

Reto 9

Técnicas de Programación y Laboratorio

Como insumo del reto se tiene el archivo *ETH-USD.csv*. Éste incluye el precio de Ethereum en dólares en el último año (Fuente: <https://finance.yahoo.com/quote/ETH-USD/history?p=ETH-USD>). El archivo está compuesto de una línea por día, con excepción de la primera línea que es el encabezado. Cada línea tiene, separados por comas y en el siguiente orden, los siguientes datos:

- La fecha (Date)
- Precio de apertura (Open)
- Precio más alto durante el día (High)
- Precio más bajo durante el día (Low)
- Precio de cierre (Close)
- Precio de cierre ajustado (Adj Close)
- Volumen transado durante el día (Volume)

Visualice el archivo, primero usando Microsoft Excel y luego usando Notepad++ o cualquier otro editor de texto plano (no guarde cambios).

Desarrolle una aplicación que realice las siguientes actividades:

1. Lea el archivo de los precios de Ethereum línea por línea, y a medida que vaya leyendo, escriba en otro archivo línea por línea lo siguiente:
 - La fecha del día que se está leyendo.
 - Un tabulador
 - Un concepto respecto al precio de apertura (sin comillas):
 - “MUY BAJO”, si el precio es menor que \$1.200.
 - “BAJO”, si el precio es mayor o igual que \$1.200 y menor que \$2.100.
 - “MEDIO”, si es mayor o igual que \$2.100 y menor que \$3.100.
 - “ALTO”, si es mayor o igual que \$3.100 y menor que \$4.600.
 - “MUY ALTO”, si es mayor o igual que \$4.600.
2. Halle el promedio y la desviación estándar de los precios de cierre de Ethereum durante el año.
3. Lea el archivo línea por línea y averigüe los siguientes datos:
 - El volumen transado más alto de Ethereum durante el año y su fecha.
 - El volumen transado más bajo de Ethereum durante el año y su fecha.

Usando los elementos de la programación funcional presentes en Java, realice los siguientes ejercicios:

4. Una función que reciba una lista de números de tipo `double` y retorne otra lista de números del mismo tipo, siendo estos la raíz cuadrada de los anteriores, usando *streams*.
5. Una función que reciba un conjunto (`Set`) de cadenas de caracteres y devuelva una lista con las cadenas de caracteres recibidas que cuenten con más de 5 caracteres, usando *streams*.