Tarea #1 - Big Data

Objetivo

Introducir a los estudiantes al uso de operaciones de Apache Spark para cargar e integrar datos a través del uso de pytest.

Resultados esperados

Para esta asignación se espera que los estudiantes concluyan dos entregables relacionados:

- Un programa principal que dada la información sobre estudiantes de una universidad, cursos ofrecidos y notas retorne los dos mejores promedios ponderados por cada carrera.
- Una serie de pruebas unitarias que permitan corroborar la correctitud de las diferentes funciones internas al programa.

Entrega: Archivo comprimido con código y PDF en TEC Digital a más tardar el 27 de noviembre de 2019 a las 11:59 PM

Datos de entrada

Asumiremos que existen 3 entidades cada una con los siguientes atributo:

1. Estudiante
    a. Número de carnet (numérico)
    b. Nombre completo (string)
    c. Carrera (string)
2. Curso
    a. Código de curso (podemos asumir un identificador numérico)
    b. Créditos (numérico)
    c. Carrera (string)
3. Nota
    a. Número de carnet de un estudiante (numérico)
    b. Código de curso (string)
    c. Nota (numérico)

Para la ejecución del programa principal, los estudiantes deberán proveer 3 archivos en formato CSV (separados por comas) con suficientes datos para ejemplificar la correcta funcionalidad.

Los archivos no deben llevar fila de encabezado y las columnas deben llevar el mismo orden mencionado en cada uno de los 3 apartados anteriores.

Para efectos de simplificación, asumimos que no existe el concepto de semestre y que un estudiante puede llevar un curso cualquier cantidad de veces.

Programa principal (20 puntos)

Se espera que los estudiantes entreguen un manual en PDF con las instrucciones para ejecutar el programa principal. Idealmente esto debería realizarse con una simple llamada a "spark-submit programaestudiante.py estudiante.csv curso.csv nota.csv"

Cualquier detalle necesario para la ejecución debe agregarse en este documento. La imposibilidad de ejecución del programa impedirá la obtención de los puntos.

Pruebas esperadas

Para realizar las pruebas unitarias se espera que los estudiantes piensen en las diferentes partes necesarias para conseguir el objetivo final. Éstas podrán arrancar de datos que se encuentren en memoria, asumiendo que el código deberá ser suficientemente modular para que el programa principal simplemente llame a funciones reutilizables que son probadas en diferentes pruebas unitarias.

Los estudiantes deberán diseñar sus propias pruebas unitarias, utilizando la discusión en clase como base para guiar su diseño. Para efectos de evaluación se espera que haya suficientes pruebas para probar las diferentes áreas funcionales:

- **Unión de los datos.** El primer paso debería ser unir los 3 conjuntos de datos diferentes. Debería existir funciones que solamente se encarguen de esta parte. Nótese que la unión de los datos no necesariamente es trivial. Por ejemplo, es posible que un estudiante no haya matriculado ningún curso aún, que haya llevado el curso múltiples veces o que haya carreras que ningún estudiante ha matriculado **(25 puntos)**

- **Agregaciones parciales.** Con el objetivo final de encontrar los/las mejores estudiantes, deberá realizarse código que se encargue de crear tablas con los resultados por estudiante y posteriormente ponderados por créditos. Se espera que haya pruebas que arranquen de dataframes intermedios ya construidos (i.e. no empezarán desde tener que unirlos) y revisen la correcta agregación de los datos **(35 puntos)**

- **Resultados finales.** Los estudiantes deberán crear pruebas que arranquen de agregaciones ya construidas y pueden retornar el top N de estudiantes por carrera **(20 puntos)**

Las pruebas deben cubrir casos excepcionales. Tanto el profesor y asistente se reservan el derecho de agregar pruebas unitarias adicionales en cada apartado para asegurar el correcto funcionamiento.

Se recuerda a los estudiantes que la nota será completamente derivada de las pruebas unitarias. Deberá ser posible ejecutar las pruebas simplemente al correr el comando pytest en la carpeta que se entrega con el código.