# To run the program continuously

While ($X -ne "X") {

#================= Excel Module =======================

# Get information about Excel Process is running or not

$Excel = Get-Process "EXCEL" -ErrorAction SilentlyContinue

#================= PowerShell Module =======================

# Get information about PowerShell Process is running or not

$PowerShell = Get-Process "powershell" -ErrorAction SilentlyContinue

#================= Word Module =======================

# Get information about Word Process is running or not

$Word = Get-Process "WINWORD" -ErrorAction SilentlyContinue

#================= PowerPoint Module =======================

# Get information about Excel Process is running or not

$PowerPoint = Get-Process "POWERPNT" -ErrorAction SilentlyContinue

#================= Email Module =======================

$EmailFrom = “dissertation.test2021@outlook.com”

$EmailTo = "dissertation.test2022@outlook.com”

$Subject = “Notification for CERBERUS”

$Body = “Dear Admin,

The Host: $env:computername, has a severe issue with Office Document.

Take immediate action.

Regards,

CERBERUS”

# If - condition starts here for Excel and PowerShell

if

(

    # Check Excel and PowerShell -both the process is running or not

    ($Excel) -and ($PowerShell) -and (!$PowerShell.HasExited)

)

    {

   # Close both the process

   $Excel | Stop-Process -Force

   $PowerShell | Stop-Process -Force

   # Get the all the process and save in a file

   Get-Process | Out-File -FilePath C:\Process\_list.txt

   # Write the process name in a file

   "Macro enable Excel file was open", "PowerShell is on execution" | Out-File C:\Reason.txt

   # SMTP acess for sending Email

   $SMTPServer = “smtp.outlook.com”

   $SMTPClient = New-Object Net.Mail.SmtpClient($SmtpServer, 587)

   # Sending Email notification

   $SMTPClient.EnableSsl = $true

   $SMTPClient.Credentials = New-Object System.Net.NetworkCredential(“dissertation.test2021@outlook.com”, “dissertationtest2021”);

   $SMTPClient.Send($EmailFrom, $EmailTo, $Subject, $Body)

# Delay for sending email

sleep 1

#Trigger shutdown systems

Stop-Computer

}

# If - condition starts here for Word and PowerShell

elseif

(

    # Check Word and PowerShell -both the process is running or not

    ($Word) -and ($PowerShell) -and (!$PowerShell.HasExited)

)

   {

    # Close both Word and PowerShell process

    $Word | Stop-Process -Force

    $PowerShell | Stop-Process -Force

    # Get the all the process and save in a file

    Get-Process | Out-File -FilePath C:\Process\_list.txt

    # Write the process name in a file

    "Macro enable Word file was open", "PowerShell is on execution" | Out-File C:\Reason.txt

    # SMTP acess for sending Email

    $SMTPServer = “smtp.outlook.com”

    $SMTPClient = New-Object Net.Mail.SmtpClient($SmtpServer, 587)

    # Sending Email notification

    $SMTPClient.EnableSsl = $true

    $SMTPClient.Credentials = New-Object System.Net.NetworkCredential(“dissertation.test2021@outlook.com”, “dissertationtest2021”);

    $SMTPClient.Send($EmailFrom, $EmailTo, $Subject, $Body)

# Delay for sending email

sleep 1

#Trigger shutdown systems

Stop-Computer

  }

# If - condition starts here for PowerPoint and PowerShell

else

{

(

    # Check PowerPoint and PowerShell -both the process is running or not

   ($PowerPoint) -and ($PowerShell) -and (!$PowerShell.HasExited)

)

   {

   # Close both PowerPoint and PowerShell process

   $PowerPoint | Stop-Process -Force

   $PowerShell | Stop-Process -Force

   # Get the all the process and save in a file

   Get-Process | Out-File -FilePath C:\Process\_list.txt

   # Write the process name in a file

   "Macro enable PowerPoint file was open", "PowerShell is on execution" | Out-File C:\Reason.txt

   # SMTP acess for sending Email

   $SMTPServer = “smtp.outlook.com”

   $SMTPClient = New-Object Net.Mail.SmtpClient($SmtpServer, 587)

   # Sending Email notification

   $SMTPClient.EnableSsl = $true

   $SMTPClient.Credentials = New-Object System.Net.NetworkCredential(“dissertation.test2021@outlook.com”, “dissertationtest2021”);

   $SMTPClient.Send($EmailFrom, $EmailTo, $Subject, $Body)

# Delay for sending email

sleep 1

#Trigger shutdown systems

Stop-Computer

   }

   }

   }