Local Store: 4ª Fase

Hasta el momento, el sistema de gestión de ventas ha utilizado una base de datos única que almacenaba de forma temporal toda la información relativa a las ventas realizadas y los envíos. Cada año, para evitar un deterioro en el rendimiento del sistema, se realizaba un proceso de archivado por el cual se exportaba la información más antigua a ficheros externos de modo que se redujese la carga de información en la base de datos. Cada fichero de exportación tiene una primera línea de cabecera que indica el tipo de dato que corresponde a cada columna y seguidamente vienen los datos. Por cada línea se almacena una venta y los datos de la venta separados por tabuladores.

No se prevé incluir los antiguos datos en el nuevo sistema debido al coste de integración. No obstante, se ha decidido crear un sistema de consultas que permita acceder a la información de mayor relevancia. Para tal fin, se ha almacenado los ficheros de exportación en un clúster Hadoop que permite realizar consultas de forma eficiente cuando es necesario recuperar información de ese periodo. Se desea implementar una serie de consultas mapreduce que permitan obtener los informes deseados.

Requisitos de la práctica (Consultas)

Se entregara dos archivos por cada consulta, uno para el mapper y otro para el reducer. Los nombres de los archivos deberán ser mX.py para el mapper y rX.py para el reducer donde X se corresponde con el número de consulta.

El archivo debe poder ser ejecutable desde el terminal, por lo que deberá comenzar con la línea #!/usr/bin/python (o la ruta correspondiente al ejecutable de python).

El fichero localstore.data contendrá el repositorio de información semi-estructurada sobre la que se realizarán las consultas. Se simulará realizar las consultas sobre clúster a través del archivo localstore.data. En ningún caso será necesario realizar las consultas sobre Hadoop. Para simular el mapreduce se utilizará el siguiente comando en un sistema con kernel de Linux.

```
cat testfile.txt | ./mapper.py | sort | ./reducer.py
```

En Windows existen similares a cat y sort que se pueden utilizar.

La librería sys de Python es necesaria para acceder a la información que se pasa por stdin tanto al mapper como al reducer: sys.stdin

Se ha de tener en cuenta que es posible que existan errores en los datos proporcionados. En caso de encontrar un error se deberá:

- Más o menos columnas: Se salta la fila.
- Cadenas de texto donde se espera valores numéricos: Se salta la fila.
- Espacios en blanco al principio y al final de una cadena: Cadenas de texto con mismo contenido con o sin espacios antes y/o después de la cadena deben ser equivalentes.

Consultas

1. Facturación total de un proveedor.
2. Facturación media por mes de un proveedor.
3. Incremento por año (en porcentaje) de la facturación de un proveedor.
4. Top tres proveedores que más gastan en envíos en un año determinado y coste total de los envíos para cada proveedor (Coste de envíos: Tipo 1: 10€, Tipo 2: 5€, Tipo 3: 3€.)
5. Ventas de mayor y menor valor realizado en un año determinado y proveedores que la han realizado.
6. Densidad media de las ventas con envío de tipo 1.
7. Tipos de envíos realizados por un proveedor en un mes determinado de un año determinado.
8. Envíos (origen – destino) con más tráfico en un año determinado.
9. Ciudad con más envíos. Entrada de envíos más salida de envíos.
10. Balance de facturación para una ciudad determinada. Ventas cuyo envío se realiza desde una ciudad menos las ventas enviadas a dicha ciudad.

Normativa de realización, entrega y evaluación de la práctica:

- La práctica se realizará y entregará en grupos de hasta dos integrantes.
- La práctica se realizará en python y haciendo uso de un terminal bash.
- La entrega se compondrá de un único fichero ZIP con cada uno de los mappers y reducers de cada consulta.
- Se considerará suspensa toda práctica cuyo fichero comprimido no contenga los ficheros fuente.
- La entrega deberá hacerse mediante el campus virtual antes del domingo 3 de enero de 2021 a las 23:59 horas (hora peninsular en España).
- Las prácticas entregadas fuera de plazo, serán calificadas sobre 9. Por cada día de retraso en la entrega se reducirá el rango de calificación en 0,2 puntos.
- La entrega se compondrá de un único fichero .py renombrado con el número de la práctica P#, seguido del número del grupo G#, y finalmente seguido con el nombre y el primer apellido de los alumnos integrantes del grupo, separados mediante guiones bajos '_'.

  Ejemplo: *P4_G25_Quijote_de_la_Mancha-Sancho_Panza.zip*
- Cualquier sospecha de COPIA entre dos o más prácticas o de código obtenido en internet derivará en la calificación de 0 para todos los alumnos involucrados en la evaluación en curso y la siguiente. En caso de que el alumno tenga duda de que el código pueda ser susceptible de ser entendido como copia, consultar con el profesor antes de la entrega.