
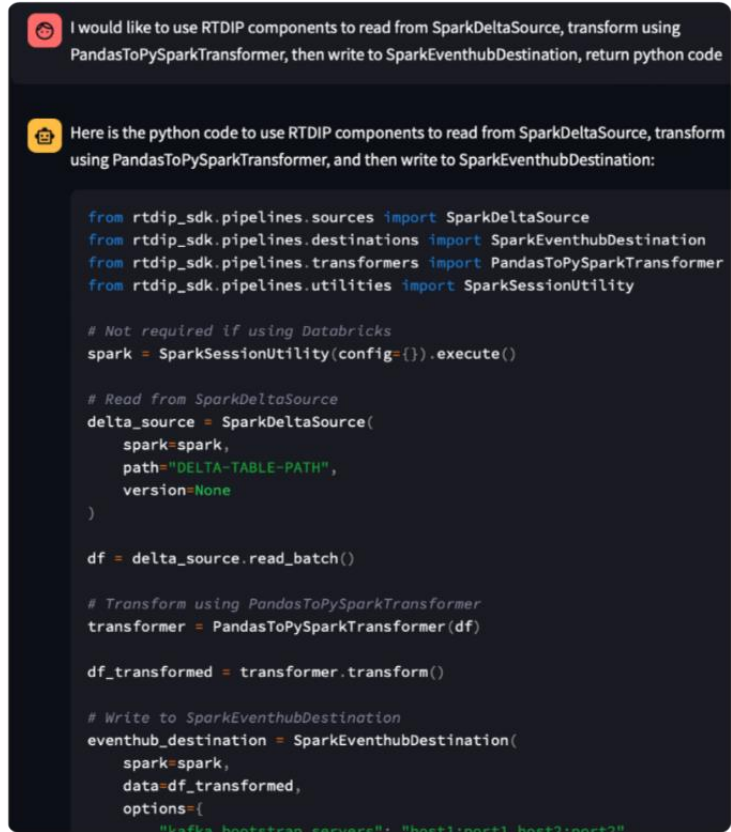


# AMOS WS23/24 Project 5

## Pipeline Configuration Chat AI

Project name	Pipeline Configuration Chat AI
Project mission	Our mission is to develop a cutting-edge tool that seamlessly blends chat AI technology with the most robust data processing capabilities of RTDIP. The project is dedicated to creating a platform where users can effortlessly configure data pipelines using a conversational interface.
Industry partner	Shell
Team logo	
Project summary	Our Chatbot is based on the RTDIP framework, an open-source solution originally developed by Shell. Using it, data from real-time sensors, like wind or temperature sensor, can be extracted, transformed, and saved, all securely. To interact with RTDIP, up until now, data engineers had to write long configuration files, sometimes multiple of those a day. Our solution, ChatRTDIP simplifies the process by enabling the creation of configuration files using only natural text input. This has the potential to save a lot of time for data teams.
Project illustration	 <p>I would like to use RTDIP components to read from SparkDeltaSource, transform using PandasToPySparkTransformer, then write to SparkEventhubDestination, return python code</p> <p>Here is the python code to use RTDIP components to read from SparkDeltaSource, transform using PandasToPySparkTransformer, and then write to SparkEventhubDestination:</p> <pre>from rtdip_sdk.pipelines.sources import SparkDeltaSource from rtdip_sdk.pipelines.destinations import SparkEventhubDestination from rtdip_sdk.pipelines.transformers import PandasToPySparkTransformer from rtdip_sdk.pipelines.utilities import SparkSessionUtility  # Not required if using Databricks spark = SparkSessionUtility(config={}).execute()  # Read from SparkDeltaSource delta_source = SparkDeltaSource(     spark=spark,     path="DELTA-TABLE-PATH",     version=None )  df = delta_source.read_batch()  # Transform using PandasToPySparkTransformer transformer = PandasToPySparkTransformer(df)  df_transformed = transformer.transform()  # Write to SparkEventhubDestination eventhub_destination = SparkEventhubDestination(     spark=spark,     data=df_transformed,     options={         "kafka.bootstrap.servers": "host1:port1,host2:port2"</pre>

Team photo	
Project repository	<a href="https://github.com/amosproj/amos2023ws05-pipeline-config-chat-ai">https://github.com/amosproj/amos2023ws05-pipeline-config-chat-ai</a>