FACULTY OF ENGINEERING AND BASIC SCIENCES
ACADEMIC PROGRAM: DATA ENGINEERING AND ARTIFICIAL INTELLIGENCE

COURSE: ETL (G01)
LAB-1: Introduction to the ETL Process (Hands-on)

1. Introduction

Extract, Transform and Load (ETL) processes are fundamental in Data Engineering and Artificial Intelligence systems. Before data can be analyzed, modeled, or used by intelligent systems, it must be collected from multiple sources, prepared into a consistent format, and stored in a structured repository.

This first laboratory is designed as an **introductory and conceptual hands-on experience**. The focus is **not on complex transformations or performance optimization**, but on understanding the **role and importance of each ETL stage** and how they connect to form a complete data pipeline.

2. Main Learning Objective

The main goal of this laboratory is to help students **understand the ETL process as an end-to-end pipeline**, clearly identifying:

- **Extraction:** Where data comes from and how it is read from different formats
- **Transformation:** How raw data is minimally prepared or standardized
- **Load:** How transformed data is stored and later queried

3. Learning Outcomes

After completing this lab, you will be able to:

- Read data from **CSV, JSON, and XML** file formats.
- Implement basic **data extraction functions** in Python.
- Apply a simple **data transformation** step.
- Save transformed data into a **CSV file** ready for loading.
- Load data into a **relational database**.
- Execute a **basic SQL query** to retrieve information.

4. Dataset Description

The datasets contain information about vehicles collected from different sources and file formats. Each file includes a subset of the following attributes:

- brand
- model
- year
- price
- fuel_type

All datasets represent the same conceptual entity (vehicles) but are stored in different formats.

5. Project Setup

- Step 1: Project Structure

Create a new project folder in your IDE and organize it using the following structure:

```
ETL_Lab_1/
│
├── data/
│  ├── raw/ # Original input files (CSV, JSON, XML)
│  ├── transformed/ # Transformed output files
│
├── src/
│  ├── extract.py # Extraction functions
│  ├── transform.py # Transformation logic
│  ├── load.py # Load to CSV and database
│  ├── main.py # ETL pipeline orchestration
│  ├── log.py # ETL logs
│  └── db.py # Database functions
│
├── logs/
│  └── log_file.txt # Execution logs
│
├── etl_database.db #Database
├── requirements.txt
└── README.md
```

This structure reflects a real-world ETL pipeline organization and could be reused in future labs**.**

- Step 2: Virtual Environment and Libraries

Create and activate a virtual environment, then install the required libraries:

- pandas
- Standard Python libraries: glob, datetime, xml.etree.ElementTree, sqlite3

6. Global Configuration

Define the following global variables in your code:

- log_file = "logs/log_file.txt"
- target_file = "data/processed/transformed_data.csv"

These files will be used across all ETL stages.

7. Task 1: Extraction

In this task, you will extract data from different file formats.

Instructions

1. Implement one extraction function per file type:
   - extract_from_csv(file_to_process)
   - extract_from_json(file_to_process)
   - extract_from_xml(file_to_process)
2. Each function must:
   - Receive the file path as input
   - Return a pandas DataFrame

8. Task 2 – Transformation

In this first laboratory, the transformation stage is intentionally simple.

Transformation rule

- Round the price of each vehicle to **two decimal places**.

Note: The goal is to understand *where* transformations occur in an ETL pipeline, not to perform complex data cleaning.

9. Task 3 – Load

Step 1: Save to CSV

- Save the transformed DataFrame into transformed_data.csv

Step 2: Load into Database

- Load the transformed data into an D**atabase**.
- Create a table named vehicles.

This Database is used to avoid infrastructure complexity and keep the focus on ETL concepts.

10. Task 4 – Basic Query

Create a menu that allows three queries to be made on the data and displays the results, for example:
- Execute an SQL query to: Display **brand, model, year, and price** of vehicles manufactured **after 2015**.

This step demonstrates the **value of ETL**: data from multiple sources can only be queried after being properly extracted, transformed, and loaded.

11. Reflection Questions

Answer the following questions in a short paragraph each:

1. What is the role of each ETL stage in this laboratory?
2. What problems could arise if the transformation step is skipped?
3. Why is it useful to load data into a database instead of keeping multiple raw files?
4. How does this ETL pipeline support future analytics or AI tasks?

12. Deliverables

Submit the following:

- Python source code (src/ folder)
- transformed_data.csv
- Database file (etl_database.db)
- Answers to the reflection questions