****

**Proyecto Final Análisis Multidimensional de Datos**

**Profesor: Héctor Fernández**

**Integrantes:**

**Aaron Vargas Malavasi**

**José Zúñiga Martínez**

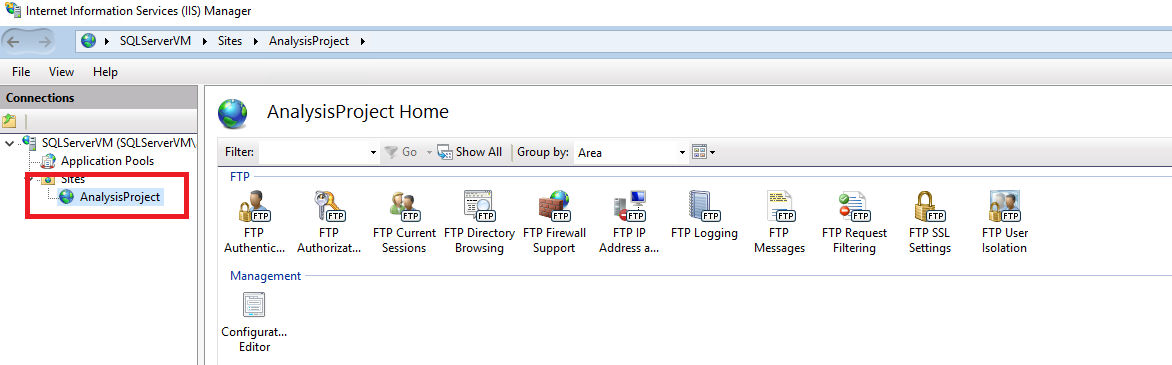
**Alonso Mesén Quesada**

**Andrés García Soto**

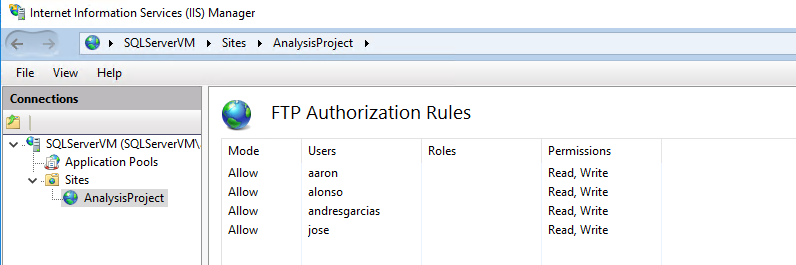
**URL del repositorio de Github:** <https://github.com/AlonsoMesen14/AnalisisMultiDimensional-Proyecto>

**Primera Parte. Creación del FTP**

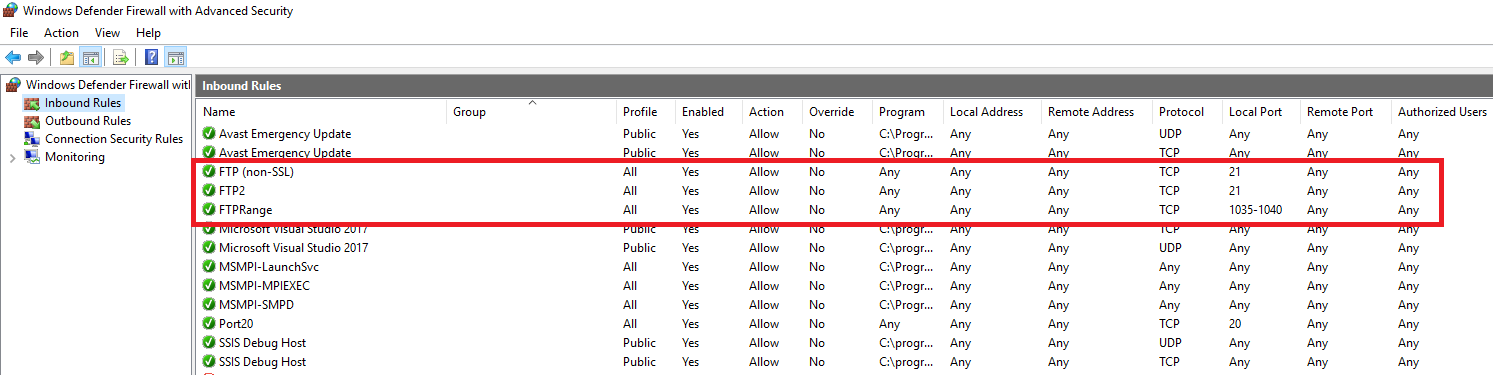
1. IIS con el FTP creado bajo el nombre de “AnalysisProject”. La dirección IP del servidor es ftp://40.114.74.47 y se accesa con los credenciales de los miembros del grupo que se encuentran en la tabla parámetros encriptada.



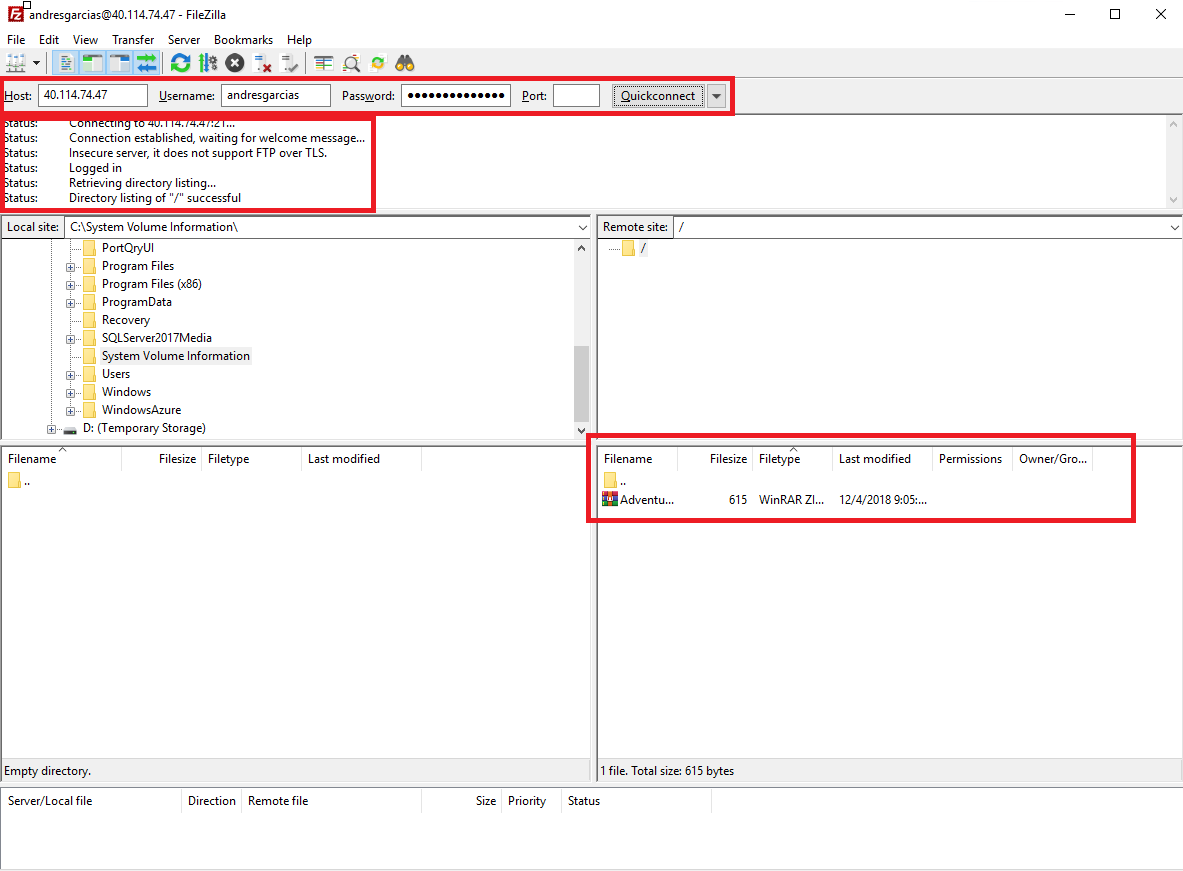
1. FTP con los permisos respectivos

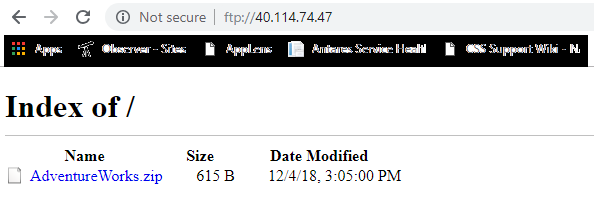


1. Firewall del servidor con las reglas habilitadas



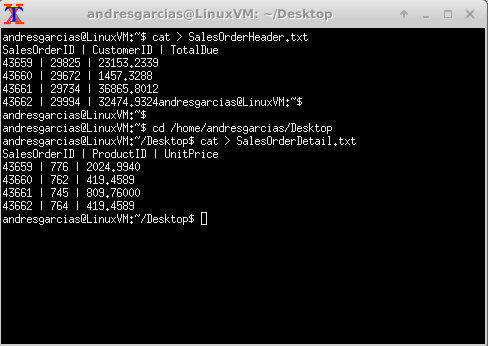
4. Conexión al FTP

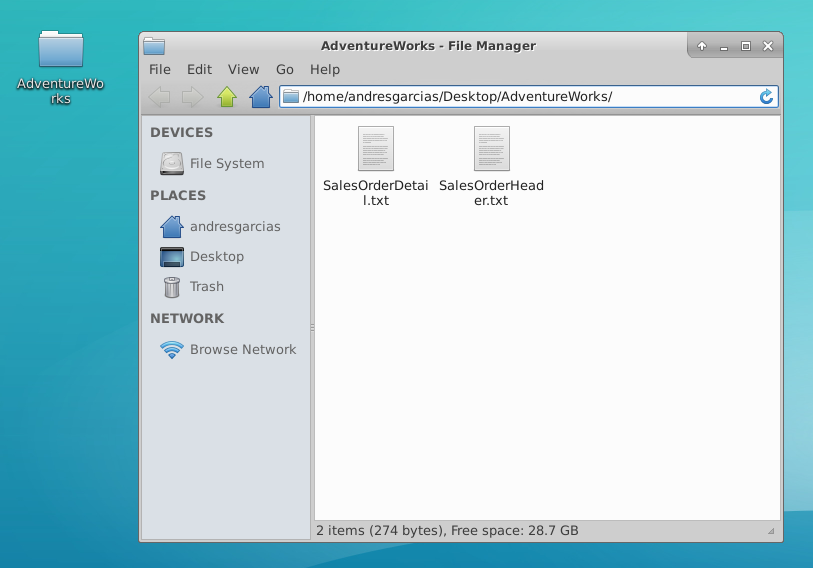


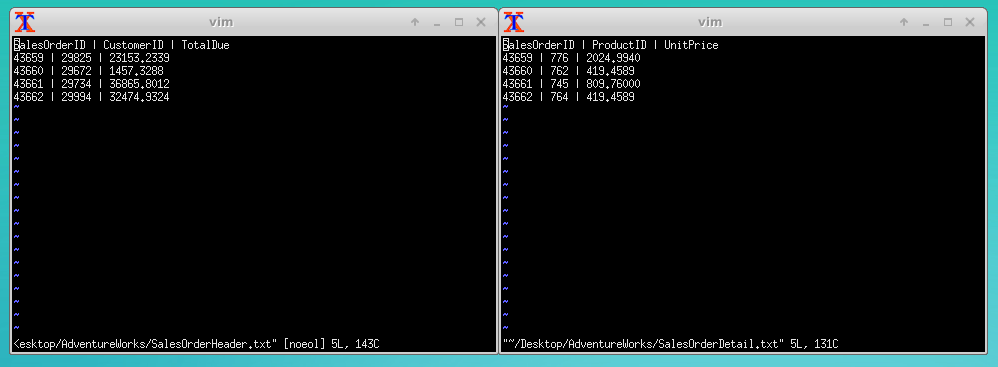


**Segunda Parte. Creación del ZIP en Linux con los textos planos conteniendo datos de la base de datos AdventureWorks.**

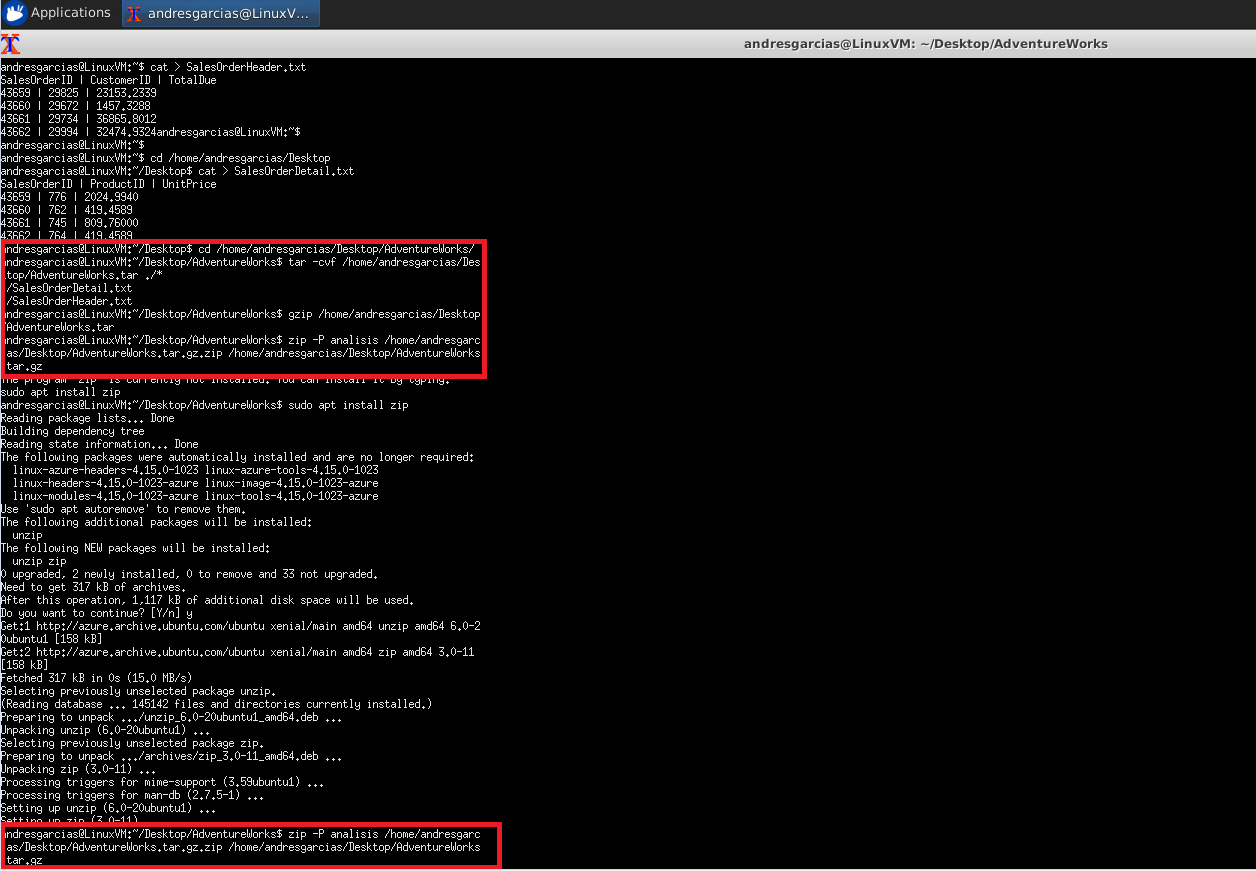
1. Creación de los textos planos en Linux con los delimitadores respectivos.

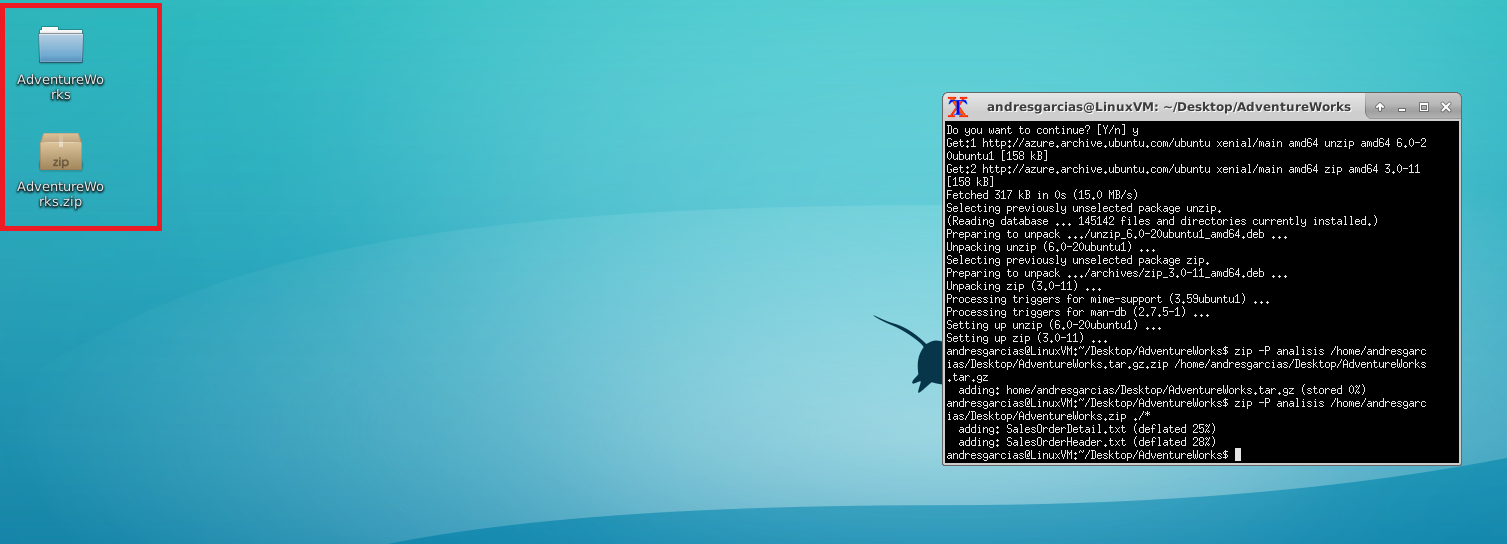




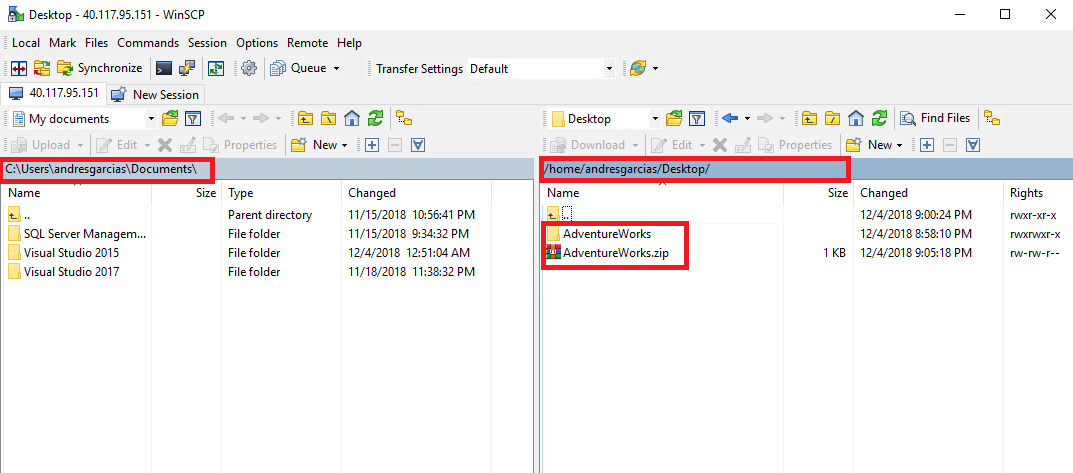


2. Compresión en .tar.gz y seteo de contraseña al archivo AdventureWorks. La contraseña será “análisis”.



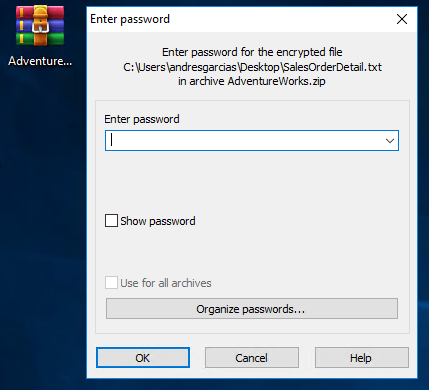


3. Ya comprimido el archivo, se necesita pasar a Windows. Para esto se usó un Software llamado WinSCP.

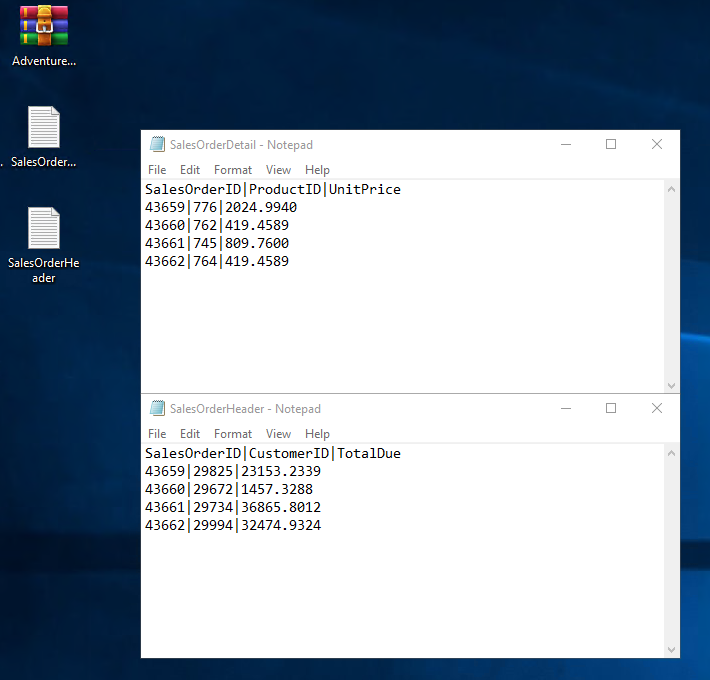


4. Posterior a eso, se arrastra al escritorio de Windows y cuando se intenta extraer, pide la

contraseña la cual era “análisis”.

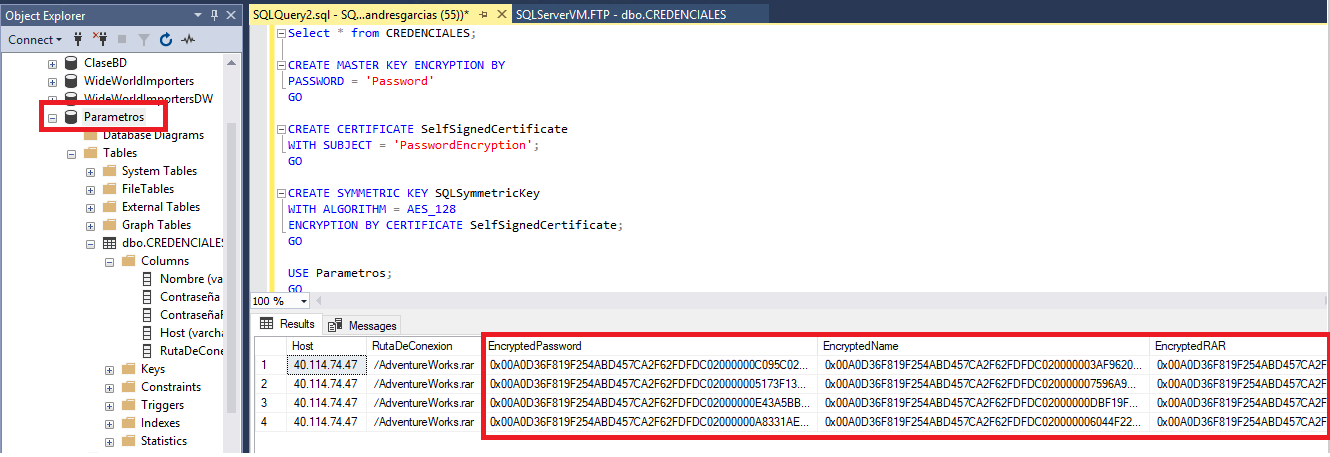


5. Los archivos se descomprimen y se encuentran delimitados con |.



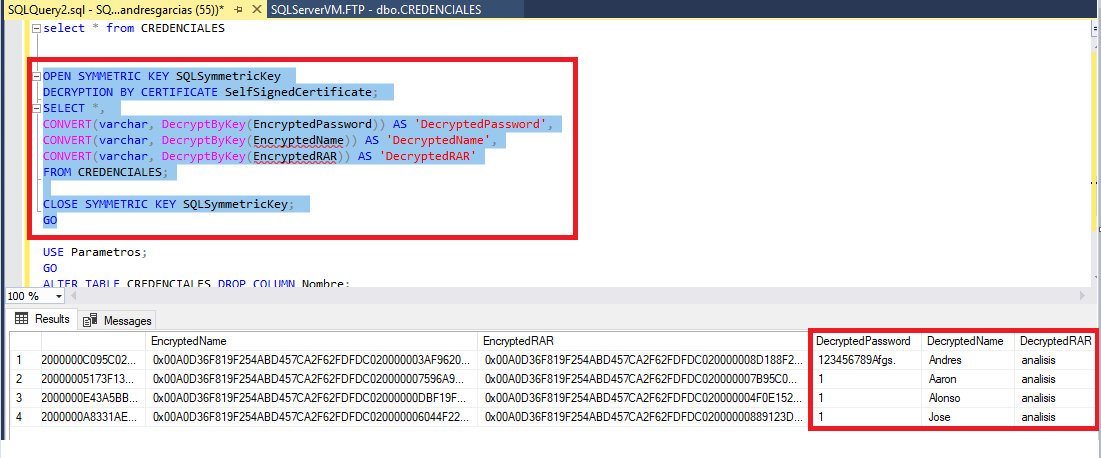
**Tercera Parte. Encriptar la base de datos “Parámetros”.**

1. Los datos encriptados son los nombres de los usuarios, las contraseñas de los usuarios y la contraseña del .zip con la información de la tabla AdventureWorks.

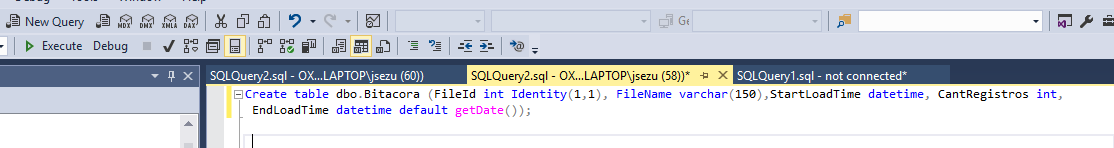


2. Los comandos SQL con los que la tabla fue encriptada se encuentran en Github bajo el nombre de: **CodigoTablaParametros.txt**

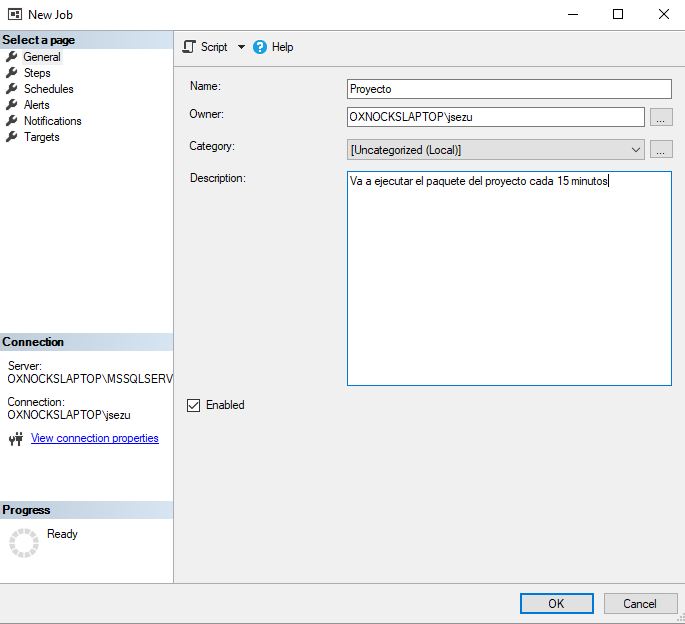
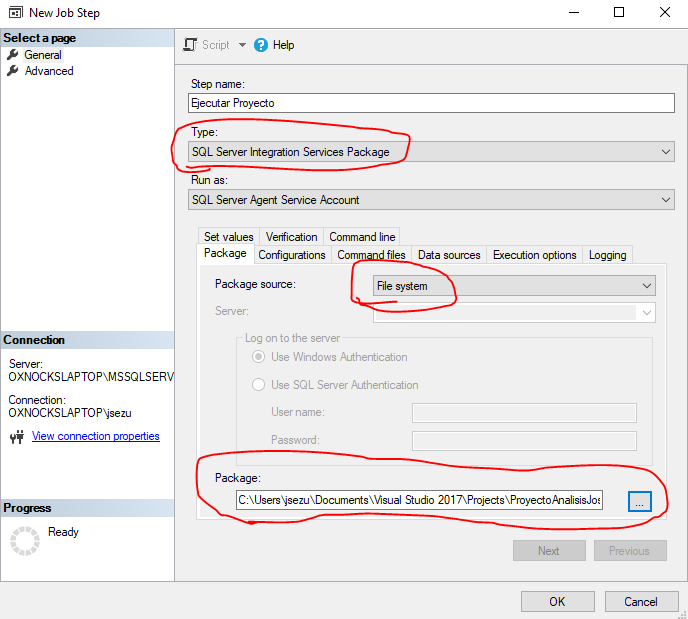
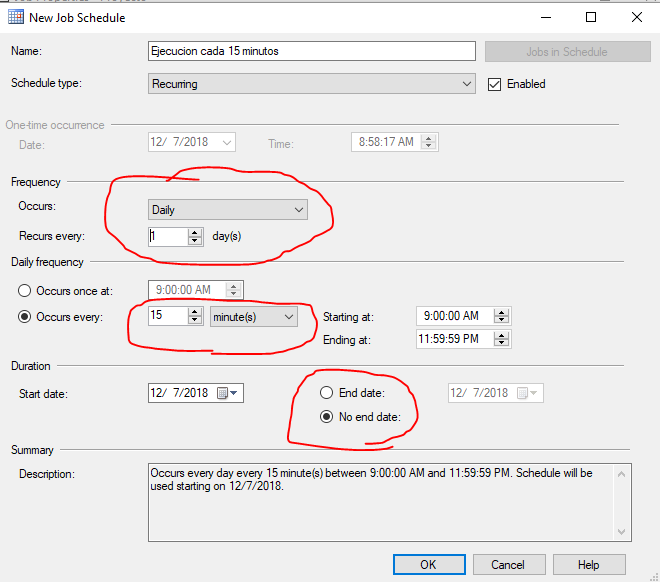
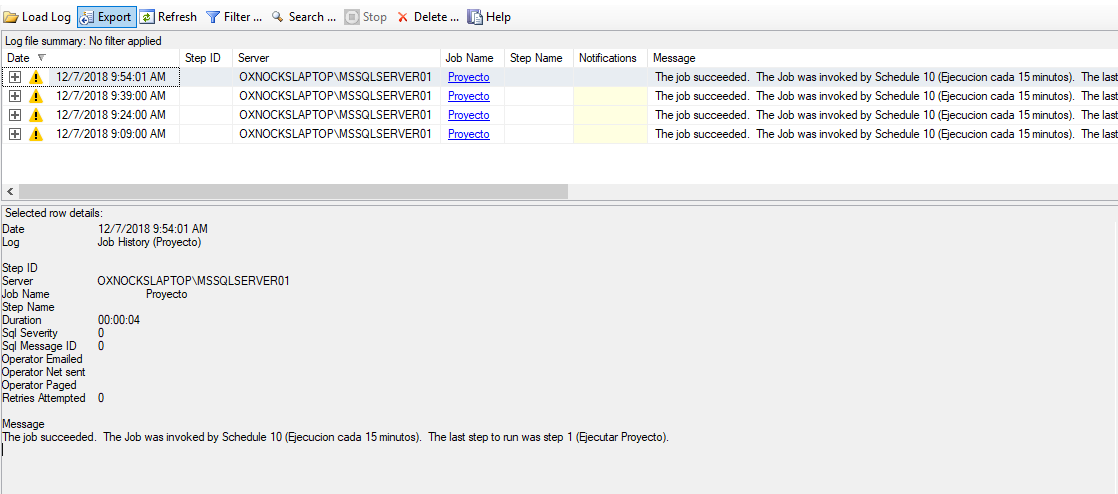
3. Los comandos para desencriptar la tabla se encuentran también en el repositorio de Github, en el mismo archivo **CodigoTablaParametros.txt.**



**Cuarta Parte. Tablas de parámetros y bitácora**

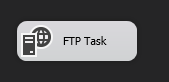
1. La tabla parámetros es la que se especificó en el punto anterior (encriptada).
2. Tabla Bitácora  
    a. Creación  
     
     
     
    b.Llenado  
   El llenado de la tabla se va a ver con los procesos que están a continuación dentro de este documento.

**Quinta Parte. Agente SQL**

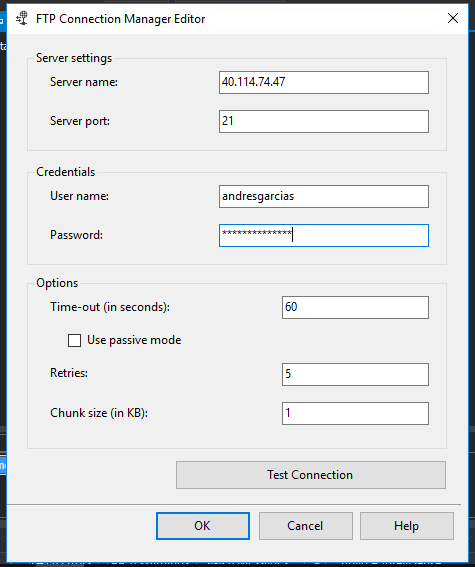
1. Dentro del MSSQL Server Management vamos a abrir el SQL Server Agent, y le damos click derecho, nuevo trabajo.  
   En la ventana general lo vamos ha hacer es especificar el nombre del “Job” y una descripción.  
     
   
2. Debemos ir a la pestaña “Step”, escribimos un nombre para el paso, seleccionamos el tipo como SSIS Package, el source como File System y en Package buscamos el paquete que queremos ejecutar cada 15 minutos.  
     
   
3. Por último debemos ir a la pestaña “Schedule”, especificamos que la frecuencia sea diaria y que ocurra cada día, luego que ocurra cada 15 minutos, seleccionamos la hora de inicio y la hora final y por último que no se va a a tener una fecha de finalización de ejecución del paquete.  
     
   
4. Para ver si el agente está ejecutando el paquete debemos darle click derecho al “Job” que ejecuta nuestro paquete y darle click a “view history”.  
   

**Sexta Parte. Carga y proceso**

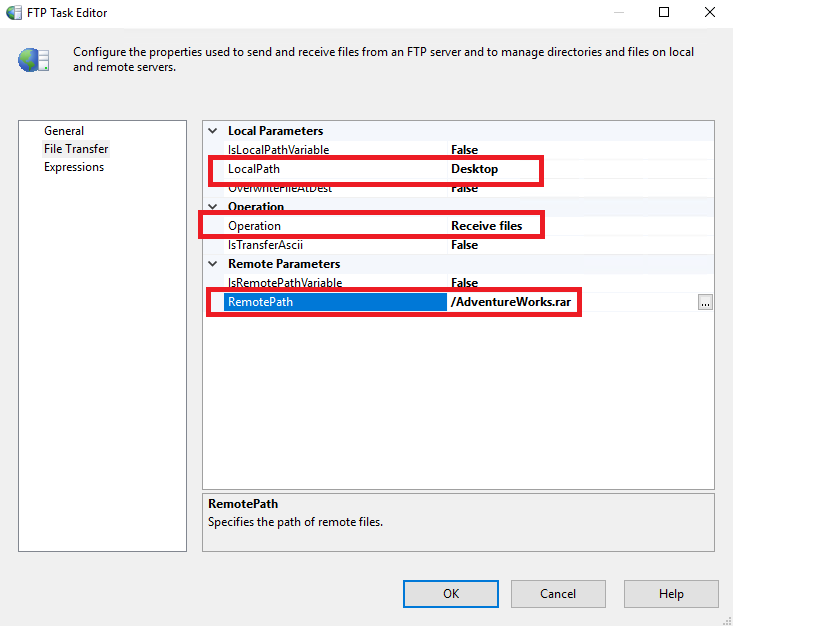
1. La primer función que se ejecuta es la de FTP Task, esta se encargará de traer el FTP público al sistema local.



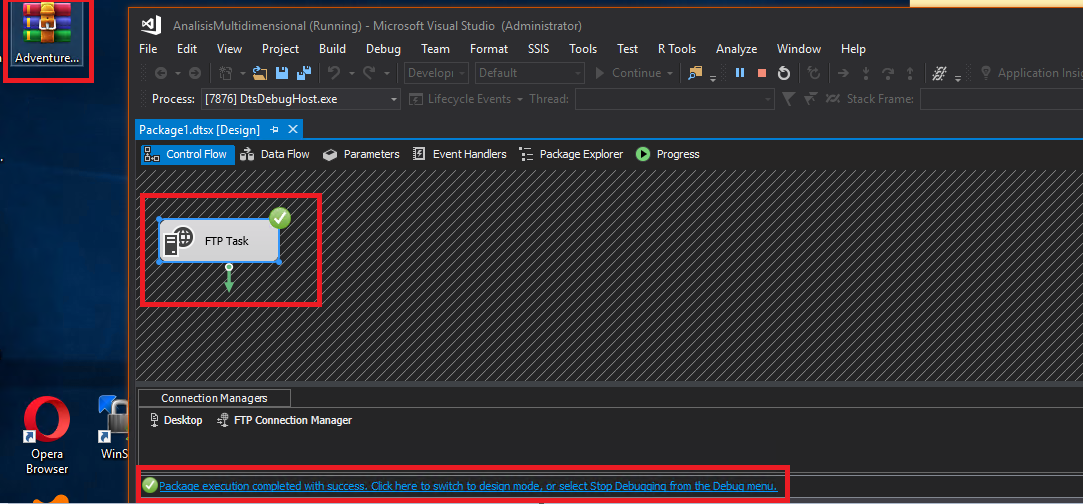
2. El FTP Connection Manager posee el host (dirección IP) y los credenciales del usuario que vaya a acceder al mismo.

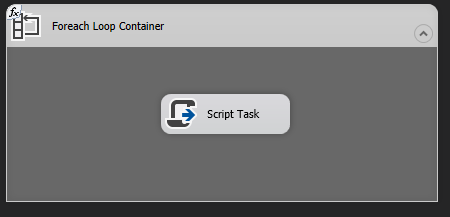
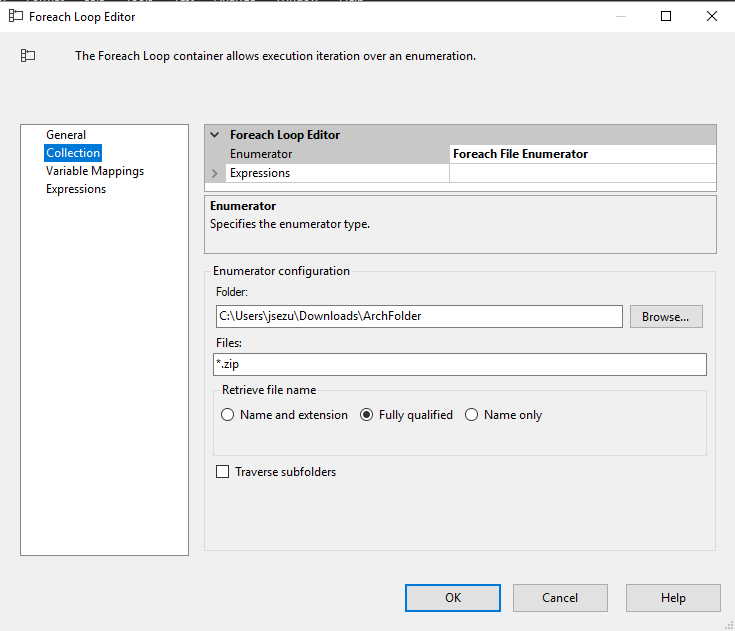
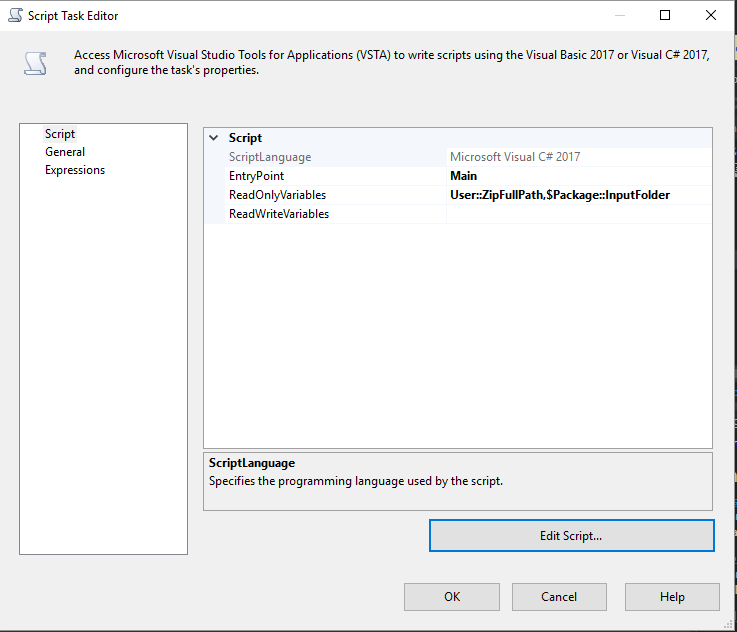


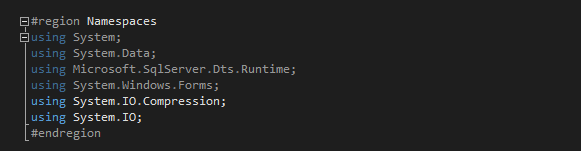
3. Se debe especificar el path destino de los archivos, el path origen de donde se encuentra el FTP o los archivos que quieren ser traídos del mismo y la operación debe ser “Receive Files”

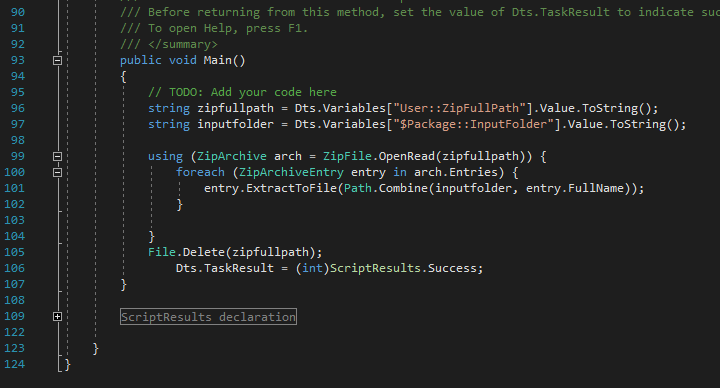


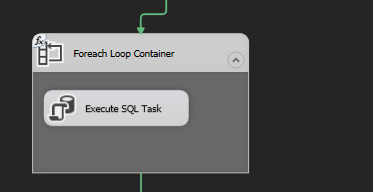
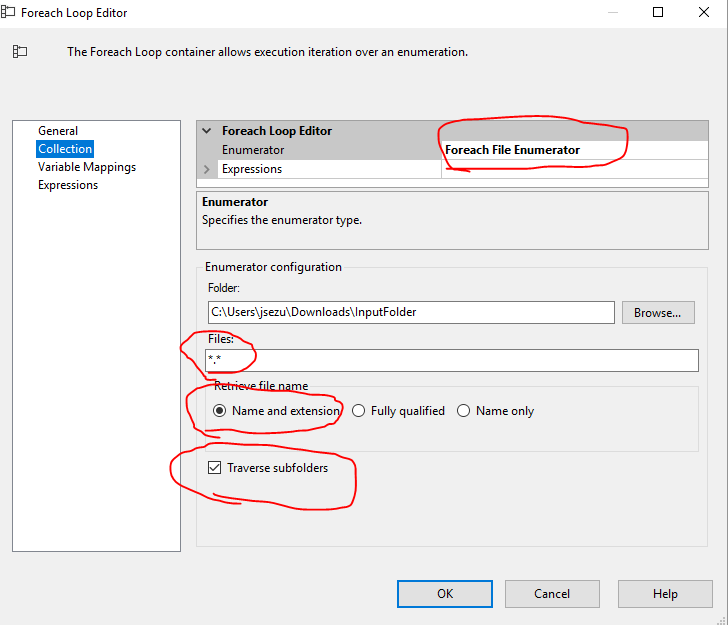
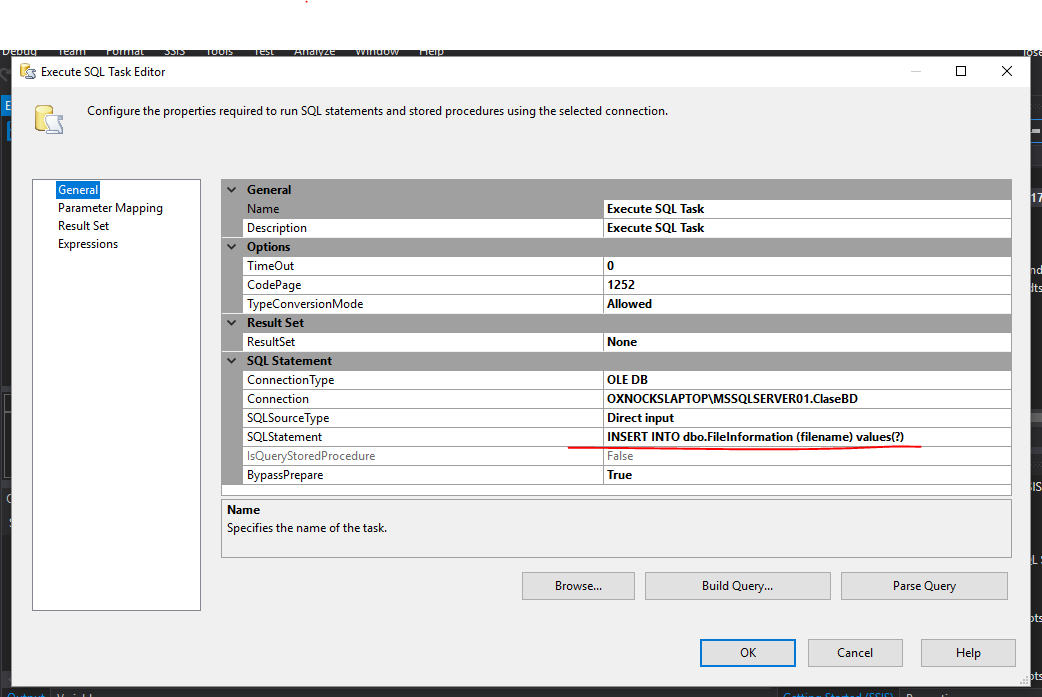
4. Se corre el task y los archivos son traídos al sistema.

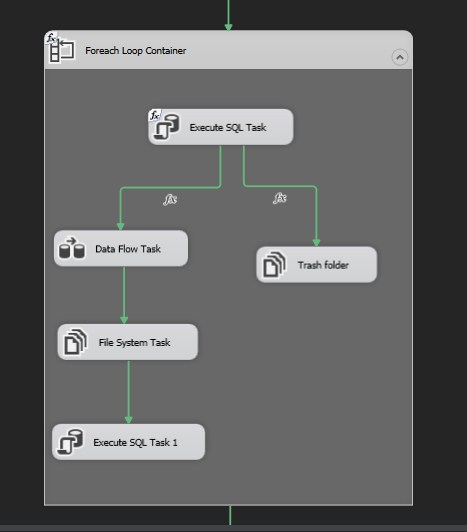


1. Lo primero que debemos hacer es descomprimir el archivo dentro del FTP para lo que implementaremos lo siguiente:  
     
     
   El foreach nos permitirá recorrer todos los archivos que estén dentro de la carpeta y que tenga el formato zip por medio de lo siguiente:  
     
   
2. Mientras que el script task es el encargado de la descompresión de los archivos, por medio de una modificación en el script, la cual es realizada en el lenguaje C#.  
     
     
     
     
     
   a. Lo primero que debemos hacer es agregar las librerías “System.IO.Compression” y “System.IO”

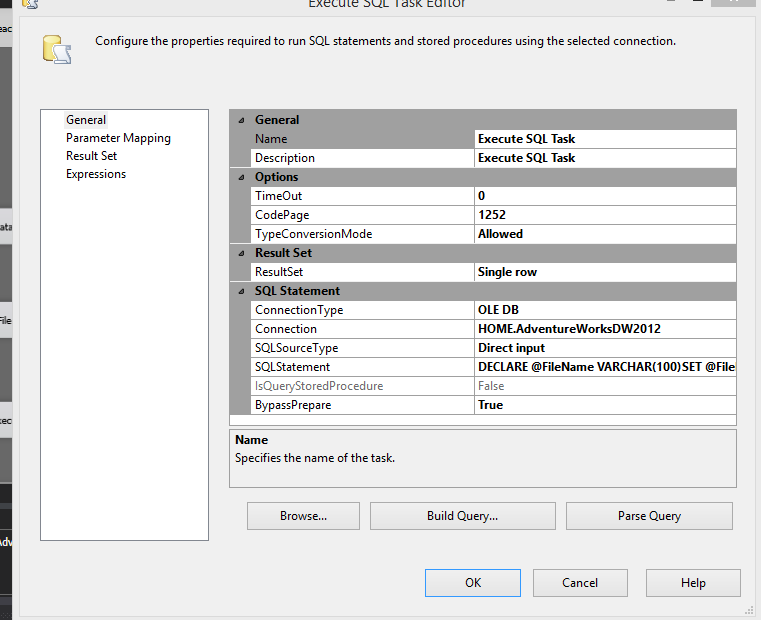


b. Y luego especificamos en el método main que vamos a descomprimir los archivos zip dentro una carpeta donde se descargan los archivos FTP y los vamos a pasar a una carpeta donde nos sea más sencillo su uso.  
  


1. Ya que descomprimimos los archivos, vamos a leer los nombres y guardarlos en una tabla bitácora. Por lo que debemos hacer un foreach para obtener todos los archivos dentro de la carpeta y un SQL task para hacer la inserción de los a nombres a la tabla bitácora.  
     
     
     
   a. En el Foreach vamos a especificar que es un File Enumerator, que se aceptan cualquier tipo de archivo, que solo vamos a traer el nombre y la extensión y por último que también va a entrar a carpetas internas dentro de la carpeta que tiene los archivos con el fin de utilizar todos los archivos dentro de la carpeta y sub-carpetas.  
     
     
   b.En el SQL task vamos a insertar en la tabla el nombre de los archivos  
   
2. Este Foreach se encarga de validar que la información sea cargada en la base de datos en caso de ser la primera iteración. Si ya la información fue cargada y no se realizó ningún cambio, la información se enviará a un folder que almacena la información repetida (Trash folder). En el caso de que se intente cargar el mismo documento pero a este se le haya realizado algún cambio, se agregaran los nuevos registros y los que ya estaban se enviaran al folder de repetidos.



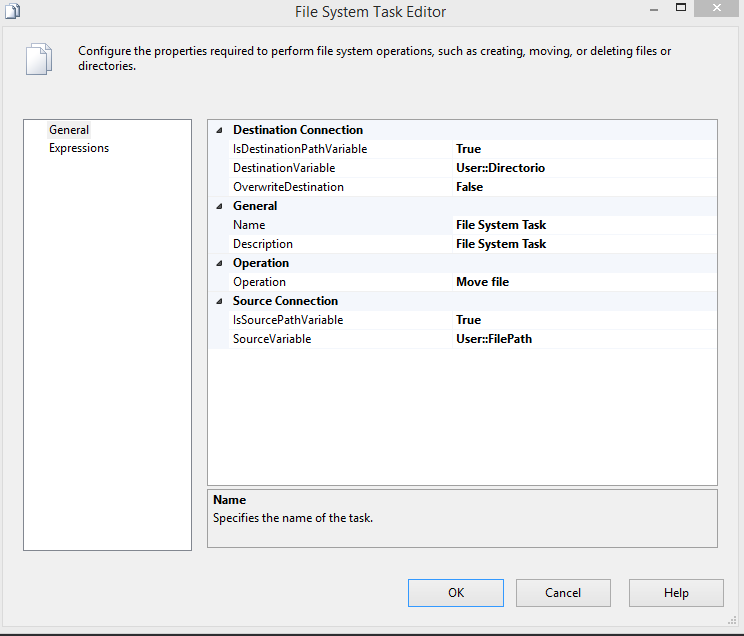
1. En este Execute task se realiza la sentencia sql para que la información que se vaya a cargar retorne un valor de 0 si es la primera vez que se carga o tener información nueva que agregar a la base de datos, 1 si ya ha sido anteriormente cargada toda la información, esto con el fin de enviar la información a su destino respectivo según sea el caso.



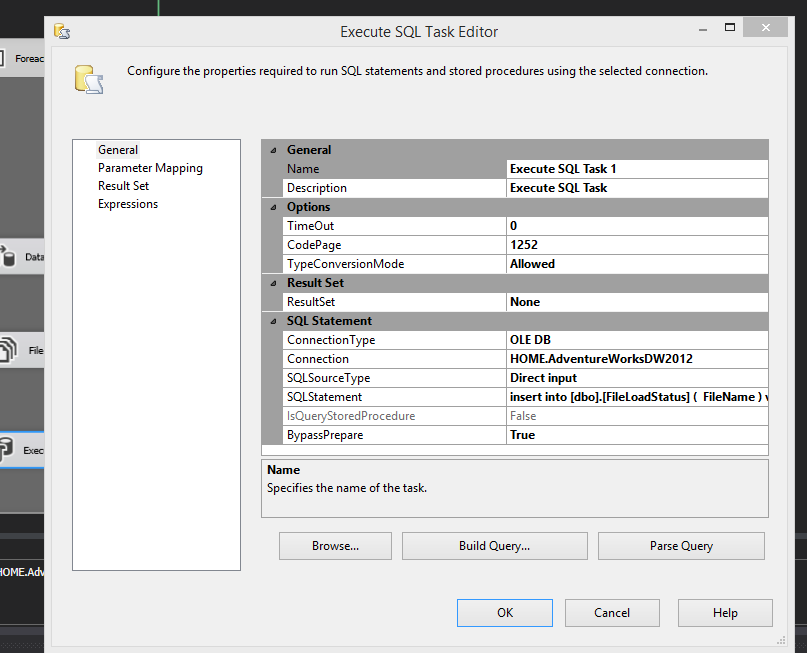
B. En el caso de ser la primera vez que se carga la información o tener algún registro nuevo que agregar, pasará por el Data Flow el cual traerá el archivo correspondiente ubicado en escritorio con la información para agregar y la agregara a la tabla de destino ubicada en la base de datos.



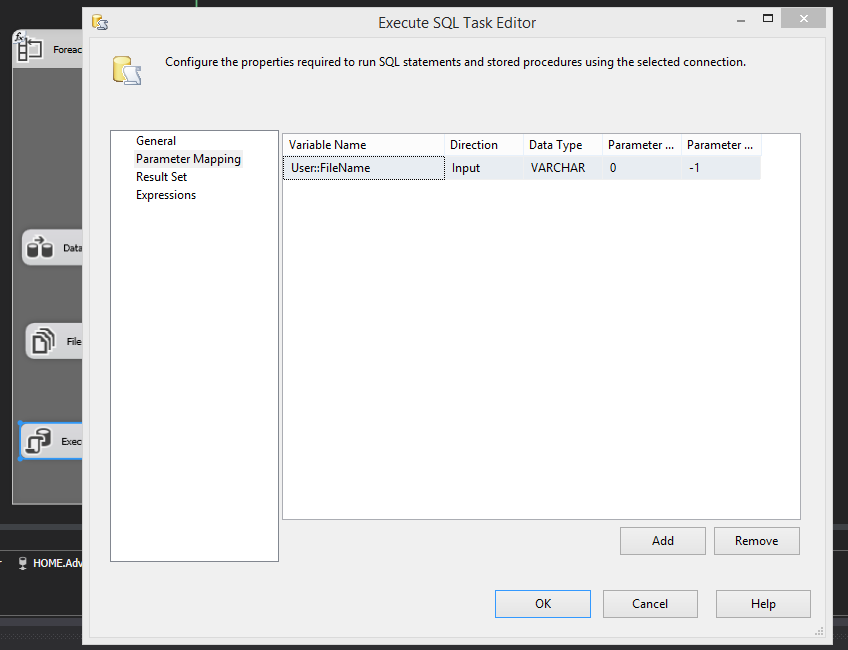
C. En este File Task le indicamos que siguiendo el flujo de cargar la información en la base de datos, se moverá el archivo por medio de una variable de destino llamada Directorio y se utiliza la variable FilePath para indicar que cargara todos los archivos y el origen de donde provienen.



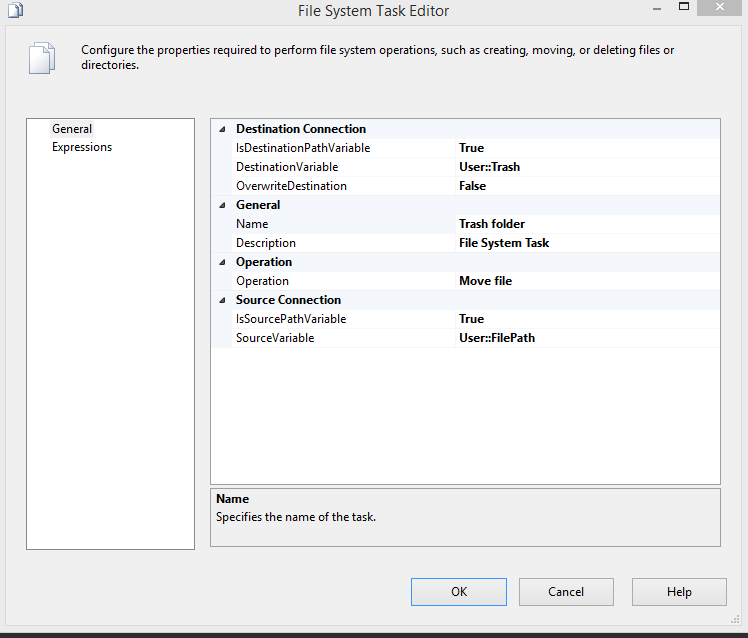
D. Por último, se realiza un Execute task para insertar en la tabla los archivos que se desean cargar.



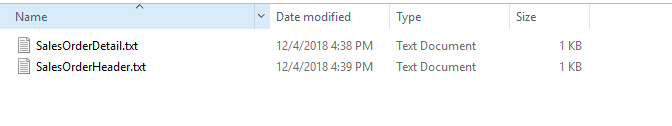
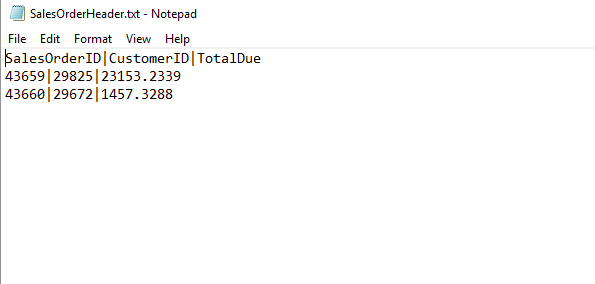
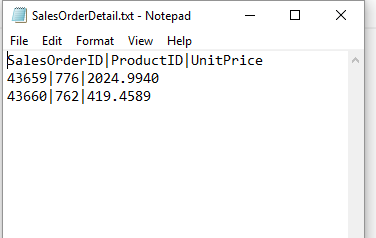
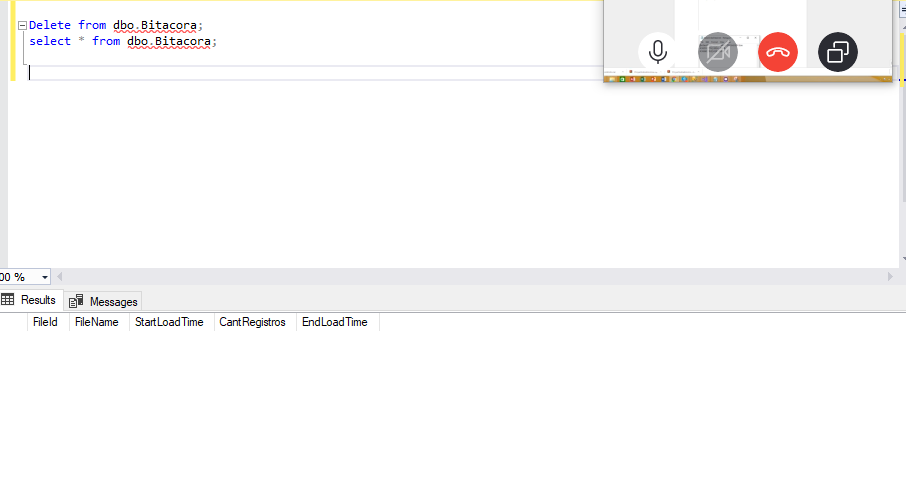
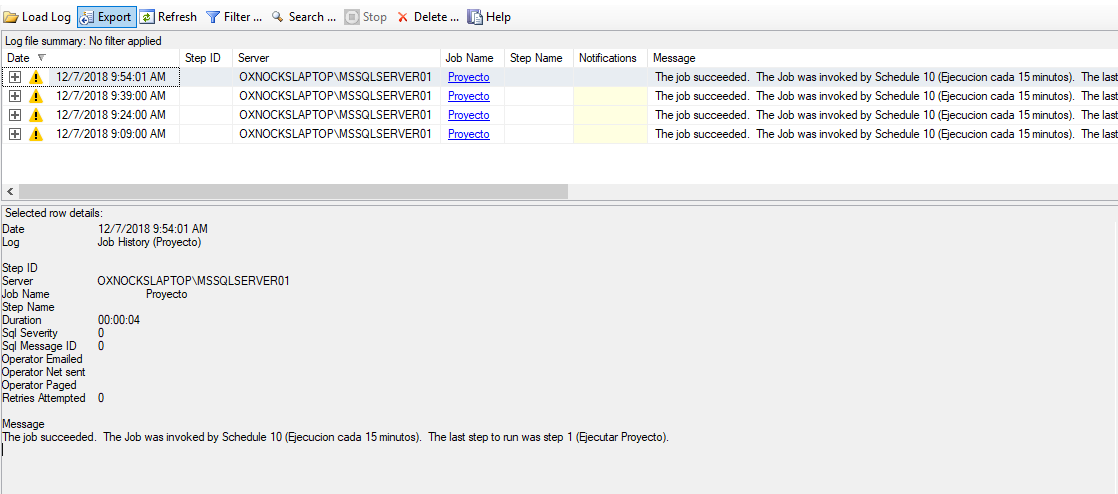
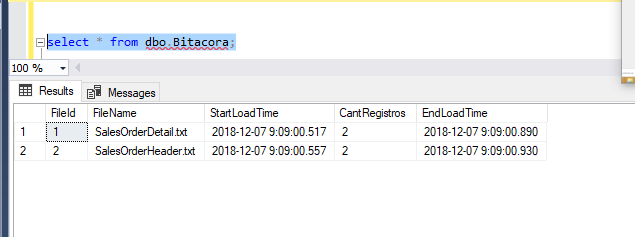
E. En el mapeo de variables especificamos el tipo de input que haremos.



F. En caso de que ya los registros o información del archivo han sido cargados, el flujo se tomará el archivo será distinto, enviando a una carpeta nueva llamada Trash la cual contiene toda la información que esté repetida.

****

**Séptima Parte. Análisis de resultados**

1. Vamos a ejecutar el paquete proyecto con los archivos  
   
2. Vemos la cantidad de registros de ambos archivos  
     
     
   
3. Revisamos la tabla bitácora para ver que este vacía  
   
4. Y se ejecuta el paquete por medio del agente de sql  
   
5. Por último revisamos la tabla Bitácora para ver si la ejecución fue exitosa.   
     
     
     
   Como se puede observar la ejecución fue exitosa y no ingresó los archivos de nuevo a pesar de que el agente de sql ejecutó el paquete varias veces más.