



UTM
UNIVERSITI TEKNOLOGI MALAYSIA

Faculty of
Computer Science
and Information
Systems

UNIVERSITI TEKNOLOGI MALAYSIA, JOHOR BAHRU

FACULTY OF COMPUTING

SECD2523-08 DATABASE

Lab 2: Data Manipulation Language (DML) 1

<i>Name</i>	<i>Matric Number</i>
<i>Ahmed Zaki Al-Gabaly</i>	<i>A22EC4003</i>

Lecturer: Dr. Noor Hidayah Binti Zakaria

Section 6 Lesson 4 Exercise 1: Data Manipulation Language

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

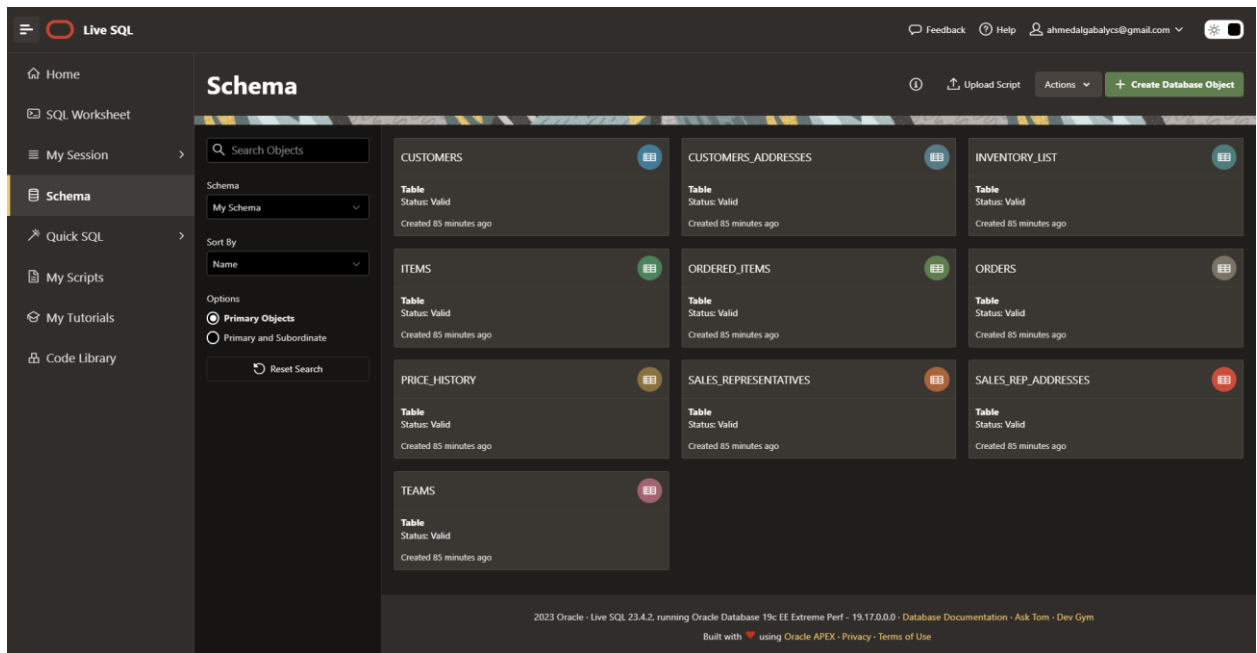
In this exercise you will populate and work with the data that is stored in the database system tables.

Part 1 : Running a script to populate the tables.

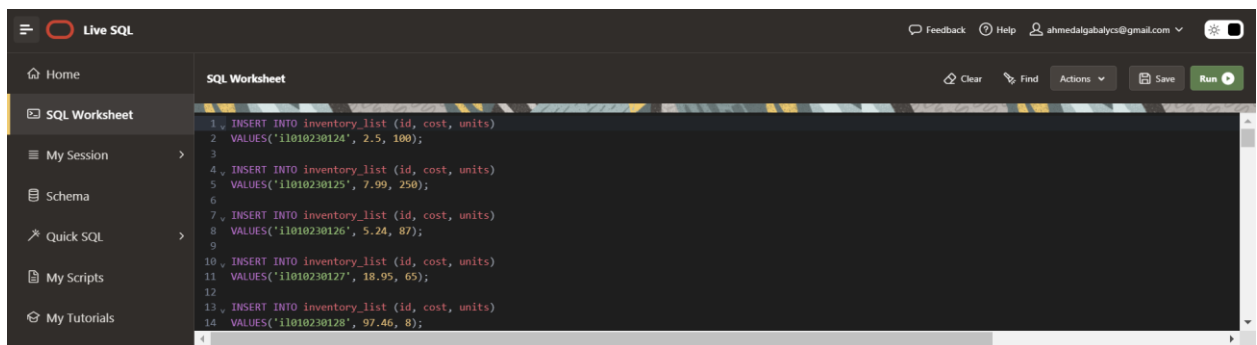
You have to consider the order of the tables when populating them. A table that has a foreign key field cannot be populated before the related table with the primary key.

1. Use the table mapping document and list the order that you would use to populate the tables.

- 1) INVENTORY_LIST
- 2) ITEMS
- 3) PRICE_HISTORY
- 4) SALES_REPRESENTATIVES
- 5) SALES_REP_ADDRESSES
- 6) CUSTOMERS
- 7) TEAMS
- 8) CUSTOMERS_ADDRESSES
- 9) ORDERS
- 10) ORDERED_ITEMS

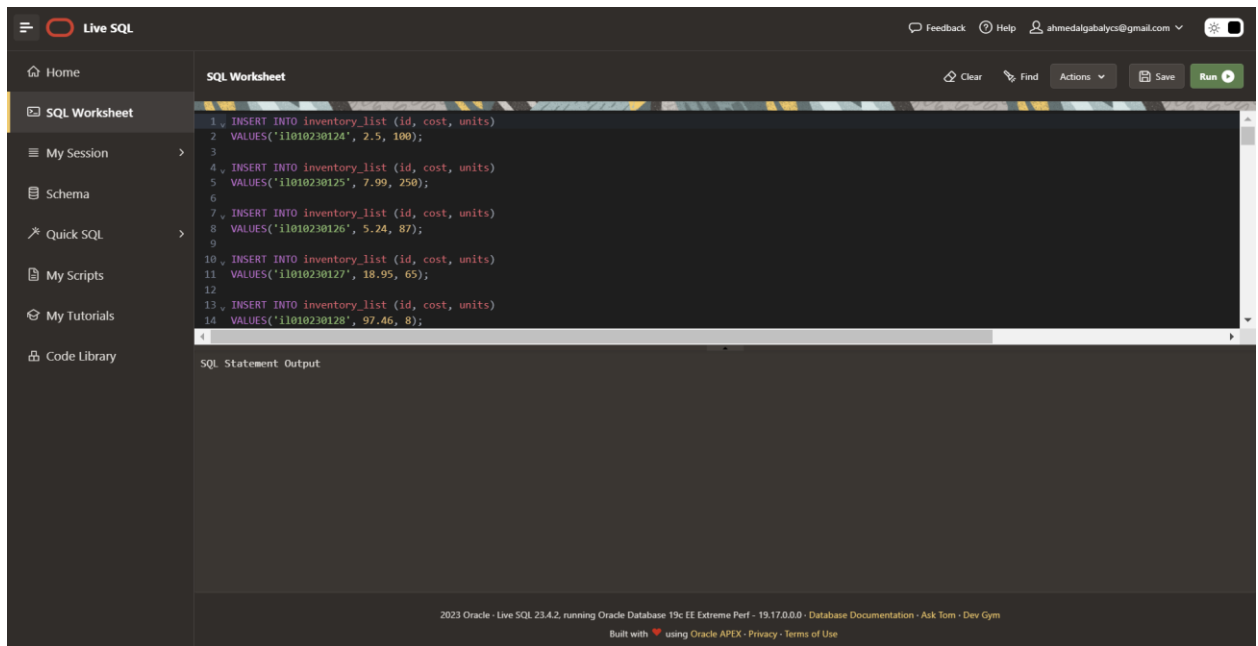


2. Open the “sports data.sql” and look at the order the data is being added there, does your list match? This file can be found in the Section 6 Lesson 4 interaction (sports data.zip) and must first be extracted.



Yes, it matched.

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




















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









Statement 36	<div><div>Insert</div><div>Edit</div></div> <pre>INSERT INTO inventory_list (id, cost, units) VALUES('11010230124', 2.5, 100)</pre> <div>1 row(s) inserted.</div>
Statement 37	<div><div>Insert</div><div>Edit</div></div> <pre>INSERT INTO inventory_list (id, cost, units) VALUES('11010230125', 7.99, 250)</pre> <div>1 row(s) inserted.</div>
Statement 38	<div><div>Insert</div><div>Edit</div></div> <pre>INSERT INTO inventory_list (id, cost, units) VALUES('11010230126', 5.24, 87)</pre> <div>1 row(s) inserted.</div>
Statement 39	<div><div>Insert</div><div>Edit</div></div> <pre>INSERT INTO inventory_list (id, cost, units) VALUES('11010230127', 18.95, 65)</pre> <div>1 row(s) inserted.</div>

Statement 40	<div><div>🗑️</div><div>Edit</div></div> <pre>INSERT INTO inventory_list (id, cost, units) VALUES('i1010230128', 97.46, 8)</pre> <div>1 row(s) inserted.</div>
Statement 41	<div><div>🗑️</div><div>Edit</div></div> <pre>INSERT INTO items (itm_number, name, description, category, color, "Size", ilt_id) VALUES('im01101044', 'gloves', 'catcher mitt', 'clothing', 'brown', 'm', 'i1010230124')</pre> <div>1 row(s) inserted.</div>
Statement 42	<div><div>🗑️</div><div>Edit</div></div> <pre>INSERT INTO items (itm_number, name, description, category, color, "Size", ilt_id) VALUES('im01101045', 'under shirt', 'top worn under the game top', 'clothing', 'white', 's', 'i1010230125')</pre> <div>1 row(s) inserted.</div>
Statement 43	<div><div>🗑️</div><div>Edit</div></div> <pre>INSERT INTO items (itm_number, name, description, category, color, "Size", ilt_id) VALUES('im01101046', 'socks', 'team socks with emblem', 'clothing', 'range', 'l', 'i1010230126')</pre> <div>1 row(s) inserted.</div>
Statement 44	<div><div>🗑️</div><div>Edit</div></div> <pre>INSERT INTO items (itm_number, name, description, category, color, "Size", ilt_id) VALUES('im01101047', 'game top', 'team shirt with emblem', 'clothing', 'range', 'm', 'i1010230127')</pre> <div>1 row(s) inserted.</div>

Statement 45	<div><div>🗑️</div><div>Edit</div></div> <pre>INSERT INTO items (itm_number, name, description, category, ilt_id) VALUES('im01101048', 'premium bat', 'high quality baseball bat', 'equipment', 'i1010230128')</pre> <div>1 row(s) inserted.</div>
Statement 46	<div><div>🗑️</div><div>Edit</div></div> <pre>INSERT INTO price_history (start_date, start_time, price, itm_number) 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')</pre> <div>1 row(s) inserted.</div>
Statement 47	<div><div>🗑️</div><div>Edit</div></div> <pre>INSERT INTO price_history (start_date, start_time, price, end_date, end_time, itm_number) 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-Jan-2017', 'DD-MM-YYYY'), TO_DATE('25-Jan-2017 17:00:00'))</pre> <div>1 row(s) inserted.</div>
Statement 48	<div><div>🗑️</div><div>Edit</div></div> <pre>INSERT INTO price_history (start_date, start_time, price, end_date, end_time, itm_number) 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-Jan-2017', 'DD-MM-YYYY'), TO_DATE('25-Jan-2017 19:00:00'))</pre> <div>1 row(s) inserted.</div>
Statement 49	<div><div>🗑️</div><div>Edit</div></div> <pre>INSERT INTO price_history (start_date, start_time, price, itm_number) 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')</pre> <div>1 row(s) inserted.</div>

Statement 50	 	<pre>INSERT INTO price_history (start_date, start_time, price, itm_number) 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')</pre>
		1 row(s) inserted.
Statement 51	 	<pre>INSERT INTO price_history (start_date, start_time, price, itm_number) 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')</pre>
		1 row(s) inserted.
Statement 52	 	<pre>INSERT INTO price_history (start_date, start_time, price, itm_number) VALUES(TO_DATE('31-May-2017', 'DD-MM-YYYY'), TO_DATE('31-May-2017 16:35:30', 'DD-MM-YYYY hh24:mi:ss'), 149.00, 'im01101048')</pre>
		1 row(s) inserted.
Statement 53	 	<pre>INSERT INTO sales_representatives (id, email, first_name, last_name, phone_number, commission_rate, supervisor_id) VALUES('sr01', 'chray@obl.com', 'Charles', 'Raymond', '0134598761', 10, 'sr01')</pre>
		1 row(s) inserted.
Statement 54	 	<pre>INSERT INTO sales_representatives (id, email, first_name, last_name, phone_number, commission_rate, supervisor_id) VALUES('sr02', 'vwrigh@obl.com', 'Victoria', 'Wright', '0134598762', 5, 'sr01')</pre>
		1 row(s) inserted.

Statement 55	 	<pre>INSERT INTO sales_representatives (id, email, first_name, last_name, phone_number, commission_rate, supervisor_id) VALUES('sr03', 'bspeed@obl.com', 'Barry', 'Speed', '0134598763', 5, 'sr01')</pre>
		1 row(s) inserted.
Statement 56	 	<pre>INSERT INTO sales_rep_addresses (id, address_line_1, address_line_2, city, zip_code) VALUES('sr01', '12 Cherry Lane', 'Denton', 'Detroit', 'DT48211')</pre>
		1 row(s) inserted.
Statement 57	 	<pre>INSERT INTO sales_rep_addresses (id, address_line_1, address_line_2, city, zip_code) VALUES('sr02', '87 Blossom Hill', 'Uptown', 'Detroit', 'DT52314')</pre>
		1 row(s) inserted.
Statement 58	 	<pre>INSERT INTO sales_rep_addresses (id, address_line_1, address_line_2, city, zip_code) VALUES('sr03', '12 Junction Row', 'Skinflats', 'Detroit', 'DT52564')</pre>
		1 row(s) inserted.
Statement 59	 	<pre>INSERT INTO teams (id, name, number_of_players, discount) VALUES('t001', 'Rockets', 25, 10)</pre>
		1 row(s) inserted.

Statement 60	 	<pre>INSERT INTO teams (id, name, number_of_players, discount) VALUES('t002', 'Celtics', 42, 20)</pre>
		1 row(s) inserted.
Statement 61	 	<pre>INSERT INTO teams (id, name, number_of_players, discount) VALUES('t003', 'Rovers', 8, null)</pre>
		1 row(s) inserted.
Statement 62	 	<pre>INSERT INTO customers (ctr_number, email, first_name, last_name, phone_number, current_balance, sre_id, tem_id, loyalty_card_number) VALUES('c00001', 'bob.thornberry@heatmail.com', 'Robert', 'Thornberry', '01234567898', 150.00, 'sr01', 't001', null)</pre>
		1 row(s) inserted.
Statement 63	 	<pre>INSERT INTO customers (ctr_number, email, first_name, last_name, phone_number, current_balance, loyalty_card_number) VALUES('c00012', 'Jjones@freemail.com', 'Jennifer', 'Jones', '01505214598', 0.00, 'lc1015')</pre>
		1 row(s) inserted.
Statement 64	 	<pre>INSERT INTO customers (ctr_number, email, first_name, last_name, phone_number, current_balance, sre_id, tem_id) VALUES('c00101', 'unknown@here.com', 'John', 'Doe', '03216547800', 987.50, 'sr01', 't002')</pre>
		1 row(s) inserted.

Statement 65	<div> <div></div> <div>Edit</div> </div> <pre>INSERT INTO customers (ctr_number, email, first_name, last_name, phone_number, current_balance, loyalty_card_number) VALUES('c00103', 'MurciaA@globaltech.com', 'Andrew', 'Murcia', '07715246890', 85.00, 'lc2341')</pre> <div>1 row(s) inserted.</div>
Statement 66	<div> <div></div> <div>Edit</div> </div> <pre>INSERT INTO customers (ctr_number, email, first_name, last_name, phone_number, current_balance, sre_id, tem_id) VALUES('c01986', 'margal87@delphiview.com', 'Maria', 'Galant', '01442736589', 125.65 , 'sr03', 't003')</pre> <div>1 row(s) inserted.</div>
Statement 67	<div> <div></div> <div>Edit</div> </div> <pre>INSERT INTO customers_addresses (id, address_line_1, address_line_2, city, zip_code, ctr_number) VALUES('ca0101', '83 Barrhill Drive', null, 'Liverpool', 'LP79JHK', 'c00001')</pre> <div>1 row(s) inserted.</div>
Statement 68	<div> <div></div> <div>Edit</div> </div> <pre>INSERT INTO customers_addresses (id, address_line_1, address_line_2, city, zip_code, ctr_number) VALUES('ca0102', '17 Gartsquare Road', 'Starford', 'Liverpool', 'LP89JHK', 'c00001')</pre> <div>1 row(s) inserted.</div>
Statement 69	<div> <div></div> <div>Edit</div> </div> <pre>INSERT INTO customers_addresses (id, address_line_1, address_line_2, city, zip_code, ctr_number) VALUES('ca0103', '54 Ropehill Crescent', 'Georgetown', 'Star', 'ST45AGV', 'c00101')</pre> <div>1 row(s) inserted.</div>

Statement 70	<div> <div></div> <div>Edit</div> </div> <pre>INSERT INTO customers_addresses (id, address_line_1, address_line_2, city, zip_code, ctr_number) VALUES('ca0104', '36 Watercross Lane', null, 'Jump', 'JP23YTH', 'c01986')</pre> <div>1 row(s) inserted.</div>
Statement 71	<div> <div></div> <div>Edit</div> </div> <pre>INSERT INTO customers_addresses (id, address_line_1, address_line_2, city, zip_code, ctr_number) VALUES('ca0105', '63 Acacia Drive', 'Skins', 'Liverpool', 'LP83JHR', 'c00001')</pre> <div>1 row(s) inserted.</div>
Statement 72	<div> <div></div> <div>Edit</div> </div> <pre>INSERT INTO orders (id, odr_date, odr_time, number_of_units, ctr_number) VALUES('or0101250', TO_DATE('17-Apr-2017', 'DD-MM-YYYY'), TO_DATE('17-Apr-2017 08:32:30', 'DD-MM-YYYY hh24:mi:ss'), 10, 'c00001')</pre> <div>1 row(s) inserted.</div>
Statement 73	<div> <div></div> <div>Edit</div> </div> <pre>INSERT INTO orders (id, odr_date, odr_time, number_of_units, ctr_number) VALUES('or0101350', TO_DATE('24-May-2017', 'DD-MM-YYYY'), TO_DATE('24-May-2017 10:30:35 ', 'DD-MM-YYYY hh24:mi:ss'), 5, 'c00001')</pre> <div>1 row(s) inserted.</div>

Statement 74	<div> <div></div> <div>Edit</div> </div> <pre>INSERT INTO orders (id, odr_date, odr_time, number_of_units, ctr_number) VALUES('or0101425', TO_DATE('28-May-2017', 'DD-MM-YYYY'), TO_DATE('28-May-2017 12:30:00 ', 'DD-MM-YYYY hh24:mi:ss'), 18, 'c00103')</pre> <div>1 row(s) inserted.</div>
Statement 75	<div> <div></div> <div>Edit</div> </div> <pre>INSERT INTO orders (id, odr_date, odr_time, number_of_units, ctr_number) VALUES('or0101601', TO_DATE('02-Jun-2017', 'DD-MM-YYYY'), TO_DATE('02-Jun-2017 14:55:30 ', 'DD-MM-YYYY hh24:mi:ss'), 10, 'c00001')</pre> <div>1 row(s) inserted.</div>
Statement 76	<div> <div></div> <div>Edit</div> </div> <pre>INSERT INTO orders (id, odr_date, odr_time, number_of_units, ctr_number) VALUES('or0101750', TO_DATE('18-Jun-2017', 'DD-MM-YYYY'), TO_DATE('18-Jun-2017 09:05:00 ', 'DD-MM-YYYY hh24:mi:ss'), 1, 'c01986')</pre> <div>1 row(s) inserted.</div>
Statement 77	<div> <div></div> <div>Edit</div> </div> <pre>INSERT INTO ordered_items (quantity_ordered, quantity_shipped, odr_id, itm_number) VALUES(5, 5, 'or0101250', 'im01101044')</pre> <div>1 row(s) inserted.</div>
Statement 78	<div> <div></div> <div>Edit</div> </div> <pre>INSERT INTO ordered_items (quantity_ordered, quantity_shipped, odr_id, itm_number) VALUES(5, 5, 'or0101250', 'im01101046')</pre> <div>1 row(s) inserted.</div>

Statement 79	<div> <div> <div></div> <div>Edit</div> </div> <pre>INSERT INTO ordered_items (quantity_ordered, quantity_shipped, odr_id, itm_number) VALUES(5, 5, 'or0101350', 'im01101044')</pre> </div>
	1 row(s) inserted.
Statement 80	<div> <div> <div></div> <div>Edit</div> </div> <pre>INSERT INTO ordered_items (quantity_ordered, quantity_shipped, odr_id, itm_number) VALUES(18, 18, 'or0101425', 'im01101047')</pre> </div>
	1 row(s) inserted.
Statement 81	<div> <div> <div></div> <div>Edit</div> </div> <pre>INSERT INTO ordered_items (quantity_ordered, quantity_shipped, odr_id, itm_number) VALUES(10, 10, 'or0101681', 'im01101047')</pre> </div>
	1 row(s) inserted.
Statement 82	<div> <div> <div></div> <div>Edit</div> </div> <pre>INSERT INTO ordered_items (quantity_ordered, quantity_shipped, odr_id, itm_number) VALUES(1, 1, 'or0101750', 'im01101048')</pre> </div>
	1 row(s) inserted.

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

≡

Live SQL

Feedback

Help

ahmedalgaabalyca@gmail.com

⚙️

Home

SQL Worksheet

My Session

Schema

Quick SQL

My Scripts

My Tutorials

Code Library

SQL Worksheet

Clear

Find

Actions

Save

Run

```
1, INSERT INTO TEAMS (id, name, number_of_players, discount)
2, VALUES ('t004', 'Jets', 10, 5)
```

1 row(s) inserted.

2023 Oracle - Live SQL 23.4.2, running Oracle Database 19c EE Extreme Perf - 19.17.0.0.0 - Database Documentation - Ask Tom - Dev Gym

Built with using Oracle APEX - Privacy - Terms of Use

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	lc4587		

Live SQL interface showing an SQL error. The SQL Worksheet contains the following SQL:

```
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, 'lc4587');
```

The output shows an error: ORA-02290: check constraint (SQL_FGULPULSHIFRBYTHYMOQXFDV.CUST_BAL_CHECK) violated ORA-06512: at 'SYS.DBMS_SQL', line 1721. More Details: <https://docs.oracle.com/error-help/db/ora-02290>

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.

Live SQL interface showing the successful execution of the SQL query. The SQL Worksheet contains the following SQL:

```
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, 'lc4587');
```

The output shows: 1 row(s) inserted.

Section 6 Lesson 4 Exercise 2: Data Manipulation Language

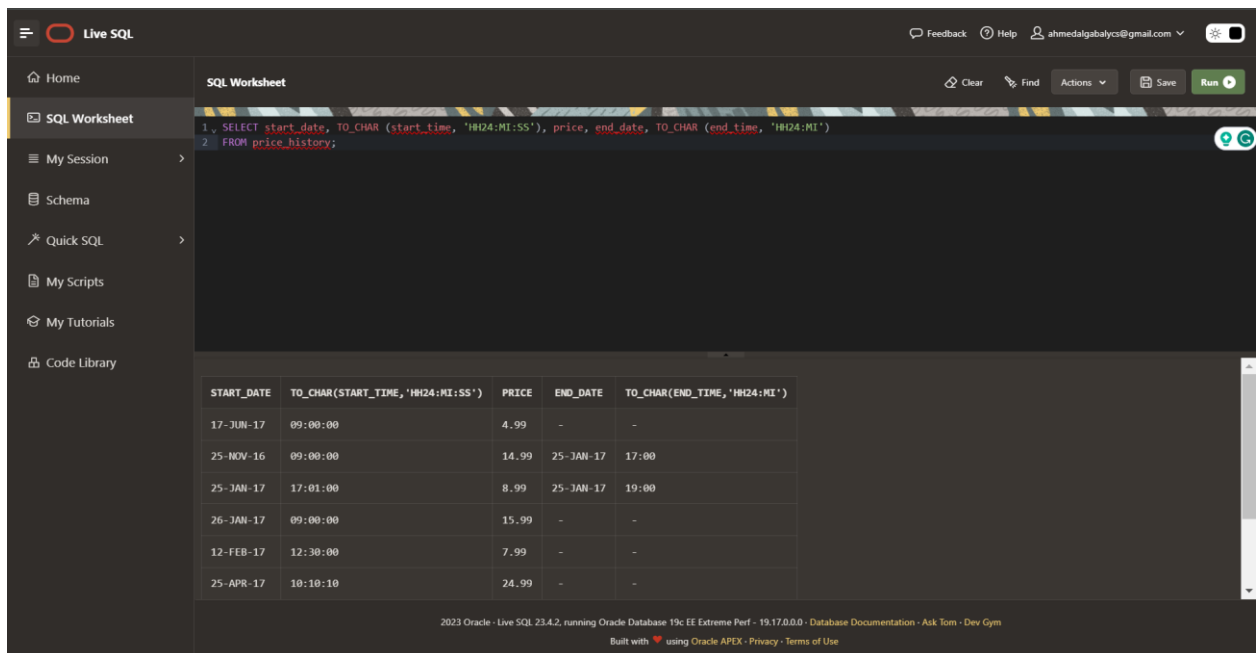
Use DML operations to manage database tables (S6L4 Objective 2)

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 Live SQL interface. The SQL Worksheet contains the following query:

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

The results of the query are displayed in a table with 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')
17-JUN-17	09:00:00	4.99	-	-
25-NOV-16	09:00:00	14.99	25-JAN-17	17:00
25-JAN-17	17:01:00	8.99	25-JAN-17	19:00
26-JAN-17	09:00:00	15.99	-	-
12-FEB-17	12:30:00	7.99	-	-
25-APR-17	10:10:10	24.99	-	-

2023 Oracle - Live SQL 23.4.2, running Oracle Database 19c EE Extreme Perf - 19.17.0.0.0 - Database Documentation - Ask Tom - Dev Gym
Built with ❤️ using Oracle APEX - Privacy - Terms of Use

The screenshot shows the 'Live SQL' web application. On the left is a sidebar with navigation links: Home, SQL Worksheet, My Session, Schema, Quick SQL, My Scripts, My Tutorials, and Code Library. The main area is titled 'SQL Worksheet' and displays a table with the following data:

START_DATE	TO_CHAR(START_TIME, 'HH24:MI:SS')	PRICE	END_DATE	TO_CHAR(END_TIME, 'HH24:MI')
17-JUN-17	09:00:00	4.99	-	-
25-NOV-16	09:00:00	14.99	25-JAN-17	17:00
25-JAN-17	17:01:00	8.99	25-JAN-17	19:00
26-JAN-17	09:00:00	15.99	-	-
12-FEB-17	12:30:00	7.99	-	-
25-APR-17	10:10:10	24.99	-	-
31-MAY-17	16:35:30	149	-	-

Below the table, there is a 'Download CSV' button and a message '7 rows selected.' At the bottom of the interface, a footer reads: '2023 Oracle - Live SQL 23.4.2, running Oracle Database 19c EE Extreme Perf - 19.17.0.0.0 - Database Documentation - Ask Tom - Dev Gym. Built with using Oracle APEX - Privacy - Terms of Use'.

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.

The screenshot shows the 'Live SQL' web application with the same sidebar as the first image. The main area is titled 'SQL Worksheet' and contains the following SQL query:

```
1. UPDATE price_history
2. SET end_date = SYSDATE, end_time = SYSDATE
3. WHERE itm_number = 'im01101045' AND end_date is null
```

Below the query, there is a message '1 row(s) updated.' At the bottom of the interface, the same footer as the first image is visible: '2023 Oracle - Live SQL 23.4.2, running Oracle Database 19c EE Extreme Perf - 19.17.0.0.0 - Database Documentation - Ask Tom - Dev Gym. Built with using Oracle APEX - Privacy - Terms of Use'.

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

Live SQL

Feedback Help ahmedalgaabaly@gmail.com

Home

SQL Worksheet

My Session

Schema

Quick SQL

My Scripts

My Tutorials

Code Library

SQL Worksheet

1. SELECT start_date, TO_CHAR (start_time, 'HH24:MI:SS'), price, end_date, TO_CHAR (end_time, 'HH24:MI:')

2. FROM price_history;

START_DATE	TO_CHAR(START_TIME, 'HH24:MI:SS')	PRICE	END_DATE	TO_CHAR(END_TIME, 'HH24:MI:')
17-JUN-17	09:00:00	4.99	-	-
25-NOV-16	09:00:00	14.99	25-JAN-17	17:00
25-JAN-17	17:01:00	8.99	25-JAN-17	19:00
26-JAN-17	09:00:00	15.99	16-DEC-23	13:55
12-FEB-17	12:30:00	7.99	-	-
25-APR-17	10:10:10	24.99	-	-

2023 Oracle - Live SQL 23.4.2, running Oracle Database 19c EE Extreme Perf - 19.17.0.0.0 - Database Documentation - Ask Tom - Dev Gym

Built with using Oracle APEX - Privacy - Terms of Use

Live SQL

Feedback Help ahmedalgaabaly@gmail.com

Home

SQL Worksheet

My Session

Schema

Quick SQL

My Scripts

My Tutorials

Code Library

SQL Worksheet

START_DATE	TO_CHAR(START_TIME, 'HH24:MI:SS')	PRICE	END_DATE	TO_CHAR(END_TIME, 'HH24:MI:')
17-JUN-17	09:00:00	4.99	-	-
25-NOV-16	09:00:00	14.99	25-JAN-17	17:00
25-JAN-17	17:01:00	8.99	25-JAN-17	19:00
26-JAN-17	09:00:00	15.99	16-DEC-23	13:55
12-FEB-17	12:30:00	7.99	-	-
25-APR-17	10:10:10	24.99	-	-
31-MAY-17	16:35:30	149	-	-

Download CSV

7 rows selected.

2023 Oracle - Live SQL 23.4.2, running Oracle Database 19c EE Extreme Perf - 19.17.0.0.0 - Database Documentation - Ask Tom - Dev Gym

Built with using Oracle APEX - Privacy - Terms of Use

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 Live SQL interface. On the left is a sidebar with navigation options: Home, SQL Worksheet (selected), My Session, Schema, Quick SQL, My Scripts, My Tutorials, and Code Library. The main area is titled 'SQL Worksheet' and contains the following SQL code:

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

Below the code, the execution result is displayed: '1 row(s) inserted.' At the bottom of the interface, there is a footer with the text: '2023 Oracle - Live SQL 23.4.2, running Oracle Database 19c EE Extreme Perf - 19.17.0.0.0 - Database Documentation - Ask Tom - Dev Gym' and 'Built with using Oracle APEX - Privacy - Terms of Use'.

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

The screenshot shows the Live SQL interface with the same sidebar. The main area is titled 'SQL Worksheet' and contains the following SQL code:

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

Below the code, the execution result is displayed as a table:

START_DATE	TO_CHAR (START_TIME, 'HH24:MI:SS')	PRICE	END_DATE	TO_CHAR (END_TIME, 'HH24:MI')
17-JUN-17	09:00:00	4.99	-	-
25-NOV-16	09:00:00	14.99	25-JAN-17	17:00
25-JAN-17	17:01:00	8.99	25-JAN-17	19:00
26-JAN-17	09:00:00	15.99	16-DEC-23	13:55
12-FEB-17	12:30:00	7.99	-	-
25-APR-17	10:10:10	24.99	-	-

At the bottom of the interface, there is a footer with the text: '2023 Oracle - Live SQL 23.4.2, running Oracle Database 19c EE Extreme Perf - 19.17.0.0.0 - Database Documentation - Ask Tom - Dev Gym' and 'Built with using Oracle APEX - Privacy - Terms of Use'.

START_DATE	TO_CHAR(START_TIME, 'HH24:MI:SS')	PRICE	END_DATE	TO_CHAR(END_TIME, 'HH24:MI:SS')
17-JUN-17	09:00:00	4.99	-	-
25-NOV-16	09:00:00	14.99	25-JAN-17	17:00
25-JAN-17	17:01:00	8.99	25-JAN-17	19:00
26-JAN-17	09:00:00	15.99	16-DEC-23	13:55
12-FEB-17	12:30:00	7.99	-	-
25-APR-17	10:10:10	24.99	-	-
31-MAY-17	16:35:30	149	-	-
16-DEC-23	14:00:34	99.99	-	-

Download CSV
8 rows selected.

2023 Oracle - Live SQL 23.4.2, running Oracle Database 19c EE Extreme Perf - 19.17.0.0.0 - Database Documentation - Ask Tom - Dev Gym
Built with using Oracle APEX - Privacy - Terms of Use

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.

```

1. DELETE FROM CUSTOMERS_ADDRESSES
2. WHERE Id = 'ca0101'

```

1 row(s) deleted.

2023 Oracle - Live SQL 23.4.2, running Oracle Database 19c EE Extreme Perf - 19.17.0.0.0 - Database Documentation - Ask Tom - Dev Gym
Built with using Oracle APEX - Privacy - Terms of Use

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

Live SQL

FeedbackHelpahmedalgalalycs@gmail.com

Home

SQL Worksheet

My Session

Schema

Quick SQL

My Scripts

My Tutorials

Code Library

SQL Worksheet

1 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

Download CSV

4 rows selected.

2023 Oracle - Live SQL 23.4.2, running Oracle Database 19c EE Extreme Perf - 19.17.0.0.0 - Database Documentation - Ask Tom - Dev Gym
Built with using Oracle APEX - Privacy - Terms of Use