# RiOverview

The logical design of Active Directory (AD) includes the choice of the domain forest structure, domain trusts, and the organizational unit (OU) design. The domain forest design was implemented in a previous guided practice. In this guided practice you will explore the other AD logical components and implement a typical OU design.

# Objectives

* Manage Organizational Units
  + Create the following Active Directory objects from the GUI and command line:
    - Organizational Unit
    - User account
  + Perform the common AD object management tasks from the GUI and command line.
  + Configure multiple accounts simultaneously.
  + Use PowerShell and a CSV file to import AD objects.

## Skills Reviewed

* Creating Users in Active Directory.

## New Skills

* Creating Organizational Units in Active Directory.
* Using PowerShell with a CSV file.
* Bulk addition of users with PowerShell and a CSV file.

## References

* Active Directory Accounts - <https://docs.microsoft.com/en-us/windows/security/identity-protection/access-control/active-directory-accounts>

# Initial Conditions

* Guided Practice – Installing Active Directory is complete

# Instructions

## ORGANIZING THE USERS AND GROUPS

The **KMW.local** domain has over 400 employees and now would like to create an organizational unit structure to manage the users and groups more easily. They would like to create an organizational structure that matches their geographic (office) location and department structure.

1. Logon to your **Server-01** virtual machine using an administrative account.
2. Open a **PowerShell** prompt
3. Type**dsa.msc** to open the **Active Directory Users and Computers** management console.
4. Create an **Organizational Unit** for the **Greenville** location as follow:
   1. **Right-Click** the domain and choose **New->Organizational Unit** from the context menu.
   2. Type **Greenville** in the **Name**: textbox and click the **OK** button.
5. Repeat the previous steps to create the **Richmond** OU.
6. Create an Organizational Unit for the **Greensboro** OU in the root of the domain by typing the following in PowerShell.

***New-ADOrganizationalUnit -Name Greensboro -Path "DC=KMW,DC=local"***

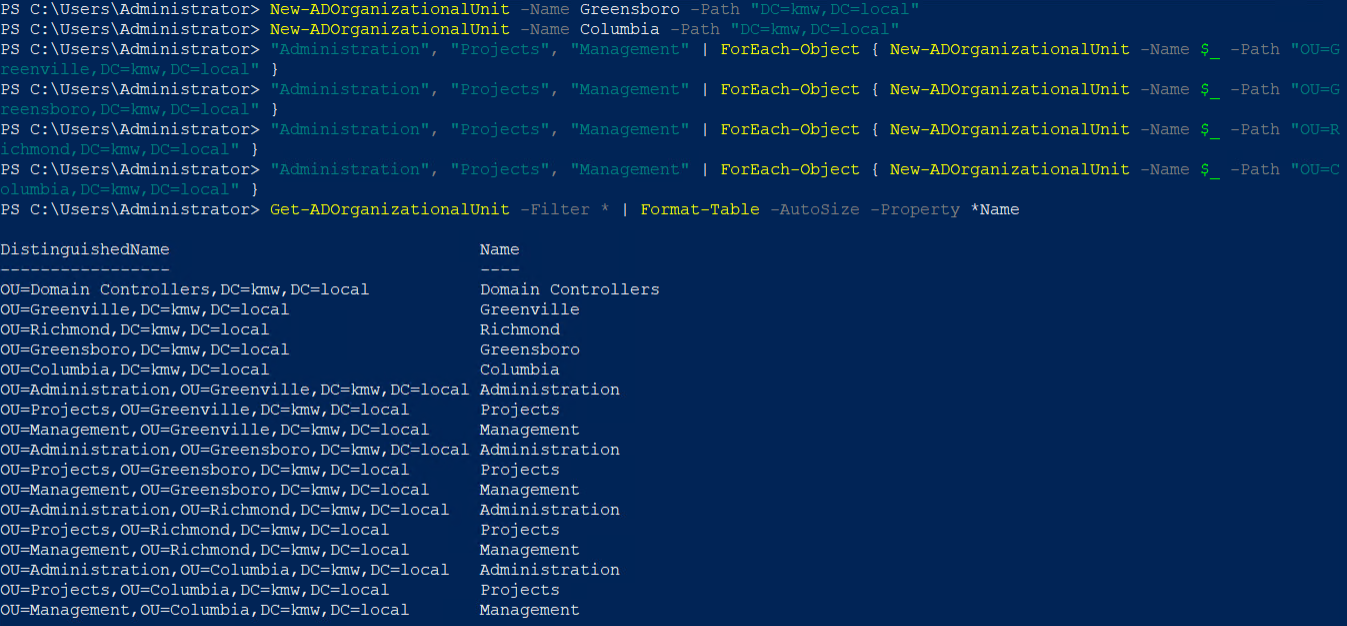
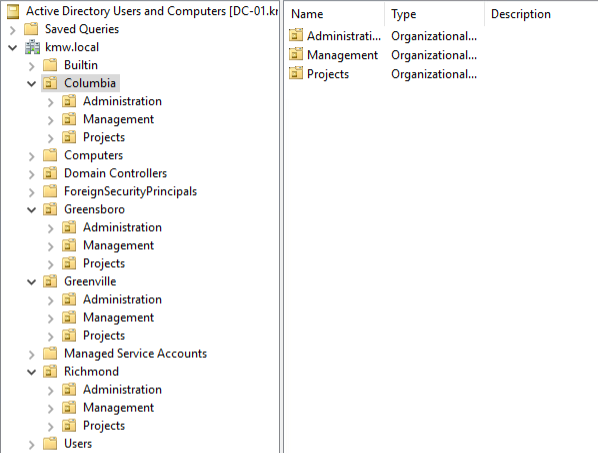
1. Repeat the step above to create the **Columbia** OU.
2. Create the **Administration**, **Projects** and **Management** OUs in the **Greenville** OU with the following command:

***"Administration", "Projects", "Management" | ForEach-Object { New-ADOrganizationalUnit -Name $\_ -Path "OU=Greenville,DC=KMW,DC=local"}***

*The command above is taking the strings (words) Administration, Projects, and Management and sending them through the pipeline one at a time. Each time the Foreach-Object command is running the command inside the curly braces {}. The string is substituted wherever the $\_ is placed.*

1. Repeat the step above to add the **Administration, Projects,** and **Management** OUs to the **Richmond**, **Greensboro,** and **Columbia** OUs.
2. Verify that the OUs have been created using the following command:

***Get-ADOrganizationalUnit -filter \* | Format-Table -AutoSize -Property DistinguishedName,Name***

1. You should see the output shown below:
2. You should see the following in the GUI.

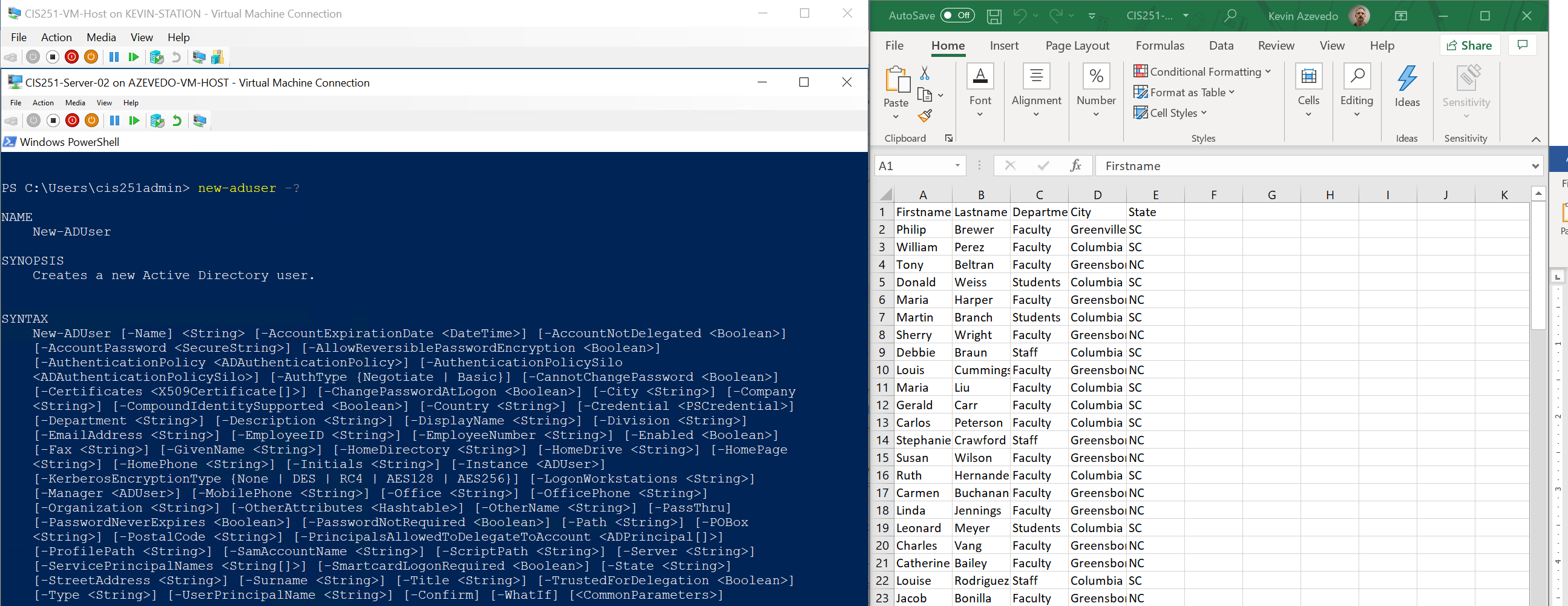
## Importing the AD Objects

The **Employees** file contains user information and will be used to import the user accounts into Active Directory. The file contains the typical information you would see if a company exports employee data to a CSV file.

In this step, you will modify a CSV file so that its field (column) names match the names of the parameter names in the **New-ADUser** cmdlet. You will then use the Import-CSV cmdlet to pipe each line to the **New-ADUser** to create a user with the parameter values in each row. This will allow you to create all 400 users in a matter of minutes, instead of hours.

### Creating the CSV file

In this step you need to modify the CSV file so that it is in the proper format. Use the following steps to do this:

1. Download the **Employees.csv** to a machine that has Excel installed. It will probably download as an Excel spreadsheet. That is okay.
2. Open the **Employees.csv** file using **Excel** on your system.
3. Openan **elevated** **PowerShell** session on the **Server-01** virtual machine.
4. Displaythe **syntax** for the **New-ADUser** cmdlet.
5. Configure the two windows so that they are on separate screens or displayed side by side as shown below.
6. The first line of the CSV file must contain the parameter name for the values that are in that column. You can determine the parameter name from the syntax statement for example, the **Firstname** column needs to be changed to **GivenName**. See if you can determine what the **LastName**, **Department**, **City**, and **State** column names need to be changed to. **Note**: After the required parameters in the syntax statement, the parameter names are sorted alphabetically.
7. Change the **columns** to match the appropriate **parameter** **names**.
8. Add the following additional columns, **SamAccountName**, **UserPrincipalname**, **Name**, **DisplayName**.
9. These attributes have no values yet and so must be calculated from other values. This can be done by using a few simple formulas in Excel and the concatenate (**&**) operator.
10. We will do this for the Pre-Windows 2000 Logon name (**SamAccountName**). In Row 2 below the **SamAccountName** column type the following formula:

=Left(A2,2) & B2

* 1. This is assuming that the first name is in the cell **A2** and the last name is in cell **B2**. This takes the value **A2** and removes a 2-character starting at the left and then concatenates (adds) it to the value in cell **B2**. This will create your common First two letters of first name + last name as the username.

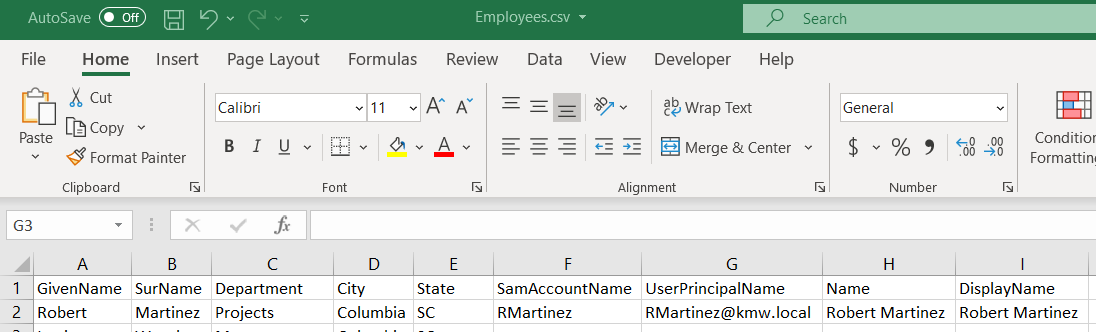
1. In the **Row 2** below both the **DisplayName** and **Name** headings, **add** the **formula** below.

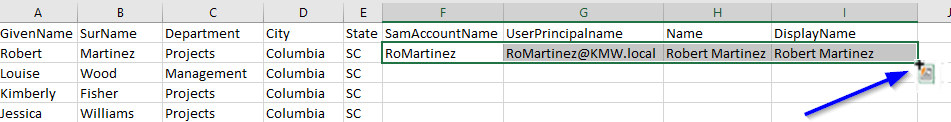
=A2 & “ “ & B2

1. The formula below will create a value for the **UserPrincipalName** attribute. Add the formula to cell **G2**.

**Note**: adjust the formula as necessary if **F2** does not contain the **samAccountName** attribute.

=F2 & “@KMW.local”

1. Your file should now look like the screen below.
2. Selectcells F2 through I2 (the ones with the formulasyou added) and **double-click** the **“+”** **sign** in the lower right corner as shown in the figure below to copy the formulas to the remaining rows.



1. Save your file as an **XLSX** document to preserve your formulas.
2. Save your file again using the CSV format as **Employees.csv**.
3. Upload your CSV file to your OneDrive account.

## Importing Active Directory Objects using the Import-CSV Command

### Import the User accounts

1. Logon to the **Server-01** virtual machine using the **KMWAdmin** account.
2. On VCastle, download your **Employees.csv** file from your ECPI One Drive folder to the desktop on **Server-01**. (If using Internet Explorer, you should turn off **IE Enhanced Security Configuration** using **Server Manager**.)
3. On your host machine copy the **Employees.csv** file and paste it to the desktop on the **Server-01** virtual machine. Rename the file **employees**.**csv**.
4. Open **PowerShell** as an **Administrator** and type the following:

Import-Csv -Path C:\Users\KMWadmin\Desktop\Employees.csv | New-ADUser -AccountPassword (ConvertTo-SecureString Password1 -AsPlainText -Force) -Enabled $true

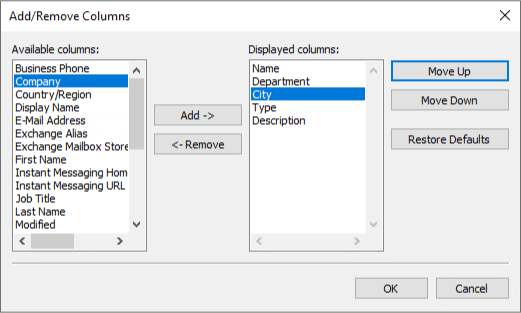
1. You should not get an output from to command.
2. To verify that the users were created you can open **AD Users and Computers** and view the new users in the **Users** folder.

**Note**: if you already have **Active Directory Users and Computers** open, you will need to refresh the view to see the new users.

## Organizing the users in Active Directory

In this step you will modify Active Directory Users and Computers so that it displays the user’s department attribute. You will then select the users from the same department and move them to the appropriate organizational units.

### Configuring AD Users and Computers to display the Department attribute

1. Logon to the **Server-01** virtual machine as the **KMWadmin**.
2. Configure the **Department** attribute to be displayed as follows:
   1. ****Open **Active Directory Users and Computers** and browse to the **Users** folder.
   2. Choose the **Add/Remove** **Columns** options from the **View** menu.
   3. Choose **Department** from the **Available columns** and click the **Add** button.
3. Verify that **Department** is **selected** in the Displayed columns and click the **Move** **Up** button until **Department is** thesecond item from the top.
4. Repeatthe step above to add the **City** column.

### Sorting Users in AD Users and Computer and Moving Multiple users simultaneously

1. In **Active Directory Users and Computers** browse to the **Users** folder and click on the **Department** column to sort the objects by department and city.
2. To move the users to their respective OUs, perform the following:
   1. Selectall the users in the **Projects** department from **Richmond.**
   2. **Right-Click** and select **Move** from the context menu.
   3. Browse to the **Richmond -> Projects OU** and click the **OK** button.
   4. Repeat for the remaining departments.
3. Repeat this for the remaininglocations.

## Allowing Users to log on through Enhanced Hyper-V session

Virtual Machine Connection (VMConnect) lets you use a computer's local resources in a virtual machine, in our case, the VM-Host machine. **Enhanced session mode** lets you resize the VMConnect window. **Enhanced session mode** also allows:

* Troubleshoot a virtual machine without a network connection to the virtual machine.
* Copy and paste files to and from the virtual machine in the same way you copy and paste using a Remote Desktop Connection (RDP).
* See [Use local resources on Hyper-V virtual machine with VMConnect](https://docs.microsoft.com/en-us/windows-server/virtualization/hyper-v/learn-more/use-local-resources-on-hyper-v-virtual-machine-with-vmconnect) for other details.

Using **Enhanced session mode** also has some baggage, it requires the user to have the right to access the computer remotely (i.e., Use an administrative account or be a member of the **Remote Desktop Users** group.) See [Connecting to a VM - Remote Desktop Users group](https://social.technet.microsoft.com/Forums/ie/en-US/2d06072e-6ee1-4bd5-9488-397bf04291fc/connecting-to-a-vm-remote-desktop-users-group) for further details.

Perform the following to allow all non-administrative users to log into the VMs.

1. In **Active Directory Users and Computers** browse to the **Users** folder.
2. Add **Domain Users** grouptothe **Remote Desktop Users** group.
3. Log onto **Client-01** as **KMWAdmin**. Add **Domain Users** to the **Remote Desktop Users Group** on **Client-01.**
4. Log onto **Client-02** as **KMWAdmin**. Add **Domain Users** to the **Remote Desktop Users Group** on **Client-02**.

## sUBMISSION rEQUIREMENTS

1. **Download** the **grading** **script** from the assignment page to the **C:\Scripts** folder.
2. Check your lab by running the following command:

Invoke-Pester -Path C:\Scripts\GP15-Implementing\_AD\_Logical \_Design.test.ps1

**Note**: You will see a security warning when running the script. Enter **R** to run the script.

If you want to see more detail, add **-Output Detailed** to the command. This may assist you with troubleshooting

Invoke-Pester -Path C:\Scripts\GP15-Implementing\_AD\_Logical \_Design.test.ps1 -Output Detailed

1. You should not see any red in the output. Red in the PowerShell way of telling you that an error condition exists. Most of the time, the output will tell you what is wrong. If it is not obvious, contact your teacher and ask for assistance. You will be learning PowerShell during this term. **Correct** any **errors** you may have and run the script until all the output has no red. You should see the output like the images below

Text

Description automatically generated

1. Capture a snippet that shows the PowerShell Command and all its output. If you must use more than one snippet to capture the output, you must have at least **one line of overlap** in the snippets. The text in the snippets **must be legible** when pasted into the Word document. Paste the snippet(s) into a **new** **Word** **document**
2. **Fill** **in** the **information** in the following table. Copy the following table into the **Word** **document** and fill in the information about all the **new** commands used in this lab (the example provided is not a new command and should be deleted):

|  |  |  |
| --- | --- | --- |
| PowerShell Commands | | |
| Command | Example | Description |
| Get-Childitem | Get-Childitem -Path C:\ | Displays the files in the C:\ directory |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

1. **Upload** the **document** in the submission area of the assignment.