**Creating Resource Group with Role Assignment**

1. [Download Code Here](https://github.com/johnpatrick23/PowerShell/blob/master/CreateResourceGroupWithRoleAssignment.ps1)
2. function Generate-FileName{
3. param($GroupCode, $username)
4. $DateNow = [DateTime]::Now
6. $Year = $DateNow.Year
7. $MonthString = $DateNow.ToString("MM")
8. $DayString = $DateNow.ToString("dddd")
9. $Day = $DateNow.Day
10. $Month = Add-Zero -Number $DateNow.Month
11. $Hour = Add-Zero -Number $DateNow.Hour
12. $Minute = Add-Zero -Number $DateNow.Minute
13. $Second = Add-Zero -Number $DateNow.Second
14. $MilliSec = Add-Zero -Number $DateNow.Millisecond
16. return "AS-$GroupCode-$username-$Year$MonthString$Day$Hour$Minute$Second$MilliSec"
17. }
18. # Adding zero to have a 2 digit number
19. function Add-Zero{
20. param($Number)
22. if($Number -le 9){
23. return '0' + $Number.ToString()
24. }else{
25. return $Number.ToString()
26. }
27. }
28. Clear-Host
29. $emails = "email1@email.com", "email2@email.com"
30. $users = "uniqueID1", "uniqueID2"
31. class UserInfo{
32. [string]$user
33. [string]$email
34. [string]$groupCode
35. }
36. $UserInfos = New-Object System.Collections.Generic.List[UserInfo]
37. $GroupCode = "SIS"
38. for($i = 0; $i -lt $users.Count; $i++){
39. $UserInfo = New-Object UserInfo
40. $UserInfo.email = $emails[$i]
41. $UserInfo.user = $users[$i]
42. $UserInfo.groupCode = $GroupCode
43. $UserInfos.Add($UserInfo)
44. }
45. $UserInfos | ForEach-Object -Process {
46. $email = $\_.email
47. $user = $\_.user
48. $groupCode = $\_.groupCode
50. Start-Job -ScriptBlock {
51. function Generate-FileName{
52. param($GroupCode, $username)
53. $DateNow = [DateTime]::Now
55. $Year = $DateNow.Year
56. $MonthString = $DateNow.ToString("MM")
57. $DayString = $DateNow.ToString("dddd")
58. $Day = $DateNow.Day
59. $Month = Add-Zero -Number $DateNow.Month
60. $Hour = Add-Zero -Number $DateNow.Hour
61. $Minute = Add-Zero -Number $DateNow.Minute
62. $Second = Add-Zero -Number $DateNow.Second
63. $MilliSec = Add-Zero -Number $DateNow.Millisecond
65. return "AS-$GroupCode-$username-$Year$MonthString$Day$Hour$Minute$Second$MilliSec"
66. }
67. # Adding zero to have a 2 digit number
68. function Add-Zero{
69. param($Number)
71. if($Number -le 9){
72. return '0' + $Number.ToString()
73. }else{
74. return $Number.ToString()
75. }
76. }
77. $resourceGroupName = Generate-FileName -GroupCode $args[2] -username $args[0]
78. if([bool](New-AzureRmResourceGroup -Name $resourceGroupName -Location "southeastasia")){
79. New-AzureRmRoleAssignment -ResourceGroupName $resourceGroupName -SignInName $args[1] -RoleDefinitionName Owner | Out-Null
80. $resourceGroupName
81. }
82. } -ArgumentList $user, $email, $groupCode;
83. Start-Sleep -s 5 # Azure gets very upset if you slam too much work at it and randomly decides not to accept the request therefore you need to put a delay with at least 5 seconds
84. }
85. Get-Job | Wait-Job | Receive-Job
86. #Get-AzureRmResourceGroup | Select-Object -Property "ResourceGroupName" | Where-Object {$\_.ResourceGroupName -like "AS-$GroupCode-\*"}
87. #(Get-AzureRmResourceGroup | Select-Object -Property "ResourceGroupName" | Where-Object {$\_.ResourceGroupName -like "AS-$GroupCode-\*"}).Count