

1. Instrucciones para ejecutar el programa.

- a. Ejecutar cada una de las celdas del archivo de Jupyter en el orden en el que se encuentran actualmente. Cada una de las celdas de código se encuentran bien documentadas y cuentan con un encabezado que indica el tipo de funciones que se encuentran en cada celda.
- b. Luego de ejecutar la celda de código que instala PySpark, es necesario ir a la opción **Entorno de ejecución**, y darle click a la opción **Reiniciar entorno de ejecución**. Si no se hace este paso, no se van a generar los archivos CSV y durante la ejecución del programa principal se va a dar un error. Este error que se presenta es un error que ocurre por Colab.
- c. Existe una función llamada main() para ejecutar el programa principal. Para correr el programa principal es necesario haber ejecutado las celdas de código que se encuentran previamente.
- d. Para ejecutar las pruebas unitarias es igual de importante haber ejecutado todas las celdas de código en el orden en el que se encuentran actualmente en el archivo.
- e. Para ejecutar las pruebas unitarias solamente se debe ejecutar la celda de código.

2. Consideraciones:

- a. El código y los archivos JSON de la TP2 se encuentra en el siguiente repositorio: https://github.com/fmejias/CienciasDeLosDatosTEC/tree/master/BigData/Tareas/Tarea_2
- b. De acuerdo a la discusión en la última clase, el archivo se corrió **únicamente con Google Colab**. En el archivo de Jupyter viene todo lo necesario para lograr correr pytest en Google Colab. El archivo nunca se corrió en un ambiente local, solamente se corrió en Google Colab.
- c. Si en algún momento algún error de Java, se debe ir a la opción Entorno de ejecución, y darle click a la opción **Reiniciar entorno de ejecución**. Este error es totalmente independiente del código escrito y pasa muy rara vez.
- d. Si en algún momento al correr una celda con UTs aparece un error, se debe volver a correr la celda. Este error lo produce Colab, no es generado por el código escrito. Y puede ocurrir porque se siguen generando CSV pero con los datos de prueba que se encuentran en memoria
- e. En el repositorio de Github se pueden ver los resultados generados con solo abrir el archivo .ipynb