Experiment 4 Output

In this experiment, an **ETL process** was designed and implemented to migrate data from operational databases to a **data warehouse**.

Creating Database and Using in MySQL

```
mysql> CREATE DATABASE RetailDataWarehouse;
Query OK, 1 row affected (0.03 sec)

mysql> show databases;
+----------------------+
| Database             |
+----------------------+
| classdb              |
| company              |
| employee             |
| hospital             |
| information_schema   |
| movie                |
| mydatabase           |
| mysql                |
| performance_schema   |
| retaildatawarehouse  |
| root                 |
| shopping             |
| student              |
| sys                  |
| utd                  |
+----------------------+
15 rows in set (0.00 sec)
```

Fig 1: Database in MySQL

Creating Tables

```
mysql> CREATE TABLE unstructured_operational_data (
    ->     id INT AUTO_INCREMENT PRIMARY KEY,
    ->     customer_name VARCHAR(255),
    ->     product_details TEXT,
    ->     purchase_date VARCHAR(50),
    ->     price VARCHAR(50),
    ->     quantity VARCHAR(50),
    ->     address TEXT
    -> );
Query OK, 0 rows affected (0.03 sec)

mysql> INSERT INTO unstructured_operational_data (customer_name, product_details, purchase_date, price,
    -> ('John Doe', 'Laptop - Model A', '2024-02-01', '1000', '1', 'New York, USA'),
    -> ('Jane Smith', 'Smartphone - Model X', '2024-02-02', '700', '2', 'Los Angeles, USA'),
    -> ('Ali Khan', 'Tablet - Model B', '02-Feb-24', '450', '1', 'Mumbai, India'),
    -> ('Emily Davis', 'Laptop - Model A', '2024/02/03', '1000', '1', 'London, UK'),
    -> ('Carlos Ruiz', 'Headphones - Wireless', '2024-02-04', '150', '3', 'Madrid, Spain'),
    -> ('Sophia Lee', 'Smartwatch - Series 6', '2nd Feb 2024', '300', '1', 'Seoul, South Korea'),
    -> ('Michael Brown', 'Monitor - 27 inch', '2024-02-05', '250', '2', 'Berlin, Germany'),
    -> ('Linda Green', 'Keyboard - Mechanical', '05-02-2024', '80', '5', 'Sydney, Australia'),
    -> ('David White', 'Mouse - Gaming', '6/2/2024', '60', '4', 'Toronto, Canada'),
    -> ('Olivia Black', 'Laptop - Model C', '2024-02-07', '1200', '1', 'Paris, France'),
    -> ('Ethan Miller', 'Smartphone - Model Y', 'Feb 08 2024', '800', '2', 'Dubai, UAE'),
    -> ('Mia Wilson', 'Headphones - Wired', '9-2-2024', '100', '2', 'Rome, Italy'),
    -> ('Lucas Adams', 'Tablet - Model D', '10th Feb 2024', '500', '1', 'Singapore'),
    -> ('Ava Scott', 'Laptop - Model B', '11-02-24', '1100', '1', 'Hong Kong'),
    -> ('Daniel Hall', 'Smartwatch - Series 7', '12/02/2024', '350', '1', 'Bangkok, Thailand'),
    -> ('Emma Carter', 'Monitor - 24 inch', 'Feb 13, 2024', '200', '2', 'Amsterdam, Netherlands'),
    -> ('Noah Phillips', 'Keyboard - RGB', '14-02-2024', '90', '3', 'Zurich, Switzerland'),
    -> ('Liam Robinson', 'Mouse - Wireless', '2024-02-15', '75', '2', 'Tokyo, Japan'),
    -> ('Charlotte Martinez', 'Laptop - Model D', '16/02/24', '1300', '1', 'Beijing, China'),
    -> ('James Anderson', 'Smartphone - Model Z', 'Feb 17 2024', '900', '2', 'Mexico City, Mexico'),
    -> ('Isabella Thomas', 'Headphones - Bluetooth', '18-02-2024', '120', '3', 'Cairo, Egypt'),
    -> ('Benjamin Harris', 'Tablet - Model E', '2024-02-19', '550', '1', 'Johannesburg, South Africa'),
    -> ('Amelia Martin', 'Laptop - Model E', 'Feb 20 2024', '1400', '1', 'Kuala Lumpur, Malaysia'),
    -> ('William Thompson', 'Smartwatch - Series 8', '21-02-2024', '400', '1', 'Hanoi, Vietnam'),
    -> ('Sophie Walker', 'Monitor - 32 inch', '22-02-2024', '300', '2', 'Wellington, New Zealand');
Query OK, 25 rows affected (0.01 sec)
Records: 25  Duplicates: 0  Warnings: 0
```

Fig 2: Creating Tables and Inserting Data

Extracting and Transforming Data

```
mysql> CREATE TABLE transformed_data AS
    -> SELECT
    ->     id,
    ->     customer_name,
    ->     SUBSTRING_INDEX(product_details, ' - ', 1) AS product_name,
    ->     CASE
    ->         WHEN purchase_date REGEXP '^[0-9]{4}-[0-9]{2}-[0-9]{2}$'
    ->             THEN STR_TO_DATE(purchase_date, '%Y-%m-%d')  -- '2024-02-05'
    ->         WHEN purchase_date REGEXP '^[0-9]{2}/[0-9]{2}/[0-9]{4}$'
    ->             THEN STR_TO_DATE(purchase_date, '%d/%m/%Y')  -- '05/02/2024'
    ->         WHEN purchase_date REGEXP '^[0-9]{2}-[A-Za-z]{3}-[0-9]{2}$'
    ->             THEN STR_TO_DATE(purchase_date, '%d-%b-%y')  -- '05-Feb-24'
    ->         WHEN purchase_date REGEXP '^[0-9]{2}-[0-9]{2}-[0-9]{4}$'
    ->             THEN STR_TO_DATE(purchase_date, '%d-%m-%Y')  -- '05-02-2024'
    ->         WHEN purchase_date REGEXP '^[0-9]{1,2} [A-Za-z]{3} [0-9]{4}$'
    ->             THEN STR_TO_DATE(purchase_date, '%d %b %Y')  -- '5 Feb 2024'
    ->         ELSE NULL
    ->     END AS formatted_date,
    ->     CAST(price AS DECIMAL(10,2)) AS price,
    ->     CAST(NULLIF(TRIM(quantity), '') AS SIGNED) AS quantity,
    ->     address
    -> FROM unstructured_operational_data;
Query OK, 25 rows affected (0.03 sec)
Records: 25  Duplicates: 0  Warnings: 0

mysql> CREATE TABLE fact_sales (
    ->     sale_id INT AUTO_INCREMENT PRIMARY KEY,
    ->     customer_name VARCHAR(255),
    ->     product_name VARCHAR(255),
    ->     sale_date DATE,
    ->     total_price DECIMAL(10,2),
    ->     address TEXT
    -> );
Query OK, 0 rows affected (0.01 sec)
```

Fig 3: Extraction and Transformation of Data

Loading data into warehouse

```
mysql> CREATE TABLE fact_sales (
    ->     sale_id INT AUTO_INCREMENT PRIMARY KEY,
    ->     customer_name VARCHAR(255),
    ->     product_name VARCHAR(255),
    ->     sale_date DATE,
    ->     total_price DECIMAL(10,2),
    ->     address TEXT
    -> );
Query OK, 0 rows affected (0.01 sec)

mysql> INSERT INTO fact_sales (customer_name, produc
    -> SELECT customer_name, product_name, formatted
    -> FROM transformed_data;
Query OK, 25 rows affected (0.01 sec)
Records: 25  Duplicates: 0  Warnings: 0
```

Fig 4: Inserting data into warehouse

Madhurima Rawat DS 42