



**UTM**  
**UNIVERSITI TEKNOLOGI MALAYSIA**

**BACHELOR OF COMPUTER SCIENCE(GRAPHIC AND MULTIMEDIA  
SOFTWARE)**

**SECD2523**

**(DATABASE)**

**SEMESTER 20232024 – 1**

**SECTION 10**

**LAB DML 1 PART 2**

<b>NAME</b>	:	Elisha Bt Afandi
<b>MATRIC NO.</b>	:	A22EC0156

## Section 6 Lesson 4 Exercise 2: Data Manipulation Language

Use DML operations to manage database tables (S6L4 Objective 2)

In this exercise you will populate and work with the data that is stored in the database system.

Part 1- Updating rows to the system

1. Run the following query to view the content of the price\_history table:

```
SELECT start_date, TO_CHAR (start_time, 'HH24:MI:SS'), price, end_date, TO_CHAR  
(end_time, 'HH24:MI')  
FROM price_history;
```

START_DATE	TO_CHAR(START_TIME,'HH24:MI:SS')	PRICE	END_DATE	TO_CHAR(END_TIME,'HH24:MI')
06/17/2017	09:00:00	4.99	-	-
11/25/2016	09:00:00	14.99	01/25/2017	17:00
01/25/2017	17:01:00	8.99	01/25/2017	19:00
01/26/2017	09:00:00	15.99	-	-
02/12/2017	12:30:00	7.99	-	-
04/25/2017	10:10:10	24.99	-	-
05/31/2017	16:35:30	149	-	-

2. Obl is going to update the price of the premium bat so you will need to write a query that will close off the current price by adding the system date values to the end\_date and end\_time fields. To run this query you will need to both match the item number and identify that the end date is null. This ensures that you are updating the latest price.

**UPDATE price\_history**

**SET end\_date = CURRENT\_DATE, end\_time = CURRENT\_TIMESTAMP**

**WHERE itm\_number = 'im01101048' AND end\_date IS NULL;**

Columns	Data	Indexes	Constraints	Grants	Statistics	Triggers	Dependencies	DDL	Sample Queries
+ Insert Row	Columns...	Filter...	Count Rows	Load Data	Download	Refresh			
	START_DATE	START_TIME	PRICE	END_DATE	END_TIME	ITM_NUMBER			
	06/17/2017	06/17/2016	4.99			im01101044			
	11/25/2016	11/25/2016	14.99	01/25/2017	01/25/2017	im01101045			
	01/25/2017	01/25/2017	8.99	01/25/2017	01/25/2017	im01101045			
	01/26/2017	01/26/2017	15.99			im01101045			
	02/12/2017	02/12/2017	7.99			im01101046			
	04/25/2017	04/25/2017	24.99			im01101047			
	05/31/2017	05/31/2017	149	12/10/2023	12/10/2023	im01101048			

3. Rerun the select statement on the price\_history table to ensure that the statement has been executed.

**SELECT \***

**FROM price\_history;**

START_DATE	START_TIME	PRICE	END_DATE	END_TIME	ITM_NUMBER
06/17/2017	06/17/2016	4.99	-	-	im01101044
11/25/2016	11/25/2016	14.99	01/25/2017	01/25/2017	im01101045
01/25/2017	01/25/2017	8.99	01/25/2017	01/25/2017	im01101045
01/26/2017	01/26/2017	15.99	-	-	im01101045
02/12/2017	02/12/2017	7.99	-	-	im01101046
04/25/2017	04/25/2017	24.99	-	-	im01101047
05/31/2017	05/31/2017	149	12/10/2023	12/10/2023	im01101048

4. Insert a new row that will use the current date and time to set the new price of the premium bat to be 99.99.

**INSERT INTO price\_history(itm\_number, price, start\_date, start\_time)**

**VALUES ('im01101048', 99.99, CURRENT\_DATE, CURRENT\_TIMESTAMP);**

```
1 INSERT INTO price_history(itm_number, price, start_date, start_time)
2 VALUES ('im01101048' , 99.99, CURRENT_DATE, CURRENT_TIMESTAMP);
3
```

**Results** Explain Describe Saved SQL History

1 row(s) inserted.

0.01 seconds

5. Rerun the select statement on the price\_history table to ensure that the statement has been executed.

**SELECT \***

**FROM price\_history;**

START_DATE	START_TIME	PRICE	END_DATE	END_TIME	ITM_NUMBER
06/17/2017	06/17/2016	4.99	-	-	im01101044
11/25/2016	11/25/2016	14.99	01/25/2017	01/25/2017	im01101045
01/25/2017	01/25/2017	8.99	01/25/2017	01/25/2017	im01101045
01/26/2017	01/26/2017	15.99	-	-	im01101045
02/12/2017	02/12/2017	7.99	-	-	im01101046
04/25/2017	04/25/2017	24.99	-	-	im01101047
05/31/2017	05/31/2017	149	12/10/2023	12/10/2023	im01101048
12/10/2023	12/10/2023	99.99	-	-	im01101048

## Part 2: Deleting rows from the system

1. Bob Thornberry has contacted Obl to ask that the 83 Barrhill Drive address be removed from the system as he can no longer receive parcels at this address. Write an SQL statement that will remove this address from the system.

**DELETE FROM customers\_addresses**

**WHERE id = 'ca0101';**

```
1  DELETE FROM customers_addresses
2  WHERE id = 'ca0101';
3
```

Results	Explain	Describe	Saved SQL	History
1 row(s) deleted.				
0.03 seconds				

2. Run a select statement on the customers\_addresses table to ensure that the statement has been executed.

**SELECT \***

**FROM customers\_addresses;**

ID	ADDRESS_LINE_1	ADDRESS_LINE_2	CITY	ZIP_CODE	CTR_NUMBER
ca0102	17 Gartsquare Road	Starford	Liverpool	LP89JHK	c00001
ca0103	54 Ropehill Crescent	Georgetown	Star	ST45AGV	c00101
ca0104	36 Watercress Lane	-	Jump	JP23YTH	c01986
ca0105	63 Acacia Drive	Skins	Liverpool	LP83JHR	c00001

