**Common Issues & Solutions in E-Commerce Data Pipeline**

This document highlights the key issues encountered during the project and their respective solutions.

**1. Syntax and Typo Errors**

**Issue:**  
Incorrect syntax or typos in SQL queries, Bash commands, or Python scripts can cause failures.

**Solution:**  
✔ Test SQL queries in MySQL or PostgreSQL before adding them to Airflow tasks.  
✔ Double-check file paths, variable names, and command options before execution.  
✔ Enable detailed logging in Airflow to identify and debug syntax errors quickly.

**2. User Access Issues & Read/Write Permissions**

**Issue:**  
Airflow tasks may fail due to insufficient read/write permissions for files and databases.

**Solution:**  
✔ Ensure the necessary directories exist before execution:

bash

mkdir -p ~/SourceInputDir/E\_Commerce\_Data

mkdir -p ~/OutputDir/E\_Commerce\_Data

✔ Set correct permissions:

bash

chmod -R 755 ~/SourceInputDir ~/OutputDir

✔ Grant database privileges:

GRANT ALL PRIVILEGES ON DATABASE airflow\_pipeline TO airflow\_user;

**3. Exception Handling in Bash & Python Scripts**

**Issue:**  
Tasks may be skipped or fail without proper handling, leading to DAG failures.

**Solution:**  
✔ Use try-except blocks in Python scripts to catch and log errors.  
✔ Add set -e in Bash scripts to stop execution on the first error.  
✔ Ensure all critical commands are wrapped in error-handling mechanisms.

**Example: Python Exception Handling**

python

try:

df = pd.read\_csv('data.csv')

except FileNotFoundError as e:

print(f"Error: {e}")

**Example: Bash Exception Handling**

bash

#!/bin/bash

set -e

if [ ! -f data.csv ]; then

echo "Error: data.csv not found"

exit 1

fi

**4. Admin-Level Configuration Issues in MySQL & PostgreSQL**

**Issue:**  
Default configurations in MySQL and PostgreSQL may block remote connections or cause authentication failures.

**Solution:**  
✔ Update **PostgreSQL's pg\_hba.conf** file to allow connections from Airflow.  
✔ Modify **MySQL's my.cnf** to bind the database to 0.0.0.0 if needed.  
✔ Ensure users have the correct authentication method configured.  
✔ Restart database services after making configuration changes.

**PostgreSQL: Allow Airflow User**

# Edit pg\_hba.conf and add the following:

host all airflow\_user 0.0.0.0/0 md5

**Restart PostgreSQL:**

bash

sudo systemctl restart postgresql

**Restart MySQL:**

bash

sudo systemctl restart mysql

**By implementing these solutions, the pipeline runs smoothly without major errors.**