

Big Data – Tarea 2

Manual con las instrucciones para la ejecución del programa.

04/12/2023

Estudiantes: Ricardo Chacón & Gabriel Valentine

Estructura de Archivos

El archivo comprimido consta de:

1. Archivos necesarios para el correr el programa. Dentro de este podemos encontrar:
 - `build_image.sh`: Este script se encarga de compilar la imagen del contenedor Docker con la configuración necesaria
 - `run_image.sh`: Proporciona las instrucciones para iniciar el contenedor utilizando la imagen previamente construida.
 - `run_program.sh`: Texto para ejecutar el programa dentro del contenedor.
 - `run_test.sh`: Instrucciones para realizar las pruebas unitarias para verificar el correcto funcionamiento del programa
2. Cinco archivos JSON: `persona1.json`, `persona2.json`, `persona3.json`, `persona4.json` y `persona5.json`. Los archivos corresponden a los datos para el programa según el formato indicado en las instrucciones de la tarea.
3. `main.py`: punto de entrada principal para ejecutar el programa. Coordina la ejecución del programa, utiliza funciones del módulo `functions` para realizar operaciones específicas en datos y se encarga de interactuar con Spark para procesar y analizar datos.
4. `functions.py`: contiene funciones utilitarias y operaciones relacionadas con el manejo de datos y consultas en Spark según las instrucciones de la tarea.
5. `test_unit_tests.py`: contiene pruebas unitarias para las funciones clave en el archivo `functions.py`. Estas pruebas aseguran que las funciones realicen las operaciones previstas y devuelvan los resultados esperados.
6. `confest.py`: entorno de prueba de `pytest` se utiliza para definir fixtures que pueden ser compartidas entre múltiples archivos de prueba.

Procedimiento para ejecutar el Programa:

1. Construir la Imagen del Contenedor:

Ejecutar el script `build_image.sh` para compilar la imagen del contenedor Docker.

2. Iniciar el Contenedor:

Ejecutar el script `run_image.sh` para iniciar el contenedor con la imagen construida.

3. Ejecutar el Programa:

Acceder al contenedor.

Ejecutar el script `run_program.sh` para ejecutar el programa dentro del contenedor.

Procedimiento para ejecutar Pruebas Unitarias:

1. Construir la Imagen del Contenedor:

Ejecutar el script `build_image.sh` para compilar la imagen del contenedor Docker.

2. Iniciar el Contenedor:

Ejecutar el script `run_image.sh` para iniciar el contenedor con la imagen construida.

3. Ejecutar Pruebas Unitarias:

Ejecutar el script `run_test.sh` para realizar pruebas unitarias y verificar el correcto funcionamiento del programa.

Nota: Asegurarse de seguir el orden correcto de los pasos y verificar que los scripts estén configurados correctamente.

Visualización de Resultados

- Después de ejecutar el programa, los resultados se mostrarán en la consola. Además, se generarán una serie de archivos en formato csv con su respectivo identificador que tendrá los resultados del programa.
- Para el caso de las pruebas, cuando se ejecuta una prueba unitaria, el framework de pruebas generalmente compara los resultados obtenidos por la función bajo prueba con los resultados esperados. Si los resultados coinciden, la prueba se considera exitosa; de lo contrario, se considera fallida.
- Recomendaciones:

Asegurarse de seguir el orden correcto de los pasos y verificar que los scripts estén configurados correctamente. Si una prueba falla, revise el código correspondiente en `functions.py` y realice las correcciones.