Task 1: Product Inventory Data Ingestion

Sample CSV Data

EmployeeID	Date	CheckInTime	CheckOutTime	HoursWorked
E001	2024-03-01	09:00	17:00	8
E002	2024-03-01	09:15	18:00	8.75
E003	2024-03-01	08:45	17:15	8.5
E004	2024-03-01	10:00	16:30	6.5
E005	2024-03-01	09:30	18:15	8.75

dbutils.fs.cp('file:/Workspace/Shared/employee_attendance.csv',

'dbfs:/FileStore/employee_attendance.csv')

1. **Load CSV Data**:

```
from pyspark.sql import SparkSession
from pyspark.sql.utils import AnalysisException
import logging
spark = SparkSession.builder \
    .appName("Employee Attendance Ingestion") \
    .getOrCreate()

file_path = 'dbfs:/FileStore/employee_attendance.csv'
logging.basicConfig(level=logging.INFO)

try:
    attendance_df = spark.read.option("header", "true")/
            .csv("dbfs:/FileStore/employee_attendance.csv")
    attendance_df.write.format("delta").mode("overwrite").save("/mnt/delta/attendance")
except FileNotFoundError:
     print("CSV file is missing.")
except Exception as e:
    print(f"Error during ingestion: {e}")
```

Task 2: Data Cleaning

```
from pyspark.sql.functions import col, unix_timestamp

cleaned_df = attendance_df.filter(col("CheckInTime").isNotNull() &
    col("CheckOutTime").isNotNull())

cleaned_df = cleaned_df.withColumn("HoursWorked",

  (unix_timestamp(col("CheckOutTime"), 'HH:mm') - unix_timestamp(col("CheckInTime"),
    'HH:mm')) / 3600)

cleaned_df.write.format("delta").mode("overwrite").save("/mnt/delta/cleaned_attendance")
```

Task 3: Attendance Summary

```
from pyspark.sql.functions import sum

monthly_summary_df = cleaned_df.groupBy("EmployeeID").agg(sum("HoursWorked")/

                                                    .alias("TotalHoursWorked"))

overtime_df = cleaned_df.filter(col("HoursWorked") > 8)

monthly_summary_df.write.format("delta").mode("overwrite")/

                                                .save("/mnt/delta/attendance_summary")

overtime_df.write.format("delta").mode("overwrite").save("/mnt/delta/overtime_summary")
```

Task 4: Create an Attendance Pipeline

```
def attendance_pipeline():

  try:

    attendance_df = spark.read.option("header",
    "true").csv("/path/to/employee_attendance.csv")

    attendance_df.write.format("delta").mode("overwrite").save("/mnt/delta/attendance")

    cleaned_df = attendance_df.filter(col("CheckInTime").isNotNull() &
    col("CheckOutTime").isNotNull())

     cleaned_df = cleaned_df.withColumn("HoursWorked",

     (unix_timestamp(col("CheckOutTime"), 'HH:mm') - unix_timestamp(col("CheckInTime"),
    'HH:mm')) / 3600)

     cleaned_df.write.format("delta").mode("overwrite").save("/mnt/delta/cleaned_attendance")


    monthly_summary_df = cleaned_df.groupBy("EmployeeID").agg(sum("HoursWorked")/

                                                    .alias("TotalHoursWorked"))

     overtime_df = cleaned_df.filter(col("HoursWorked") > 8)
```

```
    monthly_summary_df.write.format("delta").mode("overwrite")/
                                                    .save("/mnt/delta/attendance_summary")

    overtime_df.write.format("delta").mode("overwrite").save("/mnt/delta/overtime_summary")

except FileNotFoundError:
    print("CSV file is missing.")
except Exception as e:
    print(f"Error in pipeline: {e}")
```

Task 5: Time Travel with Delta Lake

```
attendance_previous_df = spark.read.format("delta").option("versionAsOf",1)/
                                                    .load("/mnt/delta/cleaned_attendance")
spark.sql("DESCRIBE HISTORY delta.`/mnt/delta/cleaned_attendance`").show()
```