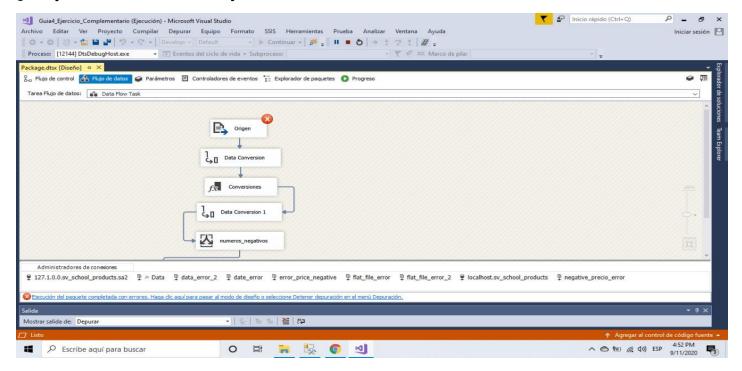
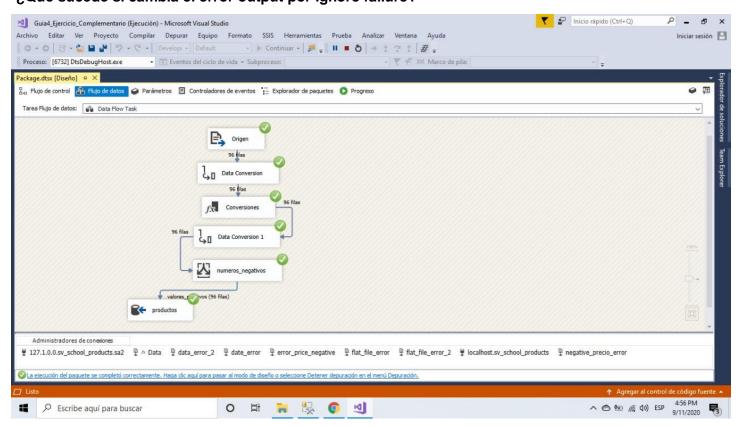
Conclusiones del ejercicio complementario Guía 4:

## ¿Se ejecuta correctamente el ejercicio?



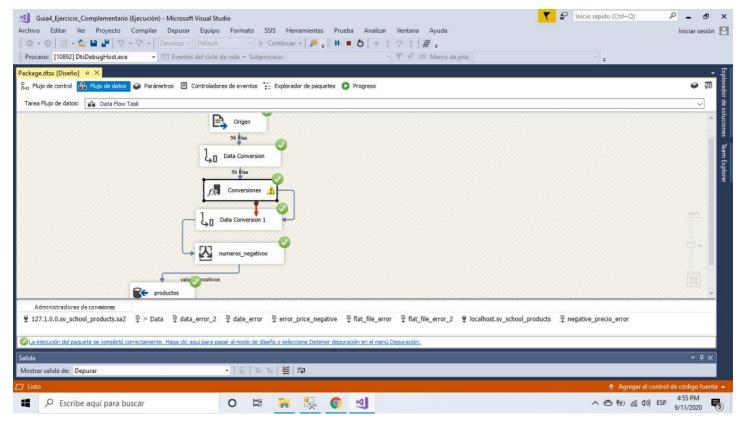
Principalmente al ejecutar el etl pero con el error de componentes al poder ejecutarlo y enviar al destino de la base de datos, da el primer error de salida en el origen de texto plano, esto hace que no pueda dar paso a los datos que se desean enviar a la base de datos destino.

## ¿Qué sucede si cambia el error output por Ignore failure?



Al cambiar de error de salida a ignorar errores o omitir dichos errores. Envía con satisfacción los datos al destino de la base de datos no teniendo inconvenientes desde el recurso principal del origen del texto plano hacia las fases donde pasa los datos que se quieren enviar a la base.

## ¿Qué sucede si configura el Error de Conversion para que realice Ignore failure?



Si en el caso que al editar el derived column y que permita ignorar el error de datos en la salida, se puede observar que si se ejecuta correctamente, pero con un inconveniente al llenado de las filas, los datos aparentemente no se envían con satisfacción a la base. Sabiendo que el etl lo ejecuta correctamente, pero siempre dando un aviso sobre el error en el derived column.

## 2.Al utilizar el For Each Loop Container el flat file error se vuelve muy extenso, lo cual significa que perdemos información ¿Puede recuperarse esa DATA?

Debido a que en muchos casos el error es ocasionado por una incompatibilidad de tipos, es posible aplicar un segundo procedimiento ETL, para identificar las diferencias entre los datos, sabiendo eso podemos clasificar la información de una mejor manera, sin embargo si los datos no son muy consistentes es posible que se sigan presentando errores. Aplicando el For Each Loop, podemos ver que los textos presentes en los archivos no son muy explicativos, por lo tanto no será posible recuperar esa DATA, por lo menos no en su totalidad.

