Advanced Data Management (CMM524)

Laboratory #3: More SQL

1. Aims

- To use SQL commands to retrieve, update, and delete data in a database.

2. Outcomes

In completing this exercise, you should be able to:
- Retrieve data from a database using various SELECT statements.
- Update data using the UPDATE statement.
- Delete data using the DELETE statement.

3. Creating the Database & Tables

Our online shopping domain has the following tables:

Table `product`:

Column	Type	NULL	Key	Extra
productID	varchar(16)	NO	Primary key	
description	varchar(255)	NO		
currentPrice	float	NO		default value 0.0

Table `customer`:

Column	Type	NULL	Key
email	varchar(64)	NO	Primary key
name	varchar(64)	NO	
address	varchar(255)	NO	
country	varchar(32)	NO	

Table `purchase`:

Column	Type	NULL	Key	Extra
orderID	int	NO	Primary key	AUTO INCREMENT
customerEmail	varchar(64)	NO		
productID	varchar(16)	NO		
purchasePrice	float	NO		

Create a new database just for this lab:
- Login as `training`, create a database `lab03`.
  - Note: If you found a database `lab03` in MySQL before you create any, may be someone has used this VM before. You can DROP the database and re-create it from scratch.
- Switch to the database `lab03`.
- Use SQL to create tables `product`, `customer`, and `purchase`. according to the schema described above.

4. Populating the Tables

Populate your tables with the following data:

Products:

Product ID	Description	Current Price
p001	Echo Dot	49.99
p002	Echo	89.99
p003	Echo Show	199.99
p004	Echo Plus	139.99
p005	Wood filament 1.75mm 1kg	29.99
p006	iPhone X 64GB	999
p007	iPad Pro	619
p008	Google Home	99.99
p009	Google Wifi	219.99
p010	Andrex 10 rolls	14.99

Customers:

Email	Name	Address	Country
frodo@shire.net	Frodo Baggins	1 Bag End	The Shire
gandalf@gmail.com	Gandalf	Lorien Gardens	The Shire
aragorn@palace.gd	Aragorn	The Palace	Gondor
legolas@mirkwood.org	Legolas	Woodland	Mirkwood
sauron@mordor.evil	Sauron	Barad-dur	Mordor

Purchases:

Customer email	Product ID	Purchase price
frodo@shire.net	p010	14.99
frodo@shire.net	p010	34.99
gandalf@gmail.com	p010	9.99
gandalf@gmail.com	p007	599
aragorn@palace.gd	p010	18.99
aragorn@palace.gd	p009	1099.99
legolas@mirkwood.org	p010	10.99
legolas@mirkwood.org	p001	40.99
legolas@mirkwood.org	p005	29.99

Notes:
- You can use the MySQL text client interactively to enter the above data but it is liable to mistakes.
- Alternatively you can create a text file with SQL INSERT statements and "source" the file later.

5. Making Queries

Write SQL statements to perform the following tasks:

5.1. Simple Retrieval

- Show all details of all customers.
- Show all product names and their current prices.

5.2. Retrieval with Simple Filtering

- Show details of product with ID `p010`.
- Show all purchases made by Aragorn. (Hint: You can simply filter by Aragorn's email.)
- Show product IDs and names with a current price between £20 and £100.
- Find all products with the word "echo" in their names.

5.3. Retrieval with Ordering

- Show all product names in alphabetic order.
- Show all product names and prices in decreasing order.
- List the countries where the customers are coming from in alphabetical order, with no duplicate.

5.4. Retrieval with Join

- Show all customers with the product IDs they have purchased. (Hint: Join required between `customer` and `purchase` tables.)
- Show all customers with the product name and price they have purchased. (Hint: You may need an `INNER JOIN` followed by another `INNER JOIN`.)

5.5. Retrieval with Aggregation

- Find the average purchase price across all orders.
- Find the total amount over all orders.
- Find the total purchase amount made by customers in "The Shire" only. (Hint: Join required between `purchase` and `customer` tables.)
- Show how many times a product has been ordered, with the product IDs only. (Hint: No join required. Only the `purchase` table is needed.)
- Show all product names which have been purchased and the number of orders on each product. (Hint: Join required between the `purchase` and `product` tables.)

6. Updating Data

Write SQL statements to perform the following tasks:

Simple retrieval:
- Change the price of "`Andrex 19 rolls`" to 18.99.
- Reduce the current price of all "`Echo`" products by 10%.

7. Deleting Data

Write SQL statements to perform the following tasks:

- Delete all purchases where ordered an "Echo" product. (Hint: You can get all product IDs of Echo products first. Then use a "WHERE … IN..." condition.)
- Delete all products with the keyword "Echo".