UTM

UNIVERSITI TEKNOLOGI MALAYSIA

UNIVERSITI TEKNOLOGI MALAYSIA
FACULTY OF COMPUTING
SEMESTER I, SESION 2023/2024

LAB EXERCISE 2 : DML

SECD2523 – DATABASE

NAME	:	**CHONG WEI YANG**
MARIC NO	:	**A23CS5027**
COURSE	:	**2 SECJH**
SUBMITION DATE	:	**17 DISEMBER 2023**
LECTURER'S NAME	:	**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.

```
CREATE TABLE inventory_list (
    id          VARCHAR2(11) NOT NULL,
    cost        NUMBER(7,2) NOT NULL,
    units       NUMBER(4) NOT NULL,
    CONSTRAINT inventory_list_pk PRIMARY KEY ( id )
)
```

```
CREATE TABLE items (
    itm_number      VARCHAR2(10) NOT NULL,
    name            VARCHAR2(20) NOT NULL,
    description     VARCHAR2(50) NOT NULL,
    category        VARCHAR2(25) NOT NULL,
    color           VARCHAR2(15),
    "Size"          CHAR(1),
    ilt_id          VARCHAR2(11) NOT NULL,
    CONSTRAINT item_pk PRIMARY KEY ( itm_number )
);
```

```
CREATE TABLE price_history (
    start_date      DATE NOT NULL,
    start_time      DATE NOT NULL,
    price           NUMBER(7,2) NOT NULL,
    end_date        DATE,
    end_time        DATE,
    itm_number      VARCHAR2(10) NOT NULL,
    CONSTRAINT price_history_pk PRIMARY KEY ( itm_number, start_date, start_time ),
    CONSTRAINT price_history_items_fk FOREIGN KEY ( itm_number ) REFERENCES items ( itm_number )

);
```

```sql
CREATE TABLE sales_representatives (
    id              VARCHAR2(4) NOT NULL,
    email           VARCHAR2(50) NOT NULL,
    first_name      VARCHAR2(20) NOT NULL,
    last_name       VARCHAR2(30) NOT NULL,
    phone_number    VARCHAR2(11) NOT NULL,
    commission_rate  NUMBER(2) NOT NULL,
    supervisor_id   VARCHAR2(4) NOT NULL,
    CONSTRAINT sales_representative_pk PRIMARY KEY ( id ),
    CONSTRAINT sre_email_uk UNIQUE (email)
);
```

```sql
CREATE TABLE teams (
    id                  VARCHAR2(4) NOT NULL,
    name                VARCHAR2(20) NOT NULL,
    number_of_players   NUMBER(2) NOT NULL,
    discount            NUMBER(2),
    CONSTRAINT team_pk PRIMARY KEY ( id )
);
```

```sql
CREATE TABLE customers (
    ctr_number          VARCHAR2(6) NOT NULL,
    email               VARCHAR2(50) NOT NULL,
    first_name          VARCHAR2(20) NOT NULL,
    last_name           VARCHAR2(30) NOT NULL,
    phone_number        VARCHAR2(11) NOT NULL,
    current_balance     NUMBER(6,2) NOT NULL,
    sre_id              VARCHAR2(4),
    tem_id              VARCHAR2(4),
    loyalty_card_number VARCHAR2(6),
    CONSTRAINT customer_pk PRIMARY KEY ( ctr_number ),
    CONSTRAINT ctr_email_uk UNIQUE (email),
    CONSTRAINT ctr_lcn_uk UNIQUE (loyalty_card_number)
);
```

```sql
CREATE TABLE customers_addresses (
    id               VARCHAR2(8) NOT NULL,
    address_line_1   VARCHAR2(30) NOT NULL,
    address_line_2   VARCHAR2(30),
    city             VARCHAR2(15) NOT NULL,
    zip_code         VARCHAR2(7) NOT NULL,
    ctr_number       VARCHAR2(6) NOT NULL,
    CONSTRAINT customer_address_pk PRIMARY KEY ( id )
);
```

```sql
CREATE TABLE orders (
    id                VARCHAR2(9) NOT NULL,
    odr_date          DATE NOT NULL,
    odr_time          DATE NOT NULL,
    number_of_units   NUMBER(2) NOT NULL,
    ctr_number        VARCHAR2(6) NOT NULL,
    CONSTRAINT orders_pk PRIMARY KEY ( id )
);
```

```sql
CREATE TABLE ordered_items (
    quantity_ordered   NUMBER(3) NOT NULL,
    quantity_shipped   NUMBER(3) NOT NULL,
    itm_number         VARCHAR2(10) NOT NULL,
    odr_id             VARCHAR2(9) NOT NULL,
    CONSTRAINT ordered_item_pk PRIMARY KEY ( itm_number,odr_id )
);
```

```sql
ALTER TABLE customers_addresses ADD CONSTRAINT customer_address_customer_fk FOREIGN KEY ( ctr_number )
    REFERENCES customers ( ctr_number );
```

```sql
ALTER TABLE customers ADD CONSTRAINT customer_sales_rep_fk FOREIGN KEY ( sre_id )
    REFERENCES sales_representatives ( id );
```

```sql
ALTER TABLE customers ADD CONSTRAINT customer_team_fk FOREIGN KEY ( tem_id )
    REFERENCES teams ( id );
```

```sql
ALTER TABLE items ADD CONSTRAINT item_inventory_list_fk FOREIGN KEY ( ilt_id )
    REFERENCES inventory_list ( id );
```

```
ALTER TABLE orders ADD CONSTRAINT order_customer_fk FOREIGN KEY ( ctr_number )
    REFERENCES customers ( ctr_number );
```

```
ALTER TABLE ordered_items ADD CONSTRAINT ordered_item_item_fk FOREIGN KEY ( itm_number )
    REFERENCES items ( itm_number );
```

```
ALTER TABLE ordered_items ADD CONSTRAINT ordered_item_order_fk FOREIGN KEY ( odr_id )
    REFERENCES orders ( id );
```

```
ALTER TABLE sales_rep_addresses ADD CONSTRAINT sales_rep_add_sales_rep_fk FOREIGN KEY ( id )
    REFERENCES sales_representatives ( id );
```

```
ALTER TABLE sales_representatives ADD CONSTRAINT sales_rep_sales_rep_fk FOREIGN KEY ( supervisor_id ) REFERENCES sales_representatives ( id );
```

```
CREATE OR REPLACE TRIGGER fkntm_orders BEFORE
    UPDATE OF ctr_number ON orders
BEGIN
    raise_application_error(
        -20225,
        'Non Transferable FK constraint  on table orders is violated'
    );
END;
```

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

Yes

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

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

SQL Scripts \ Results

Script: **sports data** Status: **Complete**

View: Detail **Summary** Rows 15 Go Create App Edit Script

Number	Elapsed	Statement	Feedback	Rows
46	0.00	INSERT INTO ordered_items (quantity_ordered, quantity_shippe	ORA-00001: unique constraint (WKSP_OQRAVDKDHVAJDE.ORDERED_ITEM_PK) violated	-
47	0.00	INSERT INTO ordered_items (quantity_ordered, quantity_shippe	ORA-00001: unique constraint (WKSP_OQRAVDKDHVAJDE.ORDERED_ITEM_PK) violated	-

Download

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

Answer:

```
1  INSERT INTO teams(id, name, Number_of_players, discount) VALUES ('t004','Jets','10','5')
```

Results | Explain | Describe | Saved SQL | History

```
1 row(s) inserted.

0.02 seconds
```

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

Answer:

```
1  INSERT INTO customers (ctr_number,email,first_name,last_name,phone_number,current_balance,loyalty_card_number)
2  VALUES ('c02001','brianrog@hootech.com','Brian','Rogers','01654564898','-5','lc4587')
```

Results | Explain | Describe | Saved SQL | History

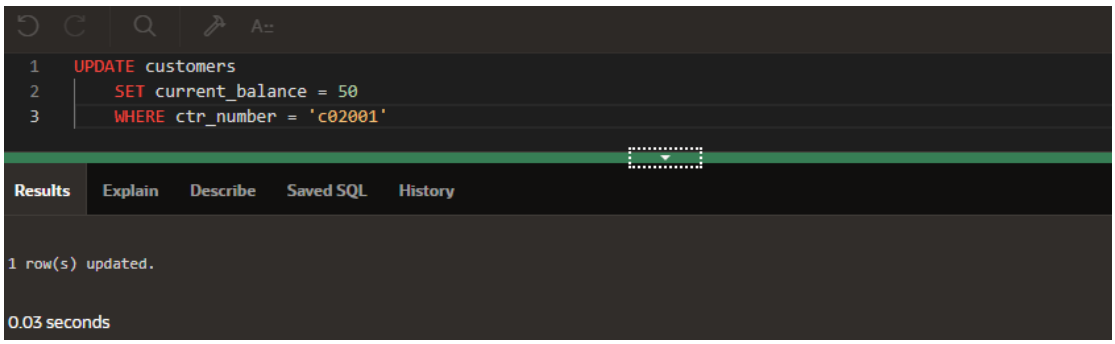
```
1 row(s) inserted.

0.04 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.

Answer:

a) Information violates

b) Change the current balance 50 and data updated



Section 6 Lesson 4 Exercisee 2: Data Manipulation Language

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

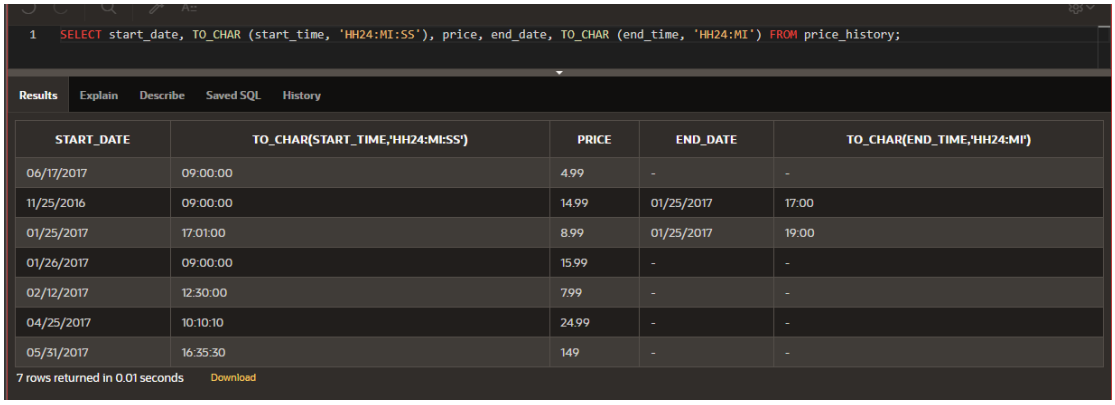
In this exercise you will populate and work with the data that is stored in the databse system.

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;
```

Answer:



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

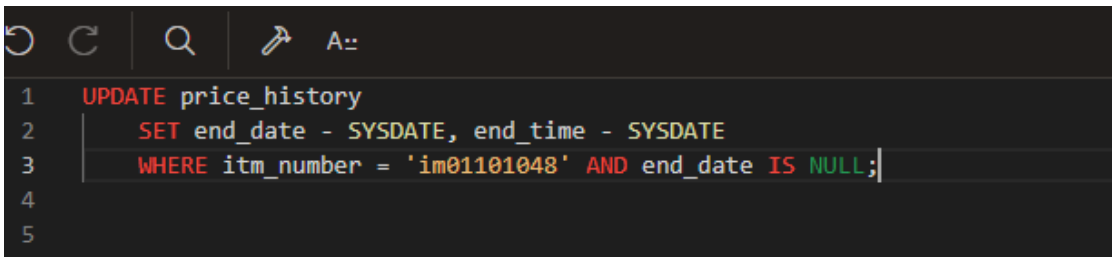
Results | Explain | Describe | Saved SQL | History

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	-	-

7 rows returned in 0.01 seconds Download

2. Obl is going to update the price of the premium bat so you will need 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 lastest price.

Answer:



```
1  UPDATE price_history
2      SET end_date - SYSDATE, end_time - SYSDATE
3      WHERE itm_number = 'im01101048' AND end_date IS NULL;
4
5
```