

```
1 #update-CancerCenterDL.ps1
2 #Written by: Jeff Brusoe
3 #Last Updated: January 20, 2021
4 #
5 #Purpose:
6
7 [CmdletBinding()]
8 param (
9     [ValidateNotNullOrEmpty()]
10     [string]$CCDepartmentMap = "$PSScriptRoot\CancerCenterDepartmentMap.csv"
11 )
12
13 #Configure environment
14 try {
15     Set-HSCEnvironment -ErrorAction Stop
16
17     Write-Output "Department Map: $CCDepartmentmap"
18 }
19 catch {
20     Write-Warning "Unable to configure environment"
21 }
22
23 #Read mapping file
24 try {
25     $CCDepartments = Import-Csv $CCDepartmentMap -ErrorAction Stop
26
27     Write-Output "Department Map:"
28     Write-Output $CCDepartments
29 }
30 catch {
31     Write-Warning "Unable to read department map"
32     Invoke-HSCExitCommand -ErrorAction Stop
33 }
34
35 foreach ($CCDepartment in $CCDepartments)
36 {
37     try {
38         $DepartmentName = $CCDepartment.DepartmentName
39         Write-Output $("Department Name: $DepartmentName")
40         Write-Output $("URL: " + $CCDepartment.URL)
41
42         $DepartmentListFile = "$PSScriptRoot\GeneratedLists\" +
43             (Get-Date -Format yyyy-MM-dd) +
44             "-$DepartmentName.txt"
45
46         Write-Output "`nDepartment List File: $DepartmentListFile"
47
48         New-Item -ItemType file -Path $DepartmentListFile -ErrorAction Stop -Force
49     }
50     catch {
51         Write-Warning "Unable to generate department list file"
52     }
53
54     try {
55         $CancerCenterDLParams = @{
56             URL = $CCDepartment.URL
57             EmailSeparator = $CCDepartment.EmailSeparator
58             ErrorAction = "Stop"
59         }
60     }
```

```
61     $DLMembers = Get-CCDLMember @CancerCenterDLParams
62
63     Add-Content -Path $DepartmentListFile -Value $DLMembers -ErrorAction Stop
64 }
65 catch {
66     Write-Warning "Error calling Get-CCDLMember"
67 }
68 }
69
70 #Everything below here is Joel George Code
71 <#
72 #Clinic_WVUHS.ps1
73 #RadOnc.ps1 *
74 #Prete.ps1 *
75 #AdminMBRCC.ps1
76 #CCB_Faculty.ps1
77 #CCB_Students.ps1
78 #MembersAll.ps1
79
80
81 echo "Adding Extras for Laura List"
82 $ExtraLauraPeople = get-content \\hs-
mbrcc\mbrcc\DistroLists\ListsToEdit\Extras_for_Laura_List.txt
83 $AllinOneLauraList += $ExtraLauraPeople
84
85
86
87 echo "Adding CPC Users"
88 & "$myDir\Prete.ps1"
89 $preteOutput = $preteOutput
90 $AllinOneLauraList += $preteOutput
91
92
93 echo "Adding MBRCC Labs Users"
94 & "$myDir\LabsMBRCC.ps1"
95 $mbrccLabsOutput = $mbrccLabsOutput
96 $AllinOneLauraList += $mbrccLabsOutput
97
98
99
100
101
102
103
104
105
106 foreach ($j in $AllinOneLauraList){
107
108     if (($j -notlike "lgibson@hsc") -and ($j -notlike "*goldberg@hsc") -and ($j -
notlike "*wallis.marsh@hsc") -and ($j -notlike "*seftor@hsc") -and ($j -notlike
"*nurkiewicz") -and ($j -notlike "* *")){
109         $AllinOneLauraListClean += @($j)
110     }
111 }
112 $AllInOneLauraListClean2 += @($AllInOneLauraListClean | sort | select -Unique)
113 $OutputLists += $AllInOneLauraListClean2
114 $OutputFileName = "Disease Teams Science Exchange Email List Created on "
115 $comboList ++
116 }
117
```

```
118 if ($choice -eq 'Clinical Trials'){echo "Adding Clinical Research Users"
119 & "$myDir\CTRU.ps1"
120 $ctrOutput = $ctrOutput
121 #####echo " "
122 #####Echo "CTRU Users added to Output script"
123 #####echo " "
124 $OutputLists += $ctrOutput
125 $OutputFileName = "CRU Email List Created on "
126 $comboList ++
127 }
128
129 if ($choice -eq 'Hematology Oncology Section'){echo "Adding Hematology Oncology
Section Users"
130 & "$myDir\HemOnc.ps1"
131 $hemOncOutput = $hemOncOutput
132 #####echo " "
133 #####Echo "HemOnc Users added to Output script"
134 #####echo " "
135 $OutputLists += $hemOncOutput
136 $OutputFileName = "HemOnc Section Email List Created on "
137 $comboList ++
138 }
139
140 if ($choice -eq 'MBRCC Labs'){echo "Adding MBRCC Labs Users"
141 & "$myDir\LabsMBRCC.ps1"
142 $mbrccLabsOutput = $mbrccLabsOutput
143 #####echo " "
144 #####Echo "MBRCC Labs Users added to Output script"
145 #####echo " "
146 $OutputLists += $mbrccLabsOutput
147 $OutputFileName = "MBRCC Labs Email List Created on "
148 $comboList ++
149 }
150
151 if ($choice -eq 'MBRCC Members'){echo "Adding MBRCC Members"
152 $MembersCalled = "Yes"
153 $Global:MembersCalled2 = $MembersCalled
154 & "$myDir\MembersALL.ps1"
155 $membersAllOutput = $membersAllOutput
156 #####echo " "
157 #####Echo "MBRCC Members added to Output script"
158 #####echo " "
159 $OutputLists += $membersAllOutput
160 $OutputFileName = "MBRCC Members Email List Created on "
161 $comboList ++
162 }
163
164 if ($choice -eq 'MBRCC Members Potential'){echo "Adding Potential MBRCC Members"
165 $MembersCalled = "Yes"
166 $Global:MembersCalled2 = $MembersCalled
167 & "$myDir\MembersPotential.ps1"
168 $membersAllOutput = $membersPotentialOutput
169 #####echo " "
170 #####Echo "MBRCC Members added to Output script"
171 #####echo " "
172 $OutputLists += $membersAllOutput
173 $OutputFileName = "Potential MBRCC Members Email List Created on "
174 $comboList ++
175 }
176
```

```
177
178
179 if ($choice -eq 'Tina: List for Laura'){echo "Creating Tina: List for Laura"
180 echo " "
181 $MembersCalled = "No"
182 $Global:MembersCalled2 = $MembersCalled
183 echo "Adding MBRCC Members"
184 & "$myDir\MembersALL.ps1"
185 $AllInOneLauraList += $membersAllOutput
186 echo "Adding Extras for Laura List"
187 $ExtraLauraPeople = get-content \\hs-
    mbrcc\mbrcc\DistroLists\ListsToEdit\Extras_for_Laura_List.txt
188 $AllinOneLauraList += $ExtraLauraPeople
189 foreach ($j in $AllinOneLauraList){
190
191 if (($j -notlike "lgibson@hsc*")){
192 $AllinOneLauraListClean += @($j)
193 }
194 }
195 $AllInOneLauraListClean2 += @($AllInOneLauraListClean | sort | select -Unique)
196 $OutputLists += $AllInOneLauraListClean2
197
198 $OutputFileName = "List for Laura Created on "
199 $comboList ++
200 }
201
202
203 if ($choice -eq 'Everyone'){echo "Creating Everyone List"
204 echo " "
205 $MembersCalled = "No"
206 $Global:MembersCalled2 = $MembersCalled
207 #####echo "Adding Clinic Users"
208 #& "$myDir\Clinic.ps1"
209 Disconnect-QADService -Service 'wvuh.wvuhs.com'
210 #echo "Adding BPBCC Users"
211 #& "$myDir\BPBCC.ps1"
212 #echo "Adding Bonnies Bus Users"
213 #& "$myDir\BonniesBus.ps1"
214 #echo "Adding CCB Users"
215 #& "$myDir\CCB.ps1"
216 echo "Adding Clinic Users"
217 #& "$myDir\Clinic.ps1"
218 & "$myDir\Clinic_wvuhs.ps1"
219 echo "Adding CTRU Users"
220 & "$myDir\CTRU.ps1"
221 echo "Adding HemOnc Users"
222 & "$myDir\HemOnc.ps1"
223 echo "Adding MBRCC Admininstration Users"
224 & "$myDir\AdminMBRCC.ps1"
225 echo "Adding MBRCC Labs Users"
226 & "$myDir\LabsMBRCC.ps1"
227 echo "Adding MBRCC Members"
228 & "$myDir\MembersALL.ps1"
229 echo "Adding Prete Users"
230 & "$myDir\Prete.ps1"
231 echo "Adding EmployeeFile Users"
232 & "$myDir\EmployeeFile.ps1"
233 #echo "Adding RadOnc Users"
234 #& "$myDir\RadOnc.ps1"
235 #####& "$myDir\ExtraUsers.ps1"
```

```

236 $AllInOneList = @()
237 $AllInOneList += $clinicOutput
238 $AllInOneList += $clinicOutputwvuhs
239 $AllInOneList += $bpbccOutput
240 $AllInOneList += $radOncOutput
241 $AllInOneList += $BonniesBusOutput
242 $AllInOneList += $ccbOutput
243 $AllInOneList += $ctrOutput
244 $AllInOneList += $hemOncOutput
245 $AllInOneList += $preteOutput
246 $AllInOneList += $mbrccLabsOutput
247 $AllInOneList += $mbrccAdminOutput
248 $AllInOneList += $membersAllOutput
249 $AllInOneList += $EmployeeFileOutput
250 $AllInOneList += $allDirectoryUsers
251 $AllInOneList = $AllInOneList | sort | select -Unique
252 $comboList ++
253 $OutputFileName = "Everyone Email List Created on "
254 foreach ($j in $AllInOneList){
255
256 if (($j -notlike "cnwatson@hsc*") -and ($j -notlike "colendac@wvu*") -and ($j -
notlike "mfarrugia@hsc*") -and ($j -notlike "fontanat@wvu*") -and ($j -notlike ";")
-and ($j -notlike "seiranmanesh@hsc*") -and ($j -notlike "bjones@hsc*") -and ($j -
notlike "smarkwell@hsc*") -and ($j -notlike "bmoses2@hsc*") -and ($j -notlike
"sbsharma@hsc*") -and ($j -notlike "akanate-hsc@*") -and ($j -notlike "jaddison@*")
-and ($j -notlike "jzheng@*") -and ($j -notlike "peckc@wvu*") -and ($j -notlike
"skennedy@hsc*") -and ($j -notlike "sonikpreet.aulakh.m@wvumedicine.org;") -and ($j
-notlike "pius.joseph@cdc.gov;")){
257 $AllInOneListClean += @($j)
258 }
259 }
260 ##echo " "
261 ##$output = "There are " + $AllInOneListClean.count + " users from all lists
combined"
262 ##echo $output
263 echo " "
264 }
265
266
267
268 if ($choice -eq 'eNews List'){echo "Creating eNews List"
269 echo " "
270 $MembersCalled = "No"
271 $Global:MembersCalled2 = $MembersCalled
272 #####echo "Adding Clinic Users"
273 #& "$myDir\Clinic.ps1"
274 Disconnect-QADService -Service 'wvuh.wvuhs.com'
275 #echo "Adding BPBCC Users"
276 #& "$myDir\BPBCC.ps1"
277 #echo "Adding Bonnie's Bus Users"
278 #& "$myDir\Bonnie'sBus.ps1"
279 #echo "Adding CCB Users"
280 #& "$myDir\CCB.ps1"
281 echo "Adding Clinic Users"
282 #& "$myDir\Clinic.ps1"
283 & "$myDir\Clinic_wvuhs.ps1"
284 echo "Adding CTRU Users"
285 & "$myDir\CTRU.ps1"
286 echo "Adding HemOnc Users"
287 & "$myDir\HemOnc.ps1"

```

```

288 echo "Adding MBRCC Admininstration Users"
289 & "$myDir\AdminMBRCC.ps1"
290 echo "Adding MBRCC Labs Users"
291 & "$myDir\LabsMBRCC.ps1"
292 echo "Adding MBRCC Members"
293 & "$myDir\MembersALL.ps1"
294 echo "Adding Prete Users"
295 & "$myDir\Prete.ps1"
296 echo "Adding EmployeeFile Users"
297 & "$myDir\EmployeeFile.ps1"
298 #echo "Adding RadOnc Users"
299 #& "$myDir\RadOnc.ps1"
300 #####& "$myDir\ExtraUsers.ps1"
301 echo "Adding Extra eNews Users"
302 & "$myDir\Extras_for_eNews.ps1"
303 $OutsideNewsPeople = get-content \\hs-
mbrcc\mbrcc\DistroLists\ListsToEdit\addOutsideNewsPeople.txt
304 $InsideNewsPeople = get-content \\hs-
mbrcc\mbrcc\DistroLists\ListsToEdit\addInsideNewsPeople.txt
305 $AllInOneList = @()
306 $AllInOneList += $clinicOutput
307 $AllInOneList += $clinicOutputwvuhs
308 $AllInOneList += $bpbccOutput
309 $AllInOneList += $radOncOutput
310 $AllInOneList += $BonniesBusOutput
311 $AllInOneList += $ccbOutput
312 $AllInOneList += $ctruOutput
313 $AllInOneList += $hemOncOutput
314 $AllInOneList += $preteOutput
315 $AllInOneList += $mbrccLabsOutput
316 $AllInOneList += $mbrccAdminOutput
317 $AllInOneList += $membersAllOutput
318 $AllInOneList += $allDirectoryUsers
319 $AllInOneList += $eNewsExtraUsers
320 $AllInOneList += $OutsideNewsPeople
321 $AllInOneList += $InsideNewsPeople
322 $AllinOneList += $EmployeeFileOutput
323 $AllInOneList = $AllInOneList | sort | select -Unique
324 #}
325
326 $OutputFileName = "eNews Email List Created on "
327 foreach ($j in $AllinOneList){
328
329 if (($j -notlike "cnwatson@hsc*") -and ($j -notlike "colendac@wvu*") -and ($j -
notlike "mfarrugia@hsc*") -and ($j -notlike "fontanat@wvu*") -and ($j -notlike
"ihare@hsc*") -and ($j -notlike ";") -and ($j -notlike "seiranmanesh@hsc*") -and
($j -notlike "bjones@hsc*") -and ($j -notlike "smarkwell@hsc*") -and ($j -notlike
"bmoses2@hsc*") -and ($j -notlike "sbsharma@hsc*") -and ($j -notlike "wslone@hsc*")
-and ($j -notlike "dvanderbilt@hsc*") -and ($j -notlike "ewalk@hsc*") -and ($j -
notlike "akanate-hsc*") -and ($j -notlike "jaddison@*") -and ($j -notlike
"jzheng@*") -and ($j -notlike "peckc@wvu*") -and ($j -notlike "skennedy@hsc*") -and
($j -notlike "sonikpreet.aulakh.m@wvumedicine.org*") -and ($j -notlike
"pius.joseph@cdc.gov;")){
330 $AllinOneListClean += @($j)
331 }
332 }
333 #####echo $AllinOneListClean.count
334
335 #####echo " "

```

```

336 #####$output = "There are " + $AllInOneListClean.count + " eNews users from all
    lists combined"
337 #####echo $output
338
339 echo " "
340 $comboList ++}
341
342
343
344
345 $OutputLists += $AllInOneListClean
346 if ($comboList -eq 0){$OutputFileName = "eNews Email List Created on "}
347 }
348
349 #####Write-Host "Press any key to create the output file(s) ..."
350 #####$x = $host.UI.RawUI.ReadKey("NoEcho,IncludeKeyDown")
351 $OutputLists = $OutputLists | sort | select -Unique
352
353
354 #####Splitting the lists into WVUH and NonWVUH Users#####
355
356 if (($choice -eq 'eNews List') -or ($choice -eq 'Everyone')){
357
358
359
360 foreach ($NonWVUHUser in $OutputLists){
361 if (($NonWVUHUser -notlike "*@wvumed*") -and ($NonWVUHUser -notlike
    "@wvuhealth*)) {
362 #echo "Found one"
363 #echo $NonWVUHUSER
364 $OutputListsNonWVUHUsers += @($NonWVUHUser)
365 }
366 }
367 #pause
368 #echo "*****"
369 #echo $OutputListsNonWVUHUsers
370 #echo $OutputListsNonWVUHUsers.count
371 #pause
372
373 #echo "I am outside the loop"
374 foreach ($WVUHUser in $OutputLists){
375 if (($WVUHUser -like "*@wvumed*") -or ($WVUHUser -like "@wvuhealth*)) {
376 #echo "Found WVUH User"
377 #echo $WVUHUSER
378 $OutputListsWVUHUsers += @($WVUHUser)
379 }
380 }
381 #pause
382 #echo "*****"
383 #echo $OutputListsWVUHUsers
384 #echo $OutputListsWVUHUsers.count
385 #Pause
386
387
388 if ($choice -eq 'eNews List'){
389 $OutputFileName = "eNews Email List Created on "
390 $eNewsWVUHUsers = "Yes"
391 $eNewsNonWVUHUsers = "Yes"
392 echo "Creating eNews List"
393 echo " "

```

```

394 }
395
396 if ($choice -eq 'Everyone'){
397 $OutputFileName = "Everyone Email List Created on "
398 $EveryoneWVUHUsers = "Yes"
399 $EveryoneNonWVUHUsers = "Yes"
400 echo "Creating Everyone List"
401 echo " "
402 }
403
404 }
405
406 #Creating the output files
407 echo " "
408 echo "*****"
409 echo "Beginning Output Process"
410 echo "*****"
411 echo " "
412 #echo " "
413 # $OutputFileName = Read-Host "Please name the output file(s). No extensions
needed."
414
415 if ($comboList -gt 1){
416 $OutputFileName = "MultiGroup Email List Created on "
417 }
418
419 echo " "
420 echo "*****"
421 echo "Do Not Open lists until prompted to close the window"
422 echo "*****"
423 echo " "
424 echo " "
425 $output = $MyName.GivenName + ", there are a total of " + $OutputLists.count + "
users from your selected lists"
426 echo $output
427 if (($choice -eq 'eNews List') -or ($choice -eq 'Everyone')){
428
429 echo " "
430 echo " "
431 echo "*****"
432 echo "Generating Lists for WVUH Users and Non-WVUH users"
433 echo "*****"
434 echo " "
435 echo " "
436 $StuffToSay = "There are " + $OutputListsWVUHUsers.count + " WVUH Users and " +
$OutputListsNonWVUHUsers.count + " Non-WVUH Users"
437 echo $StuffToSay
438 }
439
440 echo " "
441
442 if (($choice -ne 'eNews List') -and ($choice -ne 'Everyone')){
443 if ($OutputLists.count -gt 475){
444
445 $split = 475
446 echo " "}
447 Else {$split = 475}
448 $i=1
449 $part=1

```



```
450 $FilePath = $LogFilePath + " Part " + $part + " " + $OutputFileName + $Date + "
    for " + $MyName.GivenName + ".txt"
451 $FilePathdelete = $LogFilePath + $OutputFileName + $Date + ".*"
452 IF (Test-Path $filepath){
453     Remove-Item $filepathdelete
454 }
455
456 if ($OutputLists.count -lt $split){
457     $FilePath = $LogFilePath + $OutputFileName + $Date + ".txt"
458     echo " "
459     echo "Adding Users to the output file(s)"
460     ##IF (Test-Path $filepath){
461     ## Remove-Item $filepathdelete
462     ##}
463
464     #####If statement from Line 546 was here#####
465     foreach ($p in $OutputLists)
466     {
467         $output = $p
468         if ($i -gt $split){
469             $part++
470             $FilePath = $LogFilePath + " Part " + $part + " " + $OutputFileName + $Date + " for
                " + $MyName.GivenName + ".txt"
471             $screenoutput = "Adding Users to part " + $part + " of the output files"
472             echo $screenoutput
473             ##IF (Test-Path $filepath){
474             ## Remove-Item $filepath
475             ##}
476             $i=0
477         }
478         Add-content $filepath $output
479         $i++
480     }
481     echo " "
482     echo " "
483     $outputlocation = "Output has been saved to " + $LogFilePath
484     echo $outputlocation
485     echo " "
486     echo " "
487     $ScriptFinished = "Script finished. There are " + $part + " files this week."
488     echo $ScriptFinished
489     #echo $combolist
490 }
491
492 if (($choice -eq 'eNews List') -or ($choice -eq 'Everyone')){
493
494     #####
495     if ($choice -eq 'eNews List'){
496
497         if ($eNewsWVUHUsers = "Yes"){
498
499             $OutputLists = $OutputListswVUHUsers
500             $OutputFileName = "eNews List (WVUH Users) Created on "
501
502
503             if ($OutputLists.count -gt 480){
504
505                 $split = read-host "How many users do you want to include on each email? Our
                    suggestion is no more than 300 per email"
506                 echo " "}
```

```
507 Else {$split = 480}
508 $i=1
509 $part=1
510 $FilePath = $LogFilePath + " Part " + $part + " " + $OutputFileName + $Date + "
for " + $MyName.GivenName + ".txt"
511 $FilePathdelete = $LogFilePath + $OutputFileName + $Date + " for " +
$MyName.GivenName + " *.*"
512 IF (Test-Path $filepath){
513     Remove-Item $filepathdelete
514 }
515
516 if ($OutputLists.count -lt $split){
517 $FilePath = $LogFilePath + $OutputFileName + $Date + " for " + $MyName.GivenName +
".txt"}
518 echo " "
519
520
521
522 foreach ($p in $OutputLists)
523 {
524 $output = $p
525 if ($i -gt $split){
526 $part++
527 $FilePath = $LogFilePath + " Part " + $part + " " + $OutputFileName + $Date + " for
" + $MyName.GivenName + ".txt"
528 $screenoutput = "Adding WVUH Users to part " + $part + " of the output files"
529 echo $screenoutput
530 ##IF (Test-Path $filepath){
531 ## Remove-Item $filepath
532 ##}
533 $i=0
534 }
535 Add-content $filepath $output
536 $i++
537 }
538
539
540
541 $eNewsWVUHUsers = "No"
542 }
543
544
545 if ($eNewsNonWVUHUsers ="Yes"){
546
547
548 $OutputLists = $OutputListsNonWVUHUsers
549 $OutputFileName = "eNews List (Non-WVUH Users) Created on "
550
551 if ($OutputLists.count -gt 480){
552
553 $split = read-host "How many users do you want to include on each email? Our
suggestion is no more than 300 per email"
554 echo " "}
555 Else {$split = 480}
556 $i=1
557 $part=1
558 $FilePath = $LogFilePath + " Part " + $part + " " + $OutputFileName + $Date + "
for " + $MyName.GivenName + ".txt"
559 $FilePathdelete = $LogFilePath + $OutputFileName + $Date + " for " +
$MyName.GivenName + " *.*"
```

```
560 IF (Test-Path $filepath){
561     Remove-Item $filepathdelete
562 }
563
564 if ($OutputLists.count -lt $split){
565     $FilePath = $LogFilePath + $OutputFileName + $Date + " for " + $MyName.GivenName +
566         ".txt"}
567     echo " "
568
569
570
571 foreach ($p in $OutputLists)
572 {
573     $output = $p
574     if ($i -gt $split){
575         $part++
576         $FilePath = $LogFilePath + " Part " + $part + " " + $OutputFileName + $Date + " for
577             " + $MyName.GivenName + ".txt"
578         $screenoutput = "Adding Non-WVUH Users to part " + $part + " of the output files"
579         echo $screenoutput
580         ##IF (Test-Path $filepath){
581         ##    Remove-Item $filepath
582         ##}
583         $i=0
584     }
585     Add-content $filepath $output
586     $i++
587 }
588
589
590 $eNewsNonWVUHUsers = "No"
591 }
592
593
594
595
596
597
598 $part = "2"
599
600 }
601 #####
602 if ($choice -eq 'Everyone'){
603
604     if ($EveryoneWVUHUsers ="Yes"){
605
606         $OutputLists = $OutputListswVUHUsers
607         $OutputFileName = "Everyone List (WVUH Users) Created on "
608
609
610         if ($OutputLists.count -gt 480){
611
612             $split = read-host "How many users do you want to include on each email? Our
613                 suggestion is no more than 300 per email"
614             echo " "}
615             Else {$split = 480}
616             $i=1
617             $part=1
```

```
617 $FilePath = $LogFilePath + " Part " + $part + " " + $OutputFileName + $Date + "
    for " + $MyName.GivenName + ".txt"
618 $FilePathdelete = $LogFilePath + $OutputFileName + $Date + ".*"
619 IF (Test-Path $filepath){
620     Remove-Item $filepathdelete
621 }
622
623 if ($OutputLists.count -lt $split){
624 $FilePath = $LogFilePath + $OutputFileName + $Date + " for " + $MyName.GivenName +
    ".txt"}
625 echo " "
626
627
628
629 foreach ($p in $OutputLists)
630 {
631 $output = $p
632 if ($i -gt $split){
633 $part++
634 $FilePath = $LogFilePath + " Part " + $part + " " + $OutputFileName + $Date + " for
    " + $MyName.GivenName + ".txt"
635 $screenoutput = "Adding WVUH Users to part " + $part + " of the output files"
636 echo $screenoutput
637 ##IF (Test-Path $filepath){
638 ## Remove-Item $filepath
639 ##}
640 $i=0
641 }
642 Add-content $filepath $output
643 $i++
644 }
645
646
647
648 $EveryoneWVUHUsers = "No"
649 }
650
651
652 if ($EveryoneNonWVUHUsers ="Yes"){
653
654
655 $OutputLists = $OutputListsNonWVUHUsers
656 $OutputFileName = "Everyone List (Non-WVUH Users) Created on "
657
658 if ($OutputLists.count -gt 480){
659
660 $split = read-host "How many users do you want to include on each email? Our
    suggestion is no more than 300 per email"
661 echo " "}
662 Else {$split = 480}
663 $i=1
664 $part=1
665 $FilePath = $LogFilePath + " Part " + $part + " " + $OutputFileName + $Date + "
    for " + $MyName + ".txt"
666 $FilePathdelete = $LogFilePath + $OutputFileName + $Date + ".*"
667 IF (Test-Path $filepath){
668     Remove-Item $filepathdelete
669 }
670
671 if ($OutputLists.count -lt $split){
```

```
672 $FilePath = $LogFilePath + $OutputFileName + $Date + " for " + $MyName.GivenName +  
    ".txt"}  
673 echo " "  
674  
675  
676  
677  
678 foreach ($p in $OutputLists)  
679 {  
680 $output = $p  
681 if ($i -gt $split){  
682 $part++  
683 $FilePath = $LogFilePath + " Part " + $part + " " + $OutputFileName + $Date + " for  
    " + $MyName + ".txt"  
684 $screenoutput = "Adding Non-WVUH Users to part " + $part + " of the output files  
    for " + $MyName  
685 echo $screenoutput  
686 ##IF (Test-Path $filepath){  
687 ## Remove-Item $filepath  
688 ##}  
689 $i=0  
690 }  
691 Add-content $filepath $output  
692 $i++  
693 }  
694  
695  
696  
697 $EveryoneNonWVUHUsers = "No"  
698 }  
699  
700  
701  
702  
703  
704  
705 $part = "2"  
706  
707 }  
708  
709  
710  
711 $outputlocation = "Output has been saved to " + $LogFilePath  
712 echo $outputlocation  
713 echo " "  
714 echo " "  
715 $ScriptFinished = "Script finished. There are " + $part + " files this week."  
716 echo $ScriptFinished  
717 #echo $combolist  
718  
719 #####  
720 }  
721 echo " "  
722 Write-Host "Press any key to close this window"  
723 ii $LogFilePath  
724 #$x = $host.UI.RawUI.ReadKey("NoEcho,IncludeKeyDown")  
725  
726 #>
```