Este artigo aplica-se apenas a integrações via query SQL e Disparo de e-mail.

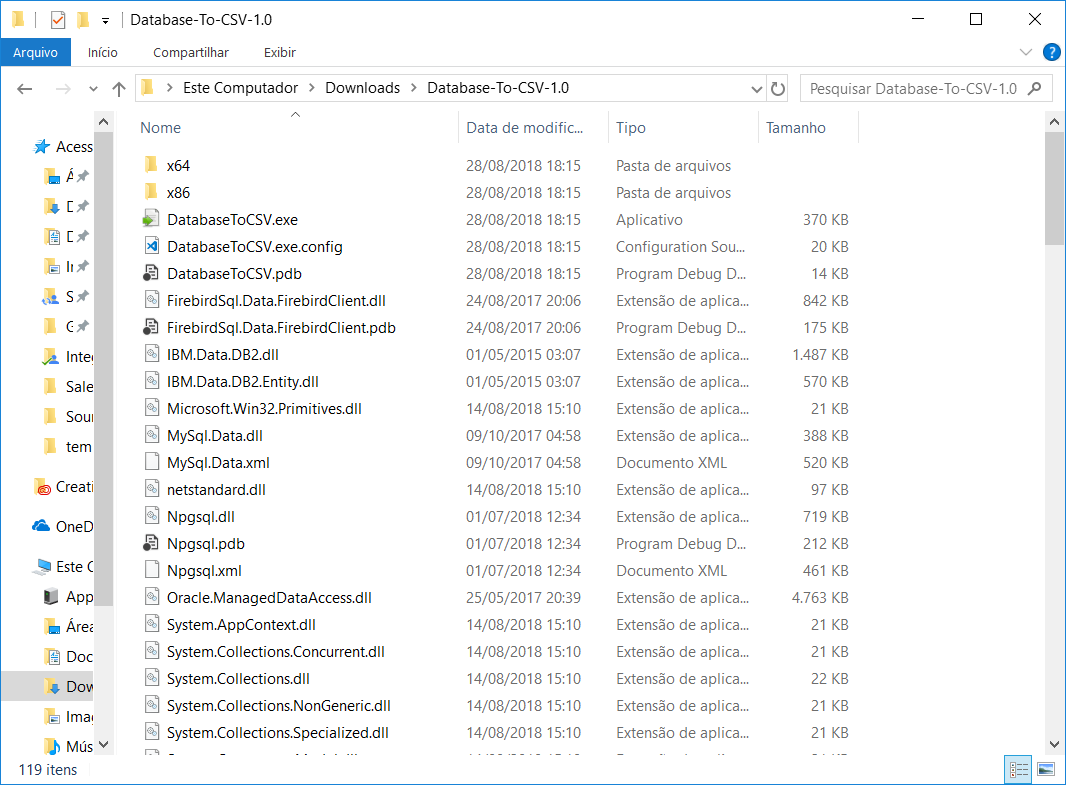
Necessário realizar o download do repositório no GitHub, irá conter os seguintes arquivos:

1. **DatabaseToCsv** – Programa que faz a consulta no banco de dados e exporta o retorno para CSV.

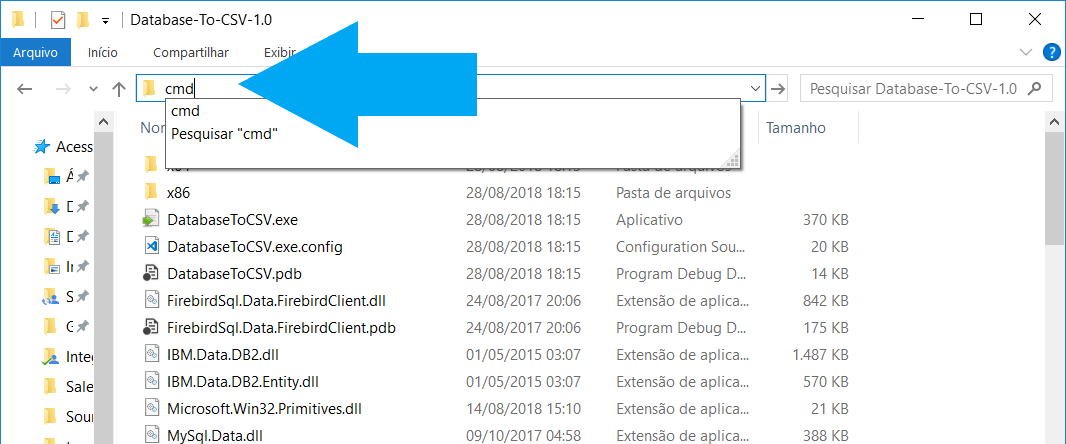
*Link***:** <https://www.salescope.com.br/suporte/implantacao/integracao-com-erp/automatizando/outros-automatizandores/automatizando-a-query-sql/>

1. **Query** – Pasta com uma Procedure para ser criada para realizar os testes. (Caso já tenha uma query especifica, pode remover essa pasta e utilizar os seus Scripts.)
2. **Send Mail** – Gerenciador para disparo de e-mail automático com PowerShell para encaminhar o CSV que foi gerado pelo software da Salescope.
3. **Docs** – Contém a documentação para realizar a automatização.

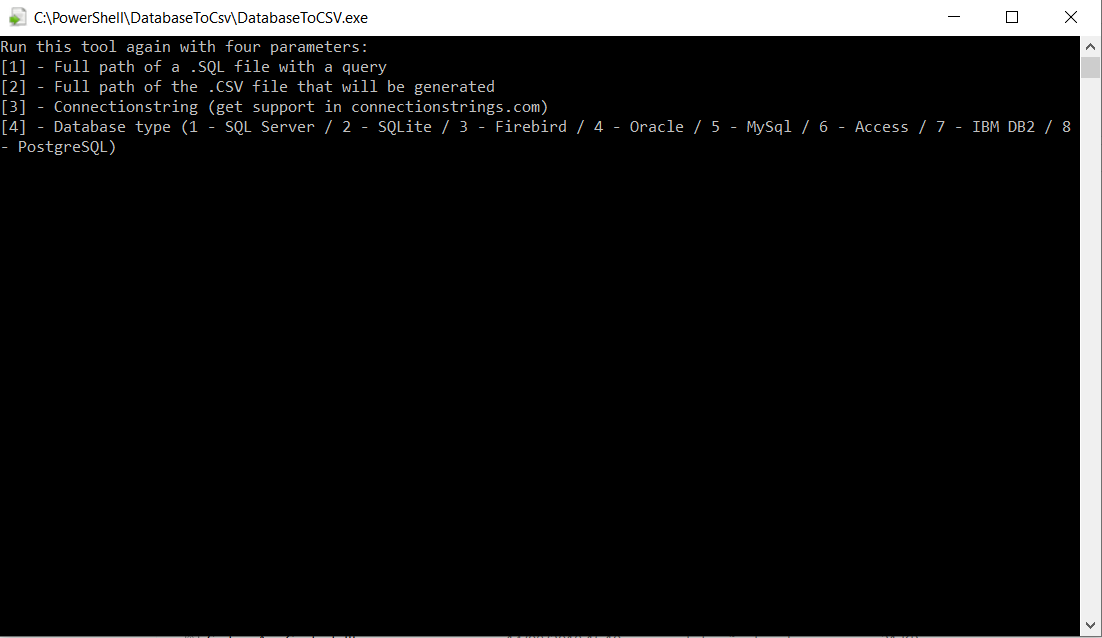
Após o download, descompacte-o em uma pasta.



Agora abra o **prompt de comando do Windows** e navegue até a pasta descompactada. Um atalho para isso é digitar **“CMD”** diretamente na barra de endereços do **Explorador de Arquivos do Windows** e teclar **“Enter”**.



O prompt de comando será aberto na pasta do aplicativo, digite **“DatabaseToCSV”** (sem aspas) e tecle **Enter** para iniciar o mesmo.



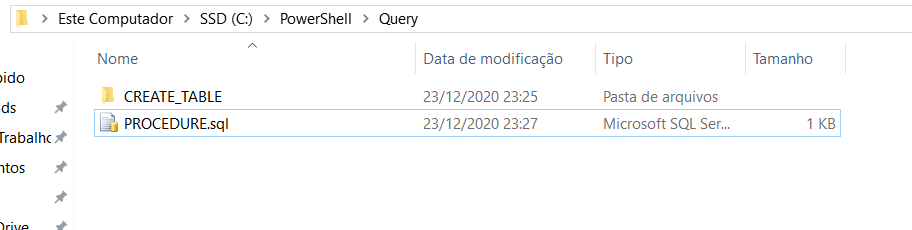
O aplicativo irá exibir uma mensagem de que deve ser iniciado com 4 parâmetros:

1. O **caminho do arquivo .SQL** (contendo a query de consulta ao banco que criamos nas etapas anteriores);
2. O **caminho do CSV** que será gerado;
3. A **string de conexão** com o banco de dados;
4. O **tipo** de banco de dados.

Os quatro parâmetros são obrigatórios e variam de empresa para empresa. Vamos configurar nossa chamada ao aplicativo detalhando cada parâmetro de acordo com nosso exemplo.

**Parâmetro 1: Arquivo SQL**

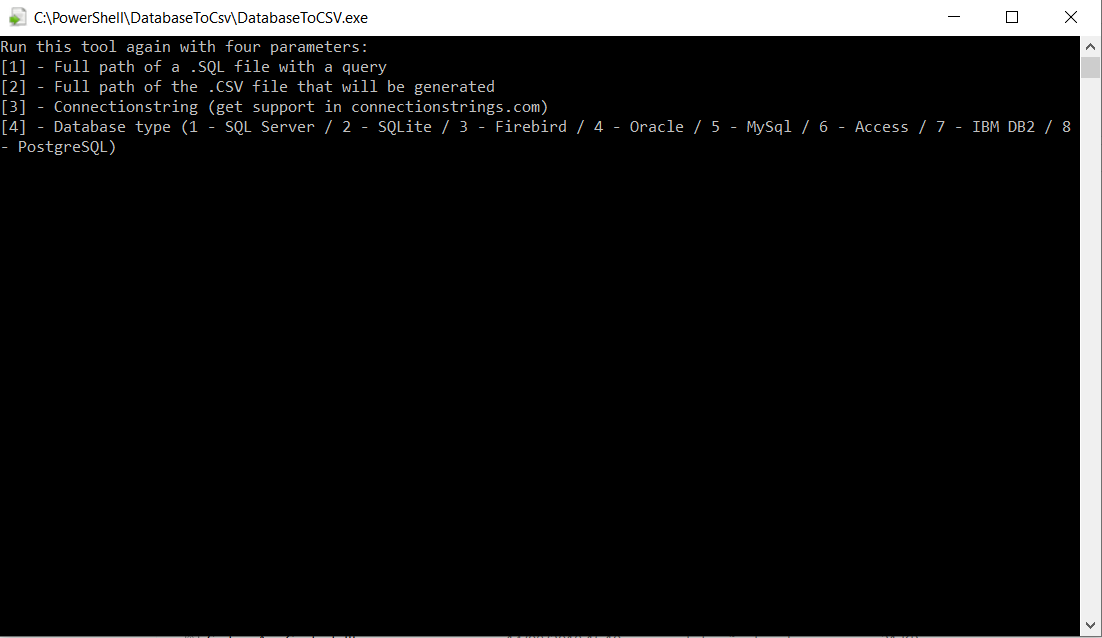
**O primeiro parâmetro é o caminho do arquivo SQL**. Defina um local em que sua query ficará armazenada; em nosso exemplo, vamos criar um arquivo em **“C:\PowerShell\Query\PROCEDURE.sql”**.

Arquivo de query  


Para criar a procedure os scripts estão na pasta “Create\_Table”.

**Parâmetro 2: Arquivo CSV**

**O segundo parâmetro é o arquivo CSV a ser gerado;** em nosso exemplo, vamos gerar um arquivo em **“C:\PowerShell\SendMail\REPORT.csv”**. Por enquanto, nossa chamada está assim:



1. Local onde está seu script; em nosso exemplo está em “C:\PowerShell\Query\PROCEDURE.sql”
2. Onde será exportado o CSV; em nosso exemplo está em “C:\PowerShell\SendMail\REPORT.csv”
3. Connection string para conectar com seu database.
4. Tipo do seu banco de dados; em nosso exemplo será 1, pois se trata de SQL Server.

Então ficará dessa forma:

**"C:\PowerShell\Query\PROCEDURE.sql" "C:\PowerShell\SendMail\REPORT.csv" "Server=SEU SERVIDOR DE SQL;Database=POWERSHELL;Trusted\_Connection=True;" 1**

**Parâmetro 3: Disparo de E-mail**

**O terceiro parâmetro é o script em PowerShell para enviar e-mail;** em nosso exemplo, vamos gerar um arquivo em **“C:\PowerShell\SendMail\Sendmail.ps1”**. Por enquanto, nossa chamada está assim:

### 

Ao abrir o arquivo terá a seguinte estrutura:

### # DEFINIÇÃO DE VARIÁVEL

### $responsavel = "SEU NOME"

### $data = Get-Date -format "dd/MM/yyyy HH:mm:ss"

### $de = "SEU EMAIL"

### $smtp = "smtp.outlook.com"

### $porta = "587"

### $para = "EMAIL DO DESTINATARIO"

### $assunto = "DISPARO AUTOMÁTICO DIÁRIO - $data"

### $Body = "RELATÓRIO DIÁRIO - AUTOMÁTICO:`n

### `DATA: $data `n`nDESCRIÇÃO: Este é um e-mail automático, por favor não responda! `n`n Em caso de dúvidas ou necessidade de suporte contate o service desk para atendimento."

### $anexo1 = "C:\PowerShell\SendMail\REPORT.csv" #Você pode especificar manualmente o caminho onde o arquivo que será enviado por e-mail, encontra-se, vale dizer, onde ele está armazenado no seu computador.

### # ENVIAR E-MAIL

### $Message = New-Object System.Net.Mail.MailMessage

### $Message.Subject = $assunto

### $Message.Body = $Body

### $Message.IsBodyHtml = $false

### $Message.To.Add( $para)

### $Message.From = $de.Replace("=", "@") # For providers that use = instead @ to identify username

### $attachment1 = new-object Net.Mail.Attachment($anexo1)

### $message.attachments.add($attachment1)

### $SMTP = New-Object System.Net.Mail.SmtpClient($SMTPServer, $SMTPPort);

### $SMTPServer = “smtp.outlook.com”

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

### $SMTPClient.EnableSsl = $true

### $SMTPClient.Credentials = New-Object System.Net.NetworkCredential($de, “SENHA DO SEU EMAIL”);

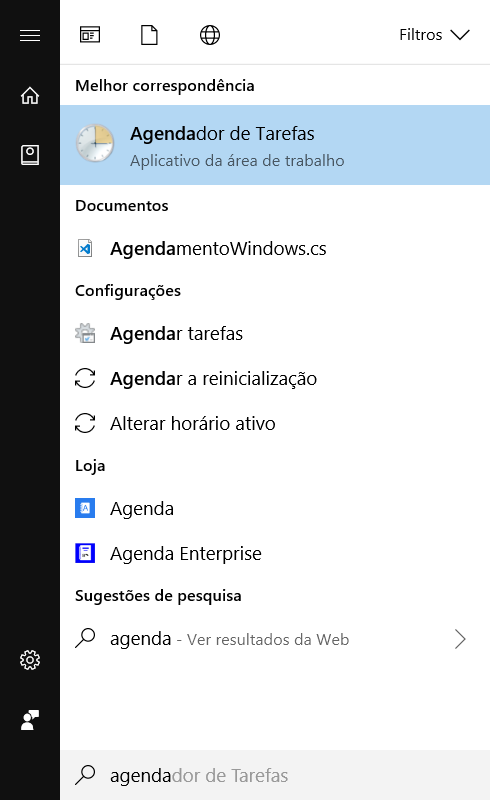
### $SMTPClient.Send($Message)

### # FIM

Após alterar, execute no PowerShell para teste e salve seu arquivo .ps1

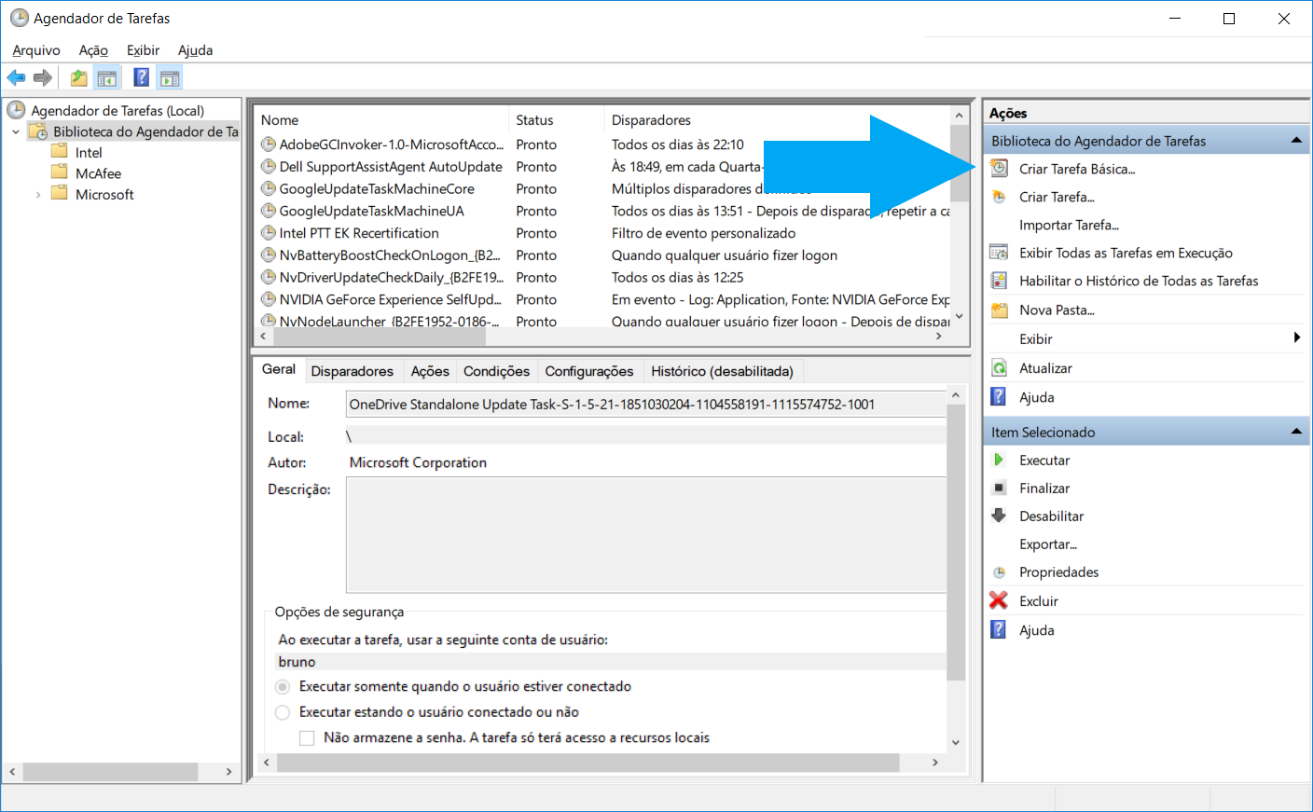
### Automatizando a conversão

Agora que validamos a conversão do arquivo, vamos automatizá-lo. Clique no menu iniciar do Windows e digite **“Agendador de tarefas”** (ou, em inglês, Task Scheduler), e tecle **Enter**.

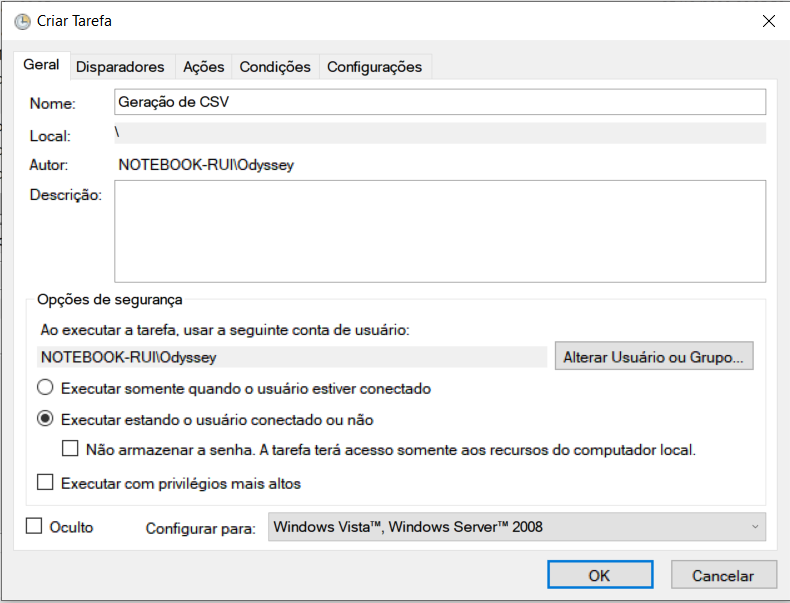


O agendador de tarefas do Windows permite que um programa seja executado de periodicamente. Iremos agendar nosso conversor para ser executado diariamente.

No lado direito da janela, clique em **“Criar tarefa”**.



Defina um nome para identificar a tarefa, por exemplo **“Geração de CSV”**, e clique em **“Avançar”**.



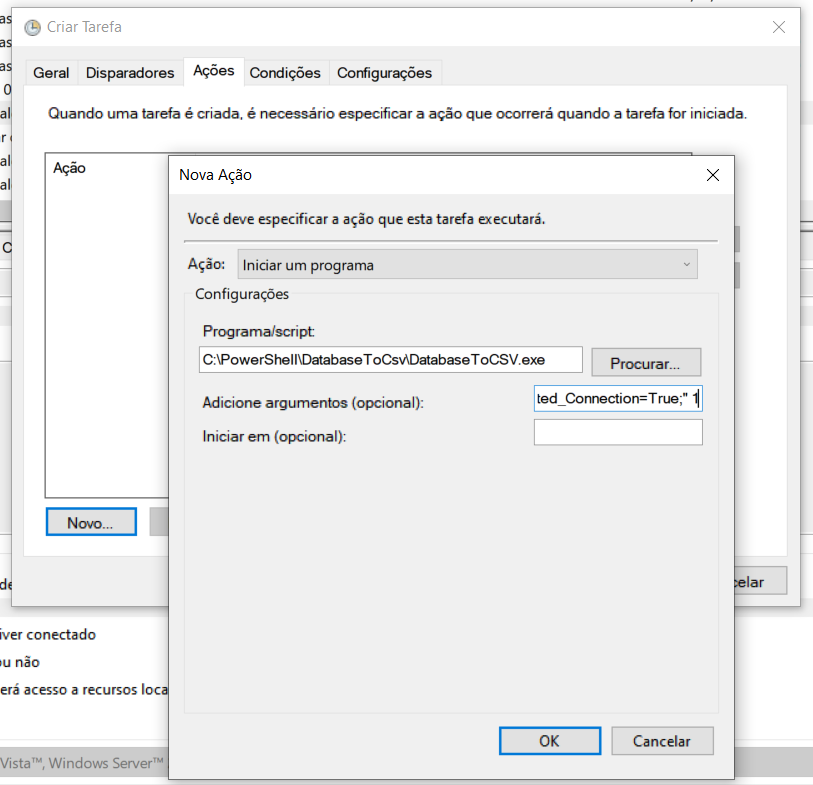
Defina um horário para disparar e logo após vá em “**Ações**” e insira os seguintes parâmetros para criarmos a geração automática do CSV:

**Programa/Script:**

**C:\PowerShell\DatabaseToCsv\DatabaseToCSV.exe**

**Adicione argumentos(opicional):**

**"C:\PowerShell\Query\PROCEDURE.sql" "C:\PowerShell\SendMail\REPORT.csv" "Server=SEU SERVIDOR DE SQL;Database=POWERSHELL;Trusted\_Connection=True;" 1**



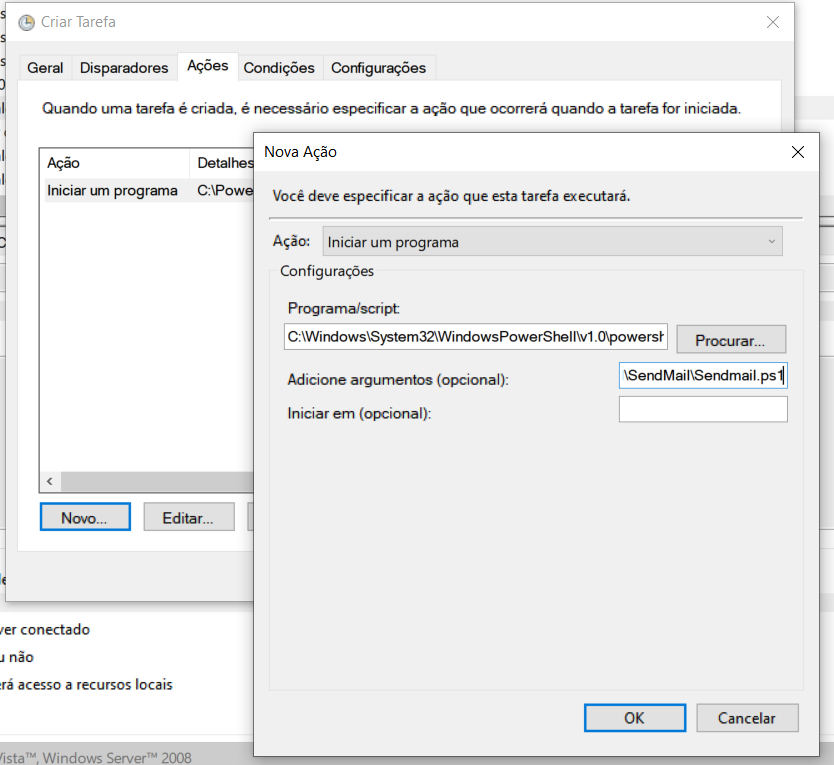
Após definir a geração do CSV vá em “**Novo**” e insira os seguintes parâmetros para criarmos o disparo de e-mail.

**Programa/Script:**

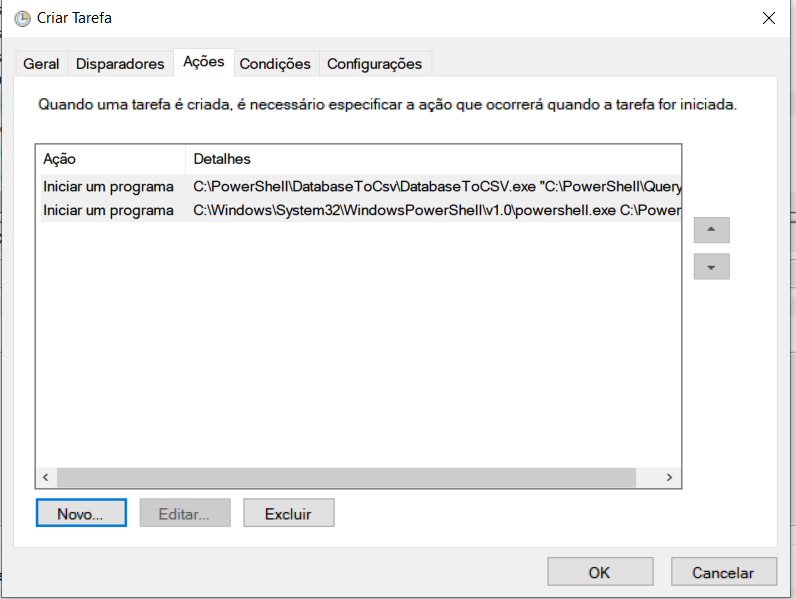
**C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe**

**Adicione argumentos(opicional):**

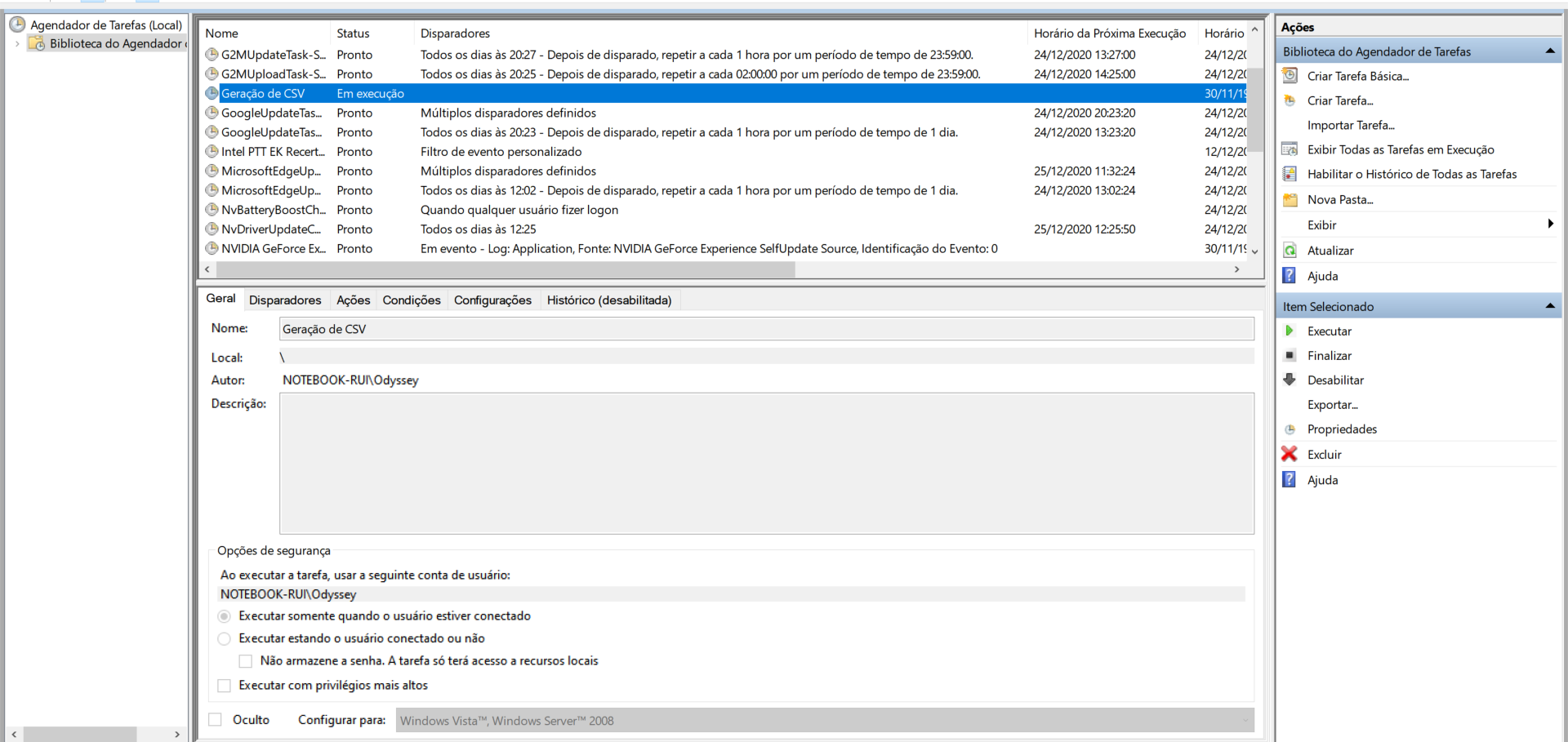
**C:\PowerShell\SendMail\Sendmail.ps1**



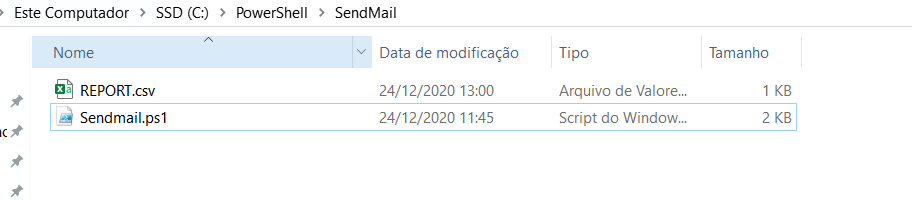
Salve as informações e clique em **“OK”**.



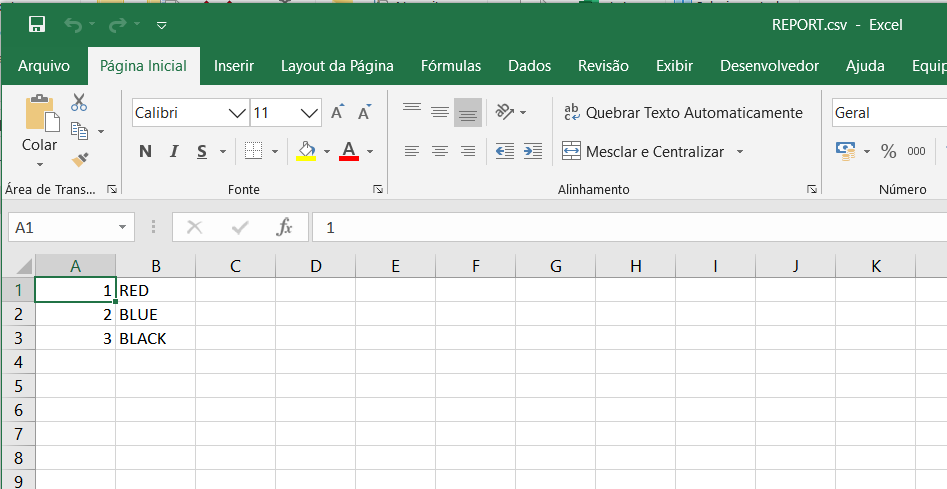
Após isto clique com o botão direito em cima da tarefa e **“Executar”**.



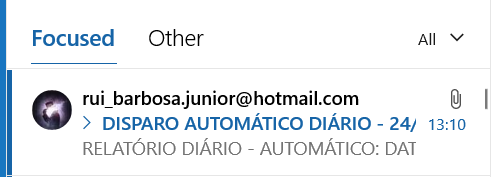
Após executar a tarefa o arquivo **“REPORT”**, vai ser gerado na pasta de envio de e-mail.



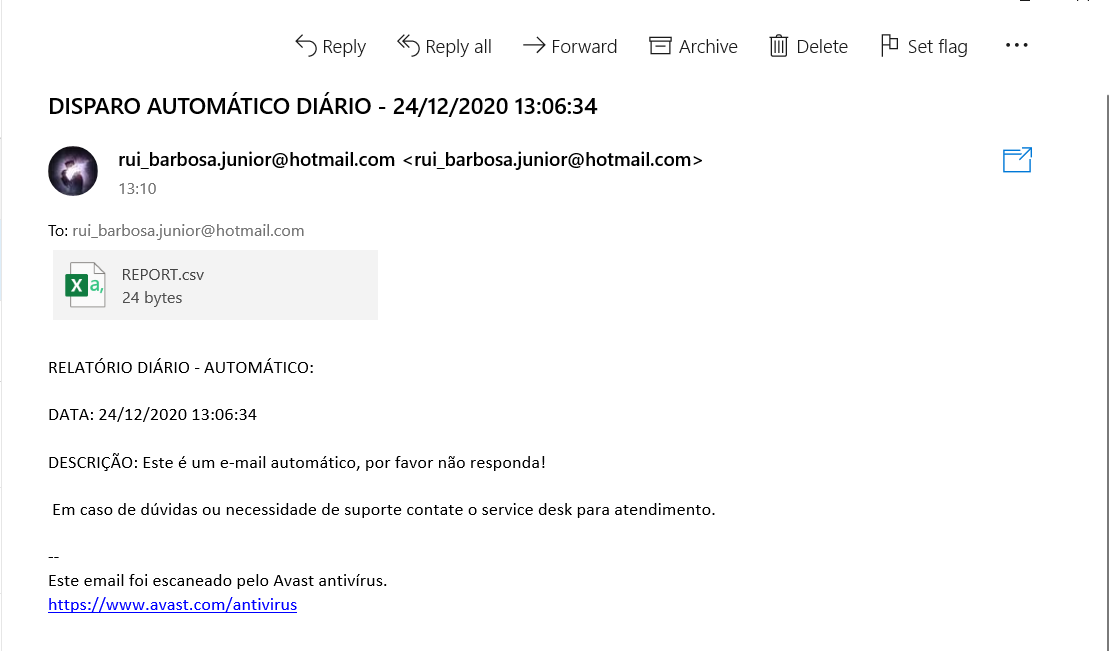
Ao abrir o arquivo **“REPORT”**, vemos os dados do retorno da Procedure.



Logo após o arquivo **“REPORT”** ser gerado, o e-mail será disparado:



Ao abrir o **“Email”** vemos o arquivo anexo e o body com as informações:



Caso houver dúvidas da implementação do software da salescope e automatizar o disparo de e-mail, entre em contato comigo através das redes sociais ou e-mail.

E-mail: rui\_barbosa.junior@hotmail.com