[Hemanth-Azure Demos-1 (vskumarcoaching.com)](https://vskumarcoaching.com/hemanth-azure-demos-1)

**Profile Appendix on Internship:**

The provided tasks demonstrate a comprehensive understanding of Azure cloud infrastructure and services. They cover a wide range of topics, including infrastructure as code (IaC) with Docker, Terraform, VM deployment and management, database integration, web server configuration, and resource management. The tasks showcase expertise in using Azure tools and services to create, deploy, and manage various cloud resources effectively.

[**Hemanth Nimmala |LinkedIn**](http://www.linkedin.com/in/hemanth-nim)

**1.Dockerize Wordpress:** This demo showcases the deployment of a Wordpress application using Docker Compose. It involves creating and running containers for Wordpress and mySQL, establishing database connectivity, and verifying the successful installation and operation of Wordpress

**Recorded Demo:** <https://youtu.be/38NDhZqPMn0>

**2. Deploy Redis using Python:** In this demo, I explored two approaches to deploying Redis on Azure. First, I used the Azure portal to create a VM and deploy Redis, verifying its functionality. Then, I automated the process using Python Infrastructure as Code (IAC), provisioning a VM with Apache and Redis, and confirming successful deployment.

**Recorded Demo**: <https://youtu.be/_krI5c9iFCk>

**3. Deploy Apache2 using Python:** First, I deployed a VM with Apache on Azure using the portal and verified the default webpage. Then, I automated the process with Python IAC, creating a script (devVM.py) and configuration file (.env) to provision VMs with Apache and confirm their functionality in the browser.

**Recorded Demo:** <https://youtu.be/XLB-ToWrS_o>

**4.Dockerize flask application using Python:** This demo walks you through building and running a Python application with Docker. Build a multi-stage image for efficiency, and launch the container to access the application from browser.

**Recorded Demo:** <https://youtu.be/1R0HR_oHD3A>

**5**.**Docker Installation:**This demo demonstrates deploying a containerized application on Azure VM using Docker image.Provisioned a Linux VM, installed Docker, downloaded and ran a container image, and uninstalled Docker, images, and the container.

**Recorded Demo:** <https://youtu.be/01DfC-rNSw8>

**6.TF-Lighttpd**: This demo demonstrated the deployment of a Lighttpd web server on an Azure VM using both manual and automated approaches. The manual approach involved using the Azure Portal, while the automated approach utilized Terraform, an infrastructure as code tool. This comparison highlighted the benefits of automation in managing infrastructure

**Recorded demo**: <https://youtu.be/qFDpzoIu89I>

**7.TF-Nginx:** This demo showcased deploying an Nginx web server on an Azure VM. I first manually created the VM and Nginx through the Azure Portal, verifying the default page. Then, I switched to Terraform, creating scripts (main.tf, variables.tf, DEV.tfvars) to automate the VM and Nginx deployment. After connecting to Azure and running Terraform commands, I confirmed a successful Nginx server setup.

**Recorded demo**: <https://youtu.be/XDVGJiONGzM>

**8.TF-Apache:** This demo compared deploying an Apache2 web server on an Azure VM. I first manually created the VM and Apache2 through the Azure Portal, verifying functionality. Then, I automated the process with Terraform. Creating scripts (main.tf, variables.tf, DEV.tfvars) defined the infrastructure and connecting to Azure allowed us to run Terraform commands for automated deployment. Finally, I confirmed a successful Apache2 server setup in the browser.

**Recorded demo**: <https://youtu.be/DgN0KNYbnzw>

**9. Mysql/VM /Image creation:** This demo demonstrated the process of moving data from a local CSV file to a MySQL database on an Azure VM. I first created Azure resources, including a storage account, container, and SAS token. Then, I deployed a Linux VM with MySQL installed and loaded the CSV data into a MySQL table. Finally, I created an Azure Compute Gallery image to share the VM configuration

**Recorded demo**: <https://youtu.be/4Dh5HeL2UtQ>

**10.Flask with Linux:**  In the demo, the process involved generating an AI prompt using ChatGPT, followed by setting up a Resource Group and creating a Virtual Machine. Next, SSH connections were configured, and Python along with Flask was installed and verified to ensure Flask was operational.

**Recorded demo**: <https://youtu.be/X2FpeW3L4pg>

**11.Nodejs with Linux:** In the demo, an AI prompt was generated with ChatGPT, and a Resource Group along with a Virtual Machine was created. SSH connections were established, and Node.js was installed and verified to ensure it's running properly.

**Recorded demo**: <https://youtu.be/e2WlHQw2r1o>

**12.Nginx with Linux:** In the demo, an AI prompt was created with ChatGPT, and a Resource Group and Virtual Machine were set up. SSH connections were configured, and NGINX was installed and verified to ensure it was operational

**Recorded demo**: <https://youtu.be/NwTBRbHNM5M>

**13.Apache2 with Ubuntu VM:**In the demo, an AI prompt was generated using Gemini, followed by the creation of a Resource Group and Virtual Machine. SSH connections were established, and the Apache2 service was verified to ensure it was running properly

**Recorded demo**: <https://youtu.be/zfJSUJ7-U68>

**14. Azure web app:** This demo showcased the creation and deletion of an Azure web app using Apache2. The web app was deployed on the Azure platform, and its functionality was verified. Subsequently, the web app was successfully removed from the Azure environment.

**Recorded demo**: <https://youtu.be/c7YxNWSj9vc>

**15. Create VM in Azure:** This demo demonstrated the creation, access, and deletion of an Azure virtual machine. A VM was created in the Azure Portal, then connected to via RDP, allowing remote access. Finally, the VM was deleted from the Azure environment.

**Recorded demo**: <https://youtu.be/--v-WBpYjtk>

**16.Handling resource:** This demo showcased the creation of Azure resource groups using different methods. Resource groups were created in the Azure Portal, through Bash commands, and using PowerShell scripts. This demonstration highlighted the flexibility in managing Azure resources.

**Recorded demo**: <https://youtu.be/xUlwcJDgsdo>