

Redis Handler Class

```
1  ## Author : Yam Jason
2  import redis
3  import json
4  from pyspark.sql import Row
5  from datetime import datetime
6  from pyspark.sql.functions import col, date_format
7
8  class RedisHandler:
9      """
10     A class to handle storing and retrieving data from Redis.
11     """
12     def __init__(self, host='localhost', port=6379, db=0):
13         """
14         Initializes the Redis client.
15
16         Parameters:
17         - host: str, Redis server hostname.
18         - port: int, Redis server port.
19         - db: int, Redis database index.
20         """
21         self.redis_client = redis.StrictRedis(host=host, port=port, db=db)
22
23     def store_dataframe(self, df, key_prefix="row"):
24         """
25         Stores a PySpark DataFrame in Redis as JSON strings.
26
27         Parameters:
28         - df: PySpark DataFrame, the DataFrame to store.
29         - key_prefix: str, prefix for the Redis keys.
30         """
31         # Convert the Date column to a string format in the DataFrame
32         df = df.withColumn("Date", date_format(col("Date"), "yyyy-MM-dd"))
33
34         # Convert cleaned DataFrame to a list of dictionaries
35         df_list = df.collect()
36         data_to_store = [row.asDict() for row in df_list]
37
38         # Store each row in Redis as a JSON string
39         for i, data in enumerate(data_to_store):
40             self.redis_client.set(f"{key_prefix}:{i}", json.dumps(data))
41
42         print("Data stored in Redis successfully.")
43         return len(data_to_store)
```

```

45 def load_data(self, num_rows, key_prefix="row"):
46     """
47     Loads data from Redis and converts it to a list of dictionaries.
48
49     Parameters:
50     - num_rows: int, the number of rows to load.
51     - key_prefix: str, prefix for the Redis keys.
52
53     Returns:
54     - list of dictionaries representing the loaded data.
55     """
56     data = []
57     for i in range(num_rows):
58         json_data = self.redis_client.get(f"{key_prefix}:{i}")
59         if json_data:
60             data.append(json.loads(json_data))
61
62     # Convert the date strings back to date objects (if needed)
63     for item in data:
64         if 'Date' in item and item['Date']:
65             item['Date'] = datetime.strptime(item['Date'], "%Y-%m-%d").date()
66
67     return data
68
69 def convert_to_dataframe(self, loaded_data, spark_session):
70     """
71     Converts the loaded data from Redis into a PySpark DataFrame.
72
73     Parameters:
74     - loaded_data: list of dictionaries, the data loaded from Redis.
75     - spark_session: SparkSession object to create DataFrame.
76
77     Returns:
78     - PySpark DataFrame.
79     """
80     df = spark_session.createDataFrame([Row(**item) for item in loaded_data])
81     return df

```