Lab 09
(Modules 15 to 17)

Instructions:

1. Log in to Azure Portal with your credentials
2. Paste all screenshots (highlighted in red) in a single Word document in the correct order
3. Name the document as **YourName-lab09**
4. Submit the document as an attachment in Bb under Labs
5. Use a WSL terminal for all activities

Lab submissions must be made by the due date (as indicated on the Critical Path). Each day thereafter will incur a **10%** deduction from the earned marks, up to a maximum of **3 days**. Submissions beyond this deadline will receive a grade of **Zero**.

Lab Objectives:

There are 5 sections in this lab as described below:

Section 1: Capture and display task output
Section 2: Work with facts (Ansible facts and package facts)
Section 3: Demonstrate the use of conditions in task execution
Section 4: Demonstrate the use of loops in task execution
Section 5: Demonstrate the use of handlers

WARNING

Code generated by ChatGPT or a similar generative AI tool, and copied and pasted without making the **right** modifications will result in a **ZERO** for that **entire section**.

Section 1

Objectives:

- Capture and display output of task execution

Part 1: Capture and display task output:

1. Make a copy of the playbook from Lab 08 section 03 that uses host variables
2. Use **register** to capture the output of the package installation task
3. Display the entire task output
4. Display the names of the packages that were installed

SCREENSHOT of the playbook and the output

========== End of Section 1 ==========

Section 2

Objectives:

- Work with facts (Ansible facts and package facts)

Part 1: Capture and display Ansible facts:

1. Write a playbook and capture facts for a single node **ansible-w-vm1**
2. Display all the facts on the screen
3. Display only the FQDN, IPv4 address, and short hostname of the node

SCREENSHOT of the playbook and all the output

Part 2: Capture and display package facts:

4. Write a playbook to list the version of a single package called **setup** from **ansible-c-vm1**

SCREENSHOT of the playbook and the output

========== End of Section 2 ==========

Section 3

Objectives:

- Demonstrate the use of conditions in task execution

Part 1: Demonstrate conditional task execution 1:

1. Write a playbook to install the **nmap** package only if it is defined in the playbook
2. Run this playbook against **ansible-c-vm1**

SCREENSHOT of the playbook and output

Part 2: Demonstrate conditional task execution 2:

1. Write a playbook to show which nodes have <**HumberID>** existed
2. Run this playbook against **linux** inventory group

SCREENSHOT of the playbook and output

Part 3: Demonstrate conditional task execution 3:

1. Write a playbook to add the string **This is my CentOS VM** to the **/tmp/conditional** file only if the RHEL version is **8.2** and the kernel version is **4.18.0-193.el8.x86_64**. Hint: use the **blockinfile** module.
2. Run this playbook against **linux** inventory group

SCREENSHOT of the playbook and output

========== End of Section 3 ==========

Section 4

Objectives:

- Demonstrate the use of loops in task execution

Part 1: Demonstrate task iteration 1:

1. Write a playbook to create 10 user accounts called **user101** to **user110** using loop and group vars. Define custom UIDs 5001 to 5010 respectively.
2. Run this playbook against **linux** group nodes

SCREENSHOT of the playbook, the group vars file, and output

Part 2: Demonstrate task iteration 2:

1. Write a playbook to use ansible_mounts to install mariadb, dracut, and snappy packages only if the / file system is mounted and has 2GB of free disk space available.
2. Use host vars to define packages
3. Run this playbook against **ansible-c-vm2**

SCREENSHOT of the playbook, the host vars file, and output

========== End of Section 4 ==========

Section 5

Objectives:

- Demonstrate the use of handlers

Part 1: Demonstrate the use of a single handler:

1. Write a playbook to install the **Apache** web server software, add hostname of the system to the **/var/www/html/index.html** file, enable the Apache service to auto-start on system reboots, and start the service.
2. Run this playbook against **ansible-c-vm1**
3. Open a browser window and enter the full DNS name of the managed node in the browser. You should see the hostname on the screen.

SCREENSHOT of the playbook, the index.html file, output of the playbook execution, and the output in the browser window.

Part 2: Demonstrate the use of multiple handlers:

1. Write a playbook to install the **Apache** web server and **mariadb-server** software, enable both Apache and mariadb services to auto-start on system reboots, and start both services.
2. Run this playbook against **ansible-c-vm2**
3. Use appropriate ad-hoc commands to display the operating state of both services.

SCREENSHOT of the playbook and outputs of the playbook execution and the ad-hoc commands

========== End of Section 5 ==========