* **Week 4 Assignment 1 - Working with a Linux VM in the cloud** *Please click on the link above to submit this week's assignment.* In this assignment, you will create a Linux VM in the cloud and then run Linux and optionally some Python commands on the newly created VM. Create a 3-4 page write-up with screen shots for this assignment. Instructions :
  + - Install PuTTY, PuttyGen and WinSCP on your Windows PC.
      * NOTE:  Mac users should use ssh instead of PuTTY and CyberDuck instead of WinSCP
      * NOTE:  If you use MobaXTerm, then you don't need to use WinSCP
    - Create a few text files with Notepad or any other text editor
    - Login to the AWS console
    - Create a Linux VM
      * For AWS, remember to download the .pem file. You will not be able to connect if you forget to download or misplace this file
      * After the VM is created, note down it's IP addresses - both private and public
      * You will need the public IP address to connect from PuTTY or ssh
      * If you are using PuTTY/MobaXTerm, don't forget to convert the .pem file to .ppk using the PuttyGen program.
    - Connect to the VM using PuTTY (or ssh) from your computer
    - Transfer the test files from your PC to the Linux VM using scp, WinScp or Mobaxterm
      * To do the inter-VM transfer, you must also transfer the .pem file to the Linux VM
    - Create a 2nd Linux VM
      * With AWS, use the same key file for the 2nd VM
      * After the VM is created, note down it's IP addresses - both private and public
      * Transfer the .pem file to the 2nd VM
      * Change permissions of the .pem file using the 'chmod' program
    - Transfer files from VM1 to VM2 using the file transfer protocol (scp or sFTP).
      * Look at <https://linuxize.com/post/how-to-use-scp-command-to-securely-transfer-files/> and  <https://linuxize.com/post/how-to-use-linux-sftp-command-to-transfer-files/>
      * With AWS VMs, you must specify the .pem file in the ftp/scp command as follows :
      * sftp  -i  yourVM.pem   Internal-ip-address of the 2nd VM
      * OR    scp -i yourVM.pem  test-file    ec2-user@IP-address-of-2nd-VM:/tmp
    - Verify that the file was copied to the 2nd VM
    - **Optional** :  Learn and practice the following Linux commands on the EC2 instance  (NOTE: This is for you to become familiar with and practice Linux commands. Run commands that operate on files and directories, view running processes, view networking ports,  find files, etc.. )
      * + ss and netstat
        + wc (word count and line count)
        + cat (to display content and also to concatenate files), head and tail commands
        + sed command  (for example, to remove the topmost line of any text file )
        + ps
        + find
        + history
        + rm, ls, mv, mkdir, cd
        + which
        + python
    - When done with the above steps, exit out of the PuTTY session(s) and stop the VMs from the AWS console
* **Requirements for the written assignments:**
  + Assignment file must have a .doc or .docx extension; screen shots should be in .jpg, .gif, or .pdf
  + Points for this assignment = 40

* **Week 4 Assignment 2 - Working with a Windows VM in the cloud** *Please click on the link above to submit this week's assignment.* In this hands-on assignment, you will create a Windows VM in the cloud and then connect to the VM using remote desktop connection.  (NOTE: Mac users should look for the RDP app in the app store). Create a 1-2 page write-up of your assignment. Provide screen shots of your work. Instructions:
  + - Create a Windows VM
      * With AWS, use the same .pem file that you used for Linux VMs
      * Once the Windows VM is running, use the .pem file to decrypt the Windows Administrator password
    - Note down the IP address and hostname of this VM
    - Connect to the VM using remote desktop connection
      * In AWS, select the Windows VM and then click on the "Connect" button.  This will download the RDP file to your PC. Double-click on the RDP file to start the RDP session
    - Start the browser and browse any website to ensure that the VM has internet connectivity
    - Transfer the files to the Windows VM via an AWS S3 bucket
      * Enable the CLI keys
      * Transfer a test file from the Linux VM you used in Week 2 assignment to S3 using the CLI
      * Then install the AWS CLI on the newly created Windows VM
      * Start the Windows command window
      * Use the CLI to download the file(s) from S3 to the Windows VM
      * Verify that the file was downloaded from S3
    - Take screen shots of your activities
    - Log off from the Windows VM
    - Stop the Windows VM
* **Requirements for the written assignments:**
  + Submit write up and screen shots for all activities by end of Week 4 .
  + Assignment file must have a .doc or .docx extension; screen shots should be in .jpg, .gif, or .pdf
  + Points for this assignment = 20

* **Week 4 Assignment 3 - Using automation to create cloud resources** *Please click on the link above to submit this week's assignment.* In this assignment you will use either the AWS EC2 CLI or the AWS Cloud Formation automation tool for working with the VMs created in previous assignments.  Choose option 1 or 2 below for your assignment.  **Option 1**:  Use the CLI to do the following :
  + Start the Linux VM and Windows VM created previously
  + Stop the running instances
  + Clone an existing Linux VM
  + https://docs.aws.amazon.com/cli/latest/userguide/cli-ec2-launch.html
  + <https://www.thegeekstuff.com/2016/04/aws-ec2-cli-examples/>
* **Option 2**:  Use CloudFormation to create/delete VMs
  + Follow the instructions given here
* *http://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/GettingStarted.Walkthrough.html*
  + Modify the sample JSON or YAML template suitably
  + Use the same Instance Type and Key Name as those used in Assignment 1
  + Run the automation script.
  + Take screen shots of the Cloud Formation screen showing the status of the script
  + Take screen shots of the AWS EC2 screen showing the running VM.
  + Ensure that you can connect to the VM using SSH.  Take screen shot of the ssh connection
  + Stop the running VMs created by the automation script
* **References**:
  + http://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/aws-properties-ec2-instance.html
  + https://s3.amazonaws.com/cloudformation-templates-us-east-1/EC2InstanceWithSecurityGroupSample.template