated/kubectl/kubectl-commands#get) command with the --watch argument.

When the EXTERNAL-IP address changes from pending to an actual public IP address, use CTRL-C to stop the kubectl watch process. The following example output shows a valid public IP address assigned to the service.

