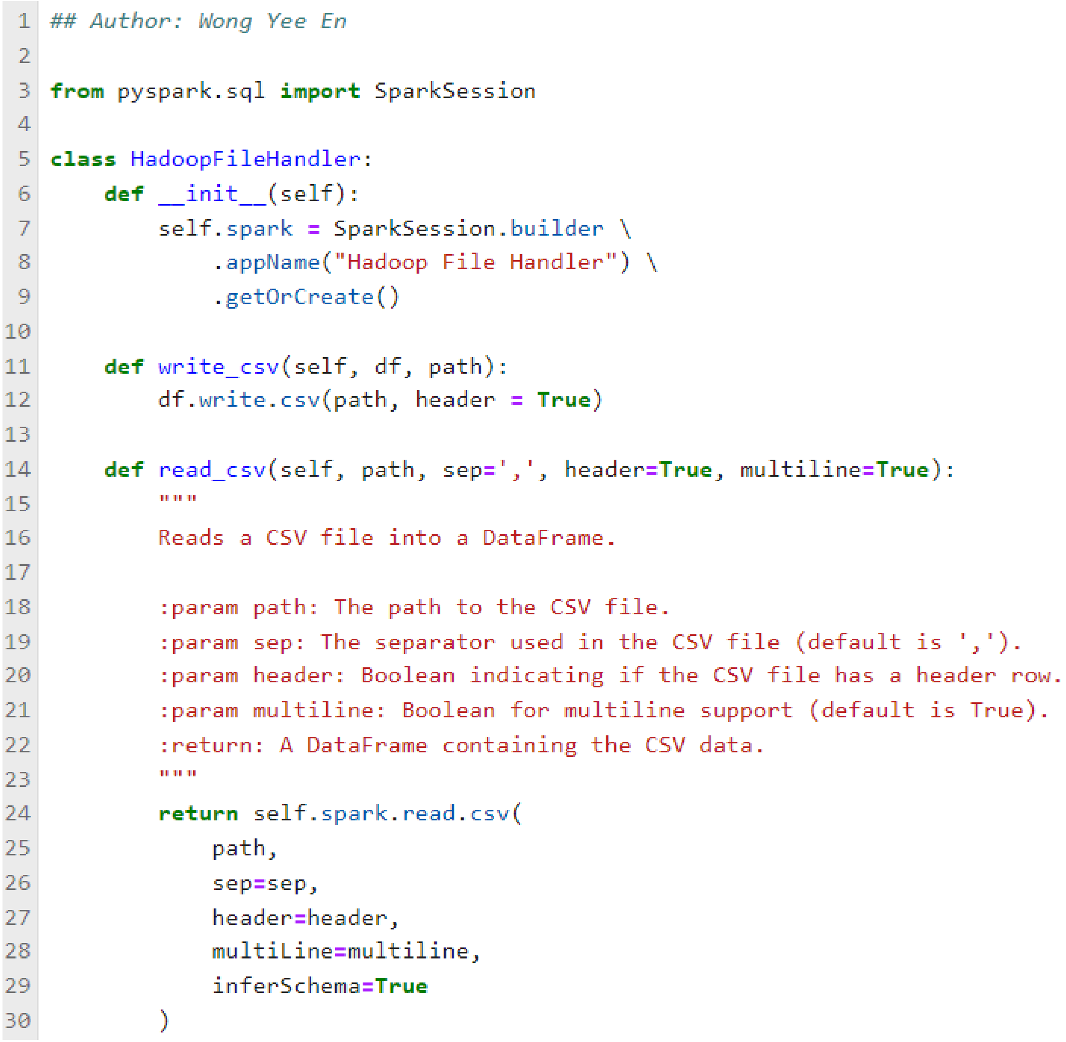
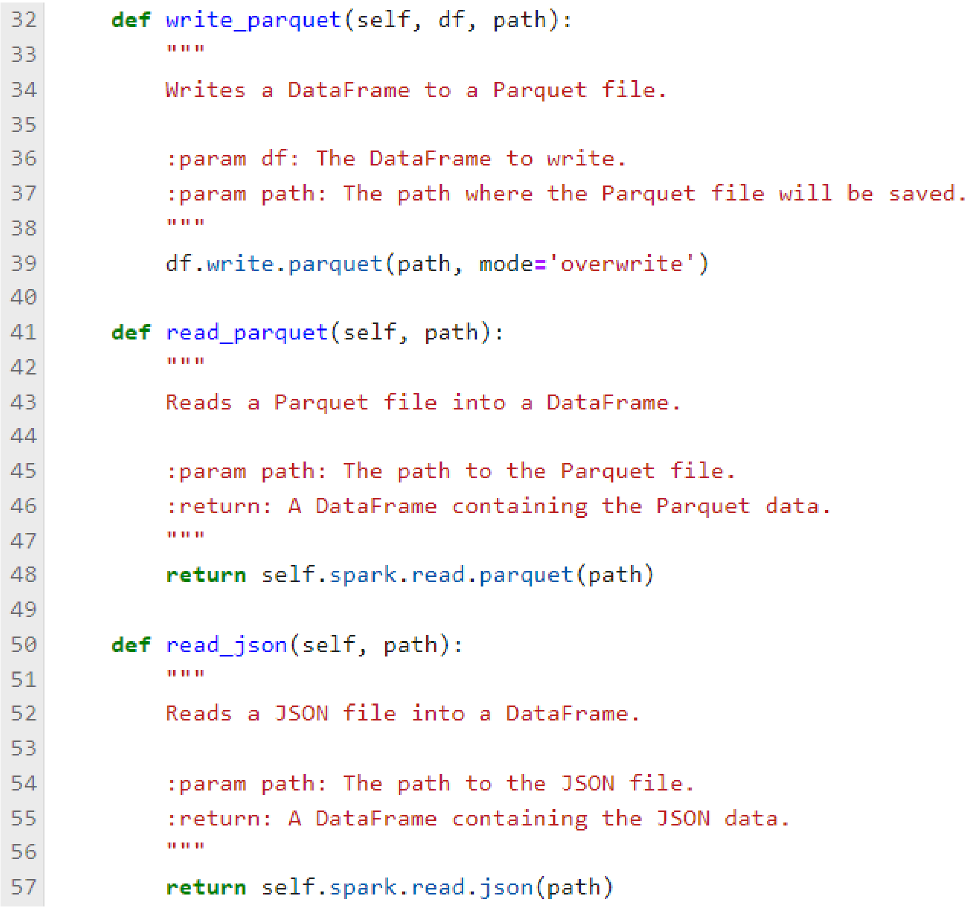
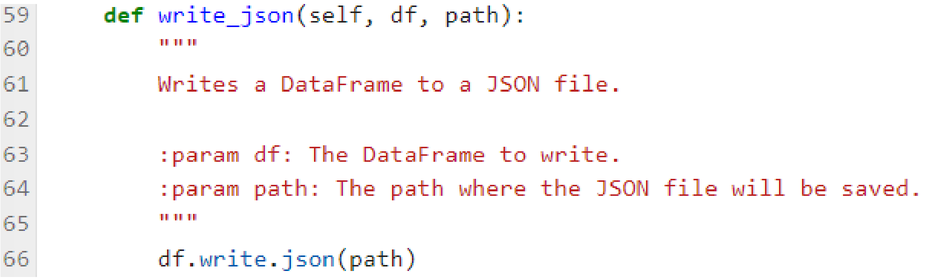
Hadoop File Handler Class



```python
## Author: Wong Yee En

from pyspark.sql import SparkSession

class HadoopFileHandler:
    def __init__(self):
        self.spark = SparkSession.builder \
            .appName("Hadoop File Handler") \
            .getOrCreate()

    def write_csv(self, df, path):
        df.write.csv(path, header = True)

    def read_csv(self, path, sep=',', header=True, multiline=True):
        """
        Reads a CSV file into a DataFrame.

        :param path: The path to the CSV file.
        :param sep: The separator used in the CSV file (default is ',').
        :param header: Boolean indicating if the CSV file has a header row.
        :param multiline: Boolean for multiline support (default is True).
        :return: A DataFrame containing the CSV data.
        """
        return self.spark.read.csv(
            path,
            sep=sep,
            header=header,
            multiLine=multiline,
            inferSchema=True
        )
```



```python
    def write_parquet(self, df, path):
        """
        Writes a DataFrame to a Parquet file.

        :param df: The DataFrame to write.
        :param path: The path where the Parquet file will be saved.
        """
        df.write.parquet(path, mode='overwrite')

    def read_parquet(self, path):
        """
        Reads a Parquet file into a DataFrame.

        :param path: The path to the Parquet file.
        :return: A DataFrame containing the Parquet data.
        """
        return self.spark.read.parquet(path)

    def read_json(self, path):
        """
        Reads a JSON file into a DataFrame.

        :param path: The path to the JSON file.
        :return: A DataFrame containing the JSON data.
        """
        return self.spark.read.json(path)
```



```python
    def write_json(self, df, path):
        """
        Writes a DataFrame to a JSON file.

        :param df: The DataFrame to write.
        :param path: The path where the JSON file will be saved.
        """
        df.write.json(path)
```