

Lab 2 :

Name : Lim Bo Yuan

No Matric : A22EC0181

Section 6 Lesson 4 Exercise 1: 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 tables.

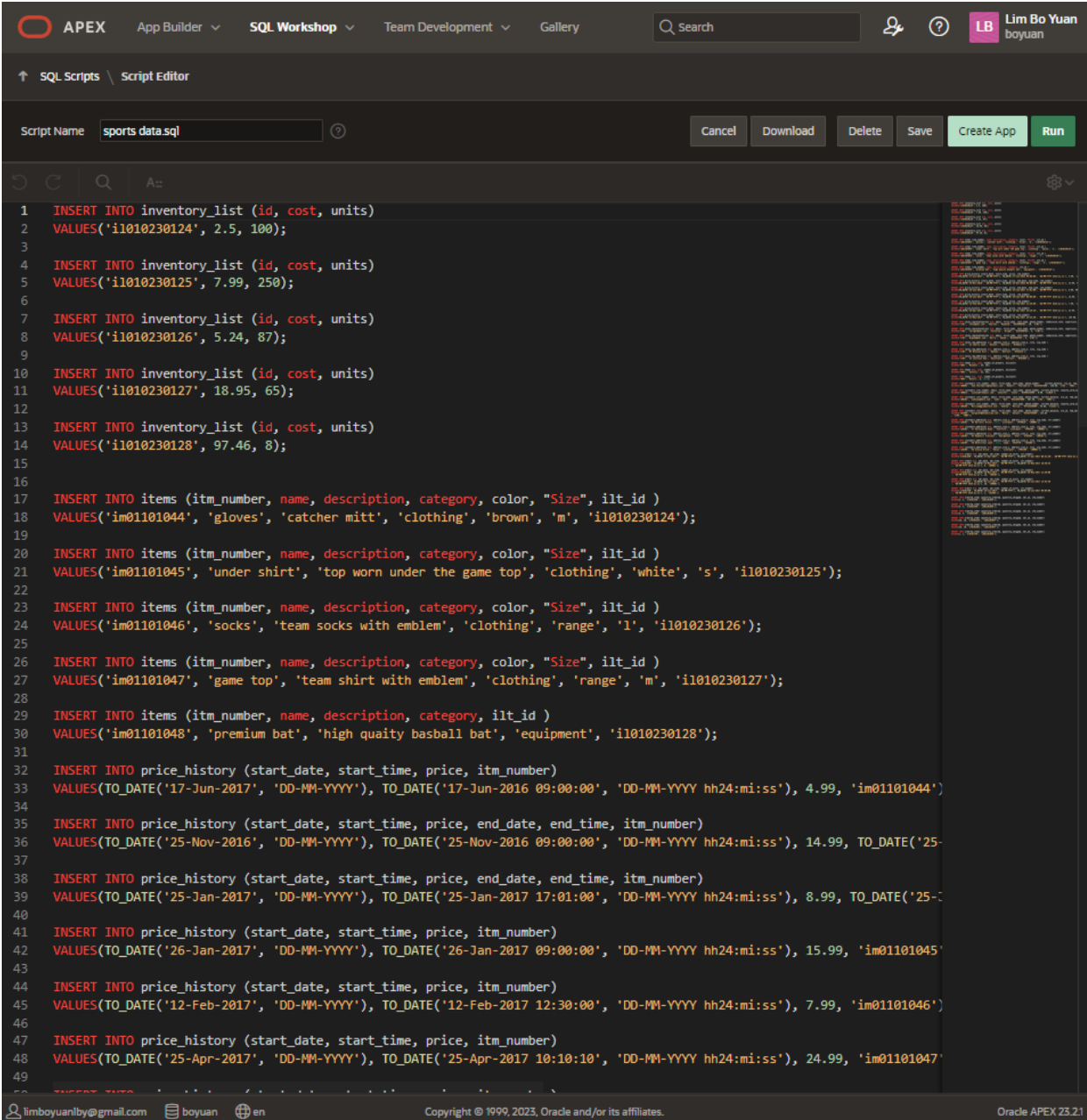
Part 1 : Running a script to populate the tables.

You have to consider the order of the tables when populating them. A table that has a foreign key field cannot be populated before the related table with the primary key.

1. Use the table mapping document and list the order that you would use to populate the tables.
 - a) inventory_list
 - b) items
 - c) price_history
 - d) sales_representatives
 - e) sales_rep_addresses
 - f) teams
 - g) customers
 - h) customers_addresses
 - i) orders
 - j) ordered_items
2. Open the “sports data.sql” and look at the order the data is being added there, does your list match? This file can be found in the Section 6 Lesson 4 interaction (sports data.zip) and must first be extracted.

Yes, the list is matched.

3. Run the “sports data.sql” script in APEX to populate your tables





The screenshot shows the Oracle APEX SQL Workshop interface. At the top, there's a navigation bar with 'APEX', 'App Builder', 'SQL Workshop', 'Team Development', and 'Gallery'. A search bar and user profile 'Lim Bo Yuan' are also present. Below the navigation bar, the 'SQL Scripts' section is active, showing a 'Script Editor' for a file named 'sports data.sql'. The editor contains SQL code for inserting data into 'inventory_list' and 'items' tables, and for the 'price_history' table. The code is as follows:


```
1 INSERT INTO inventory_list (id, cost, units)
2 VALUES('i1010230124', 2.5, 100);
3
4 INSERT INTO inventory_list (id, cost, units)
5 VALUES('i1010230125', 7.99, 250);
6
7 INSERT INTO inventory_list (id, cost, units)
8 VALUES('i1010230126', 5.24, 87);
9
10 INSERT INTO inventory_list (id, cost, units)
11 VALUES('i1010230127', 18.95, 65);
12
13 INSERT INTO inventory_list (id, cost, units)
14 VALUES('i1010230128', 97.46, 8);
15
16
17 INSERT INTO items (itm_number, name, description, category, color, "Size", ilt_id )
18 VALUES('im01101044', 'gloves', 'catcher mitt', 'clothing', 'brown', 'm', 'i1010230124');
19
20 INSERT INTO items (itm_number, name, description, category, color, "Size", ilt_id )
21 VALUES('im01101045', 'under shirt', 'top worn under the game top', 'clothing', 'white', 's', 'i1010230125');
22
23 INSERT INTO items (itm_number, name, description, category, color, "Size", ilt_id )
24 VALUES('im01101046', 'socks', 'team socks with emblem', 'clothing', 'range', 'l', 'i1010230126');
25
26 INSERT INTO items (itm_number, name, description, category, color, "Size", ilt_id )
27 VALUES('im01101047', 'game top', 'team shirt with emblem', 'clothing', 'range', 'm', 'i1010230127');
28
29 INSERT INTO items (itm_number, name, description, category, ilt_id )
30 VALUES('im01101048', 'premium bat', 'high quality baseball bat', 'equipment', 'i1010230128');
31
32 INSERT INTO price_history (start_date, start_time, price, itm_number)
33 VALUES(TO_DATE('17-Jun-2017', 'DD-MM-YYYY'), TO_DATE('17-Jun-2016 09:00:00', 'DD-MM-YYYY hh24:mi:ss'), 4.99, 'im01101044')
34
35 INSERT INTO price_history (start_date, start_time, price, end_date, end_time, itm_number)
36 VALUES(TO_DATE('25-Nov-2016', 'DD-MM-YYYY'), TO_DATE('25-Nov-2016 09:00:00', 'DD-MM-YYYY hh24:mi:ss'), 14.99, TO_DATE('25-
37
38 INSERT INTO price_history (start_date, start_time, price, end_date, end_time, itm_number)
39 VALUES(TO_DATE('25-Jan-2017', 'DD-MM-YYYY'), TO_DATE('25-Jan-2017 17:01:00', 'DD-MM-YYYY hh24:mi:ss'), 8.99, TO_DATE('25-
40
41 INSERT INTO price_history (start_date, start_time, price, itm_number)
42 VALUES(TO_DATE('26-Jan-2017', 'DD-MM-YYYY'), TO_DATE('26-Jan-2017 09:00:00', 'DD-MM-YYYY hh24:mi:ss'), 15.99, 'im01101045')
43
44 INSERT INTO price_history (start_date, start_time, price, itm_number)
45 VALUES(TO_DATE('12-Feb-2017', 'DD-MM-YYYY'), TO_DATE('12-Feb-2017 12:30:00', 'DD-MM-YYYY hh24:mi:ss'), 7.99, 'im01101046')
46
47 INSERT INTO price_history (start_date, start_time, price, itm_number)
48 VALUES(TO_DATE('25-Apr-2017', 'DD-MM-YYYY'), TO_DATE('25-Apr-2017 10:10:10', 'DD-MM-YYYY hh24:mi:ss'), 24.99, 'im01101047')
49
```


The bottom of the interface shows the user's email 'limboyanlby@gmail.com', the name 'boyuan', and the Oracle APEX version '23.2.1'.

4. Check that no errors occurred when you ran the script.


 **APEX**




App Builder 

SQL Workshop 

Team Development 

Gallery

 Search

   **LB** Lim Bo Yuan boyuan

 SQL Scripts \ Results

Script: **sports data.sql** 

Status: **Complete** 

View:  Detail  **Summary** 

Rows:  

Download

row(s) 1 - 15 of 47 

47

Statements Processed

47

Successful

0

With Errors

Part 2- Inserting rows to the system

- 1. Add a new team to the system

id	name	Number_of_players	discount
t004	Jets	10	5

APEX

App Builder

SQL Workshop

Team Development

Gallery

Search

LB

Lim Bo Yuan

boyuan

SQL Commands

SchemaWKSP_BOYUAN

LanguageSQL

Rows10

Clear Command

Find Tables

Save

Run

1

INSERT INTO teams (id, name, number_of_players, discount)

2

VALUES ('t004', 'Jets', 10, 5);

Results

Explain

Describe

Saved SQL

History

1 row(s) inserted.

0.01 seconds

2. Add a new Customer with the following details to the system

ctr number	email	First name	Last name	Phone number	Current balance	Loyalty card number	tem id	sre id
c02001	brianrog@hoootech.com	Brian	Rogers	01654564898	-5	1c4587		

The screenshot shows the APEX SQL Workshop interface. The SQL Commands panel contains the following code:

```
1 INSERT INTO customers (ctr_number ,email, first_name, last_name, phone_number, current_balance, loyalty_card_number)
2 VALUES('c02001','brianrog@hoootech.com','Brian','Rogers','01654564898',-5,'1c4587');
```

The Results panel shows an error message: **ORA-02290: check constraint (WKSP_BOYUAN.MIN_CURRENT_BALANCE) violated**. The execution time is 0.00 seconds.

3. This information violates the check constraint that the current balance must not be less than zero. Change the current balance to 50 and rerun the query.

The screenshot shows the APEX SQL Workshop interface. The SQL Commands panel contains the following code:

```
1 INSERT INTO customers (ctr_number ,email, first_name, last_name, phone_number, current_balance, loyalty_card_number)
2 VALUES('c02001','brianrog@hoootech.com','Brian','Rogers','01654564898',50,'1c4587');
```

The Results panel shows the message: **1 row(s) inserted.** The execution time is 0.01 seconds.

Section 6 Lesson 4 Exercise 2: Data Manipulation Language

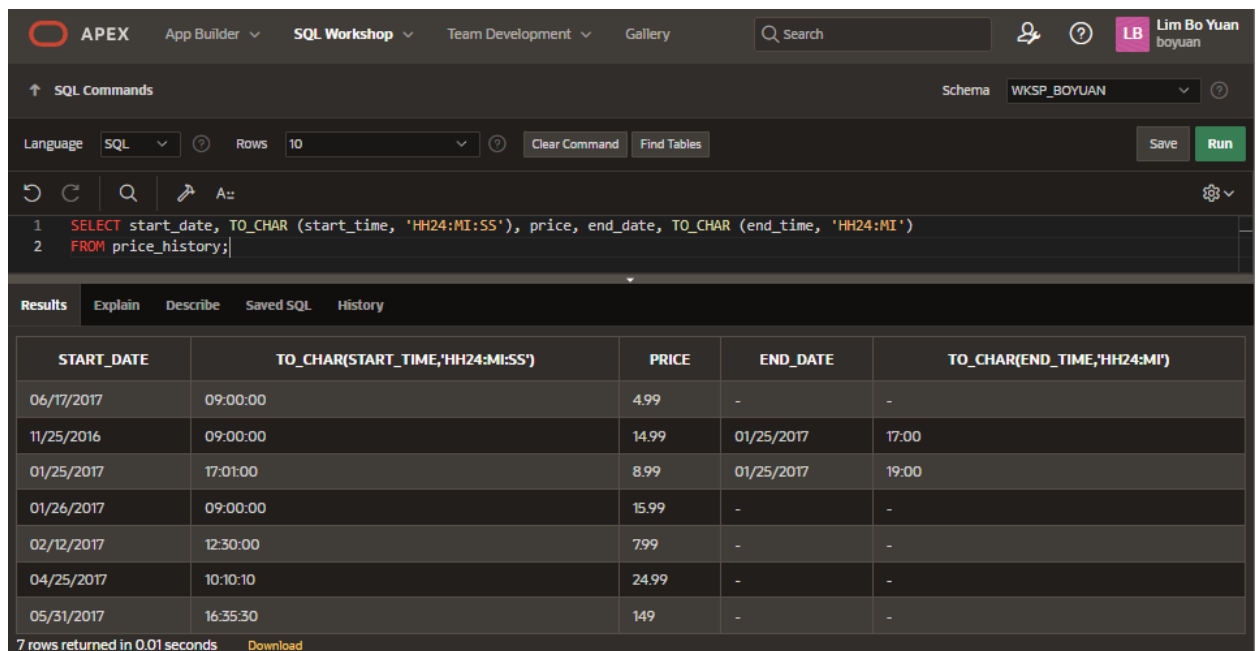
Use DML operations to manage database tables (S6L4 Objective 2)

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;
```



The screenshot shows the APEX SQL Workshop interface. The top navigation bar includes 'APEX', 'App Builder', 'SQL Workshop', 'Team Development', and 'Gallery'. A search bar and user profile 'Lim Bo Yuan' are also visible. The 'SQL Commands' section shows the executed query: `SELECT start_date, TO_CHAR (start_time, 'HH24:MI:SS'), price, end_date, TO_CHAR (end_time, 'HH24:MI') FROM price_history;`. The 'Results' tab is active, displaying a table with 7 rows of data. The table columns are: START_DATE, TO_CHAR(START_TIME, 'HH24:MI:SS'), PRICE, END_DATE, and TO_CHAR(END_TIME, 'HH24:MI'). The data rows show various dates and times, with prices ranging from 4.99 to 149. The status bar at the bottom indicates '7 rows returned in 0.01 seconds' and provides a 'Download' link.

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	-	-

- 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.

The screenshot shows the APEX SQL Workshop interface. The top navigation bar includes 'APEX', 'App Builder', 'SQL Workshop', 'Team Development', and 'Gallery'. A search bar and user profile 'Lim Bo Yuan' are on the right. The 'SQL Commands' section is active, showing a query in the editor:

```
1 UPDATE price_history
2 SET end_date = SYSDATE , end_time = CURRENT_TIMESTAMP
3 WHERE itm_number = 'im01101048' AND end_date IS NULL;
```

Below the editor, the 'Results' tab shows the execution outcome:

1 row(s) updated.
0.01 seconds

- Rerun the select statement on the price_history table to ensure that the statement has been executed.

The screenshot shows the APEX SQL Workshop interface with a SELECT query in the editor:

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

The 'Results' tab displays the query output as a table with 7 rows:

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	12/18/2023	13:40

7 rows returned in 0.01 seconds [Download](#)

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.

The screenshot shows the APEX SQL Workshop interface. The top navigation bar includes 'APEX', 'App Builder', 'SQL Workshop', 'Team Development', and 'Gallery'. A search bar and user profile 'Lim Bo Yuan boyuan' are on the right. The 'SQL Commands' section is active, with 'Schema' set to 'WKSP_BOYUAN'. The 'Language' is 'SQL' and 'Rows' is set to '10'. The SQL command entered is:

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

The 'Results' tab is selected, showing '1 row(s) inserted.' and '0.00 seconds'.

5. Rerun the select statement on the price_history table to ensure that the statement has been executed.

The screenshot shows the APEX SQL Workshop interface with the same top navigation bar. The 'SQL Commands' section is active, with 'Schema' set to 'WKSP_BOYUAN'. The 'Language' is 'SQL' and 'Rows' is set to '10'. The SQL command entered is:

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

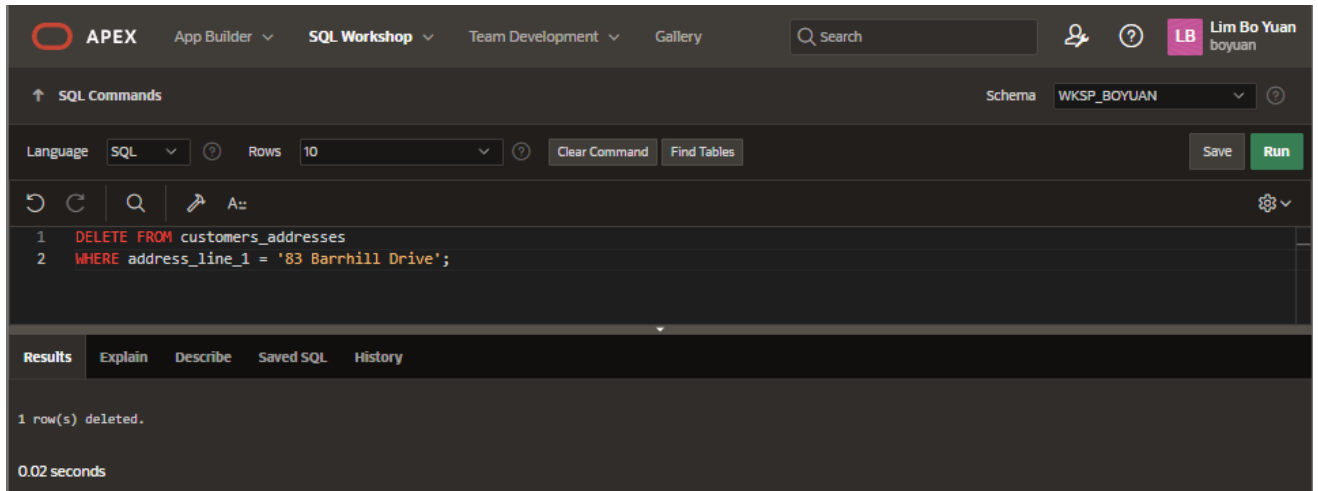
The 'Results' tab is selected, displaying a table with 8 rows. The table has the following columns: START_DATE, TO_CHAR(START_TIME,'HH24:MI:SS'), PRICE, END_DATE, and TO_CHAR(END_TIME,'HH24:MI').

START_DATE	TO_CHAR(START_TIME,'HH24:MI:SS')	PRICE	END_DATE	TO_CHAR(END_TIME,'HH24:MI')
12/18/2023	13:45:44	99.99	-	-
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	14.99	12/18/2023	13:40

8 rows returned in 0.01 seconds [Download](#)

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 longer receive parcels at this address. Write a SQL statement that will remove this address from the system.

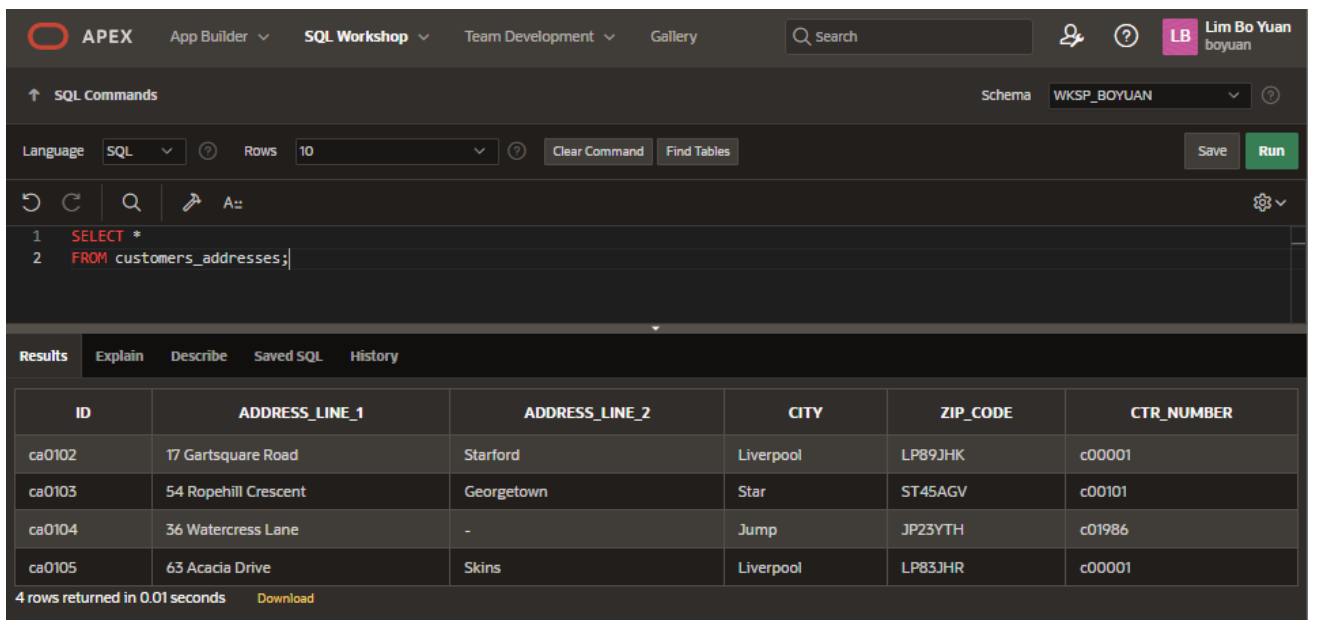


The screenshot shows the APEX SQL Workshop interface. The top navigation bar includes 'APEX', 'App Builder', 'SQL Workshop', 'Team Development', and 'Gallery'. A search bar and user profile 'Lim Bo Yuan boyuan' are on the right. The 'SQL Commands' section shows the schema 'WKSP_BOYUAN'. The 'Language' is set to 'SQL' and 'Rows' to '10'. The SQL command entered is:

```
1 DELETE FROM customers_addresses
2 WHERE address_line_1 = '83 Barrhill Drive';
```

The 'Results' tab is active, displaying the message '1 row(s) deleted.' and the execution time '0.02 seconds'.

2. Run a select statement on the customers_addresses table to ensure that the statement has been executed.



The screenshot shows the APEX SQL Workshop interface. The top navigation bar is the same as the previous screenshot. The 'SQL Commands' section shows the schema 'WKSP_BOYUAN'. The 'Language' is set to 'SQL' and 'Rows' to '10'. The SQL command entered is:

```
1 SELECT *
2 FROM customers_addresses;
```

The 'Results' tab is active, displaying a table with 4 rows. The table has the following columns: ID, ADDRESS_LINE_1, ADDRESS_LINE_2, CITY, ZIP_CODE, and CTR_NUMBER.

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	ST4SAGV	c00101
ca0104	36 Watercress Lane	-	Jump	JP23YTH	c01986
ca0105	63 Acacia Drive	Skins	Liverpool	LP83JHR	c00001

4 rows returned in 0.01 seconds [Download](#)