Database Programming with SQL

12-1: INSERT Statements

1. Give two examples of why it is important to be able to alter the data in a database.
    a. **Updating Inventory**: For a business, keeping the inventory database up-to-date is essential. If product quantities are not altered based on sales and new stock, it could lead to stock-outs or overstocking, disrupting business operations.
    b. **Correcting Customer Information**: Over time, customer contact information such as addresses, phone numbers, or emails may change. Altering this data in the database ensures that communication remains effective and accurate, which is crucial for customer relationship management.
2. DJs on Demand just purchased four new CDs. Use an explicit INSERT statement to add each CD to the copy_d_cds table. After completing the entries, execute a SELECT * statement to verify your work.

CD_Number	Title	Producer	Year
97	Celebrate the Day	R & B Inc.	2003
98	Holiday Tunes for All Ages	Tunes are Us	2004
99	Party Music	Old Town Records	2004
100	Best of Rock and Roll	Old Town Records	2004

```
1   SELECT * FROM COPY_D_CDS
```

Results | Explain | Describe | Saved SQL | History

CD_NUMBER	TITLE	PRODUCER	YEAR
98	Holiday Tunes for All Ages	Tunes are Us	2003
97	Celebrate the Day	R & B Inc.	2003
99	Party Music	Old Town Records	2004
100	Best of Rock and Roll	Old Town Records	2004

3. DJs on Demand has two new events coming up. One event is a fall football party and the other event is a sixties theme party. The DJs on Demand clients requested the songs shown in the table for their events. Add these songs to the copy_d_songs table using an implicit INSERT statement.

ID	Title	Duration	Type_Code
52	Surfing Summer	Not Known	12
53	Victory Victory	5 min	12

```
INSERT INTO copy_d_songs (ID, Title, Duration, Type_Code)
VALUES (52, 'Surfing Summer', 'Not known', 12);

INSERT INTO copy_d_songs (ID, Title, Duration, Type_Code)
VALUES (53, 'Victory Victory', '5 min', 12);

Select * from copy_d_songs
```

Results | Explain | Describe | Saved SQL | History

ID	TITLE	DURATION	TYPE_CODE
53	Victory Victory	5 min	12
52	Surfing Summer	Not known	12

4. Add the two new clients to the copy_d_clients table. Use either and implicit or an explicit INSERT.

Client_Number	First_Name	Last_Name	Phone	Email
6655	Ayako	Dahish	3608859030	dahisha@harbor.net
6689	Nick	Neuville	9048953049	nnicky@charter.net

```
INSERT INTO copy_d_clients (Client_Number, First_Name, Last_Name, Phone, Email)
VALUES (6655, 'Ayako', 'Dahish', '3608859030', 'dahisha@harbor.net');

INSERT INTO copy_d_clients (Client_Number, First_Name, Last_Name, Phone, Email)
VALUES (6689, 'Nick', 'Neuville', '9048953049', 'nnicky@charter.net');

SELECT * FROM COPY_D_CLIENTS
```

Results | Explain | Describe | Saved SQL | History

CLIENT_NUMBER	FIRST_NAME	LAST_NAME	PHONE	EMAIL
6655	Ayako	Dahish	3608859030	dahisha@harbor.net
6689	Nick	Neuville	9048953049	nnicky@charter.net

5. Add the new client's events to the copy_d_events table. The cost of each event has not been determined at this date.

```
INSERT INTO copy_d_events (ID, Name, Event_Date, Description, Cost, Venue_ID, Package_Code, Theme_Code, Client_Number)
VALUES (110, 'Ayako Anniversary', TO_DATE('07-Jul-2004', 'DD-Mon-YYYY'), 'Party for 50, sixties dress, decorations', NULL,

INSERT INTO copy_d_events (ID, Name, Event_Date, Description, Cost, Venue_ID, Package_Code, Theme_Code, Client_Number)
VALUES (115, 'Neuville Sports Banquet', TO_DATE('09-Sep-2004', 'DD-Mon-YYYY'), 'Barbecue at residence, college alumni, 10

SELECT * FROM COPY_D_EVENTS
```

Results | Explain | Describe | Saved SQL | History

ID	NAME	EVENT_DATE	DESCRIPTION	COST	VENUE_ID	PACKAGE_CODE	THEME_CODE	CLIENT_NUMBER
110	Ayako Anniversary	07-Jul-2004	Party for 50, sixties dress, decorations	-	245	79	240	6655
115	Neuville Sports Banquet	09-Sep-2004	Barbecue at residence, college alumni, 100 people	-	315	87	340	6689

6. Create a table called rep_email using the following statement:
   CREATE TABLE rep_email (
   id NUMBER(3) CONSTRAINT rel_id_pk PRIMARY KEY,
   first_name VARCHAR2(10),
   last_name VARCHAR2(10),
   email_address VARCHAR2(10))

   Populate this table by running a query on the employees table that includes only those employees who are REP's.

```
1  INSERT INTO rep_email (id, first_name, last_name, email_address)
2  SELECT employee_id, first_name, last_name, email
3  FROM employees
4  WHERE job_id = 'REP';
5
6  SELECT * FROM REP_EMAIL
```

Results | Explain | Describe | Saved SQL | History

no data found

12-2: Updating Column Values and Deleting Rows

1. Monique Tuttle, the manager of Global Fast Foods, sent a memo requesting an immediate change in prices. The price for a strawberry shake will be raised from $3.59 to $3.75, and the price for fries will increase to $1.20. Make these changes to the copy_f_food_items table.

```
1  UPDATE copy_f_food_items
2  SET Price = 3.75
3  WHERE Description = 'Strawberry Shake';
4
5  UPDATE copy_f_food_items
6  SET Price = 1.20
7  WHERE Description = 'Fries';
8
9  select * from copy_f_food_items
```

Results | Explain | Describe | Saved SQL | History

FOOD_ITEM_NUMBER	DESCRIPTION	PRICE	REGULAR_CODE	PROMO_CODE
90	Fries	1.2	20	-
93	Strawberry Shake	3.75	-	110

2. Bob Miller and Sue Doe have been outstanding employees at Global Fast Foods. Management has decided to reward them by increasing their overtime pay. Bob Miller will receive an additional $0.75 per hour and Sue Doe will receive an additional $0.85 per hour. Update the copy_f_staffs table to show these new values. (Note: Bob Miller currently doesn't get overtime pay. What function do you need to use to convert a null value to 0?)

```
UPDATE copy_f_staffs
SET OVERTIME_RATE = NVL(OVERTIME_RATE, 0) + 0.75
WHERE FIRST_NAME = 'Bob' AND LAST_NAME = 'Miller';

UPDATE copy_f_staffs
SET OVERTIME_RATE = NVL(OVERTIME_RATE, 0) + 0.85
WHERE FIRST_NAME = 'Sue' AND LAST_NAME = 'Doe';

Select * from copy_f_staffs
```

Results | Explain | Describe | Saved SQL | History

ID	FIRST_NAME	LAST_NAME	BIRTHDATE	SALARY	OVERTIME_RATE	TRAINING	STAFF_TYPE	MANAGER_ID	MANAGER_BUDGET	MANAGER_TARGET
12	Sue	Doe	01-Jul-1980	6.75	11.1	-	Order Taker	19	-	-
9	Bob	Miller	19-Mar-1979	10	.75	-	Grill Cook	19	-	-
19	Monique	Tuttle	30-Mar-1969	60	-	-	Manager	-	50000	70000

3. Add the orders shown to the Global Fast Foods copy_f_orders table

```
select * from copy_f_orders
```

Results | Explain | Describe | Saved SQL | History

ORDER_NUMBER	ORDER_DATE	ORDER_TOTAL	CUST_ID	STAFF_ID
5678	10-Dec-2002	103.02	123	12
5680	12-Jun-2004	159.78	145	9
5691	23-Sep-2004	145.98	225	12
5701	04-Jul-2004	229.31	230	12

4. Add the new customers shown below to the copy_f_customers table. You may already have added Katie Hernandez. Will you be able to add all these records successfully?

```
select * from copy_f_customers
```

Results | Explain | Describe | Saved SQL | History

ID	FIRST_NAME	LAST_NAME	ADDRESS	CITY	STATE	ZIP	PHONE_NUMBER
123	Cole	Bee	123 Main Street	Orlando	FL	32838	4075558234
456	Zoe	Twee	1009 Oliver Avenue	Boston	MA	12889	7098675309
225	Daniel	Spode	1923 Silverado	Denver	CO	80219	7193343523
145	Katie	Hernandez	92 Chico Way	Los Angeles	CA	90001	8586667641
230	Adam	Zurn	5 Admiral Way	Seattle	WA	98008	4258879009

5. Sue Doe has been an outstanding Global Foods staff member and has been given a salary raise. She will now be paid the same as Bob Miller. Update her record in copy_f_staffs.

```
1  Select * from copy_f_staffs
```

Results | Explain | Describe | Saved SQL | History

ID	FIRST_NAME	LAST_NAME	BIRTHDATE	SALARY	OVERTIME_RATE	TRAINING	STAFF_TYPE	MANAGER_ID	MANAGER_BUDGET	MANAGER_TARGET
12	Sue	Doe	01-Jul-1980	10	11.1	-	Order Taker	19	-	-
9	Bob	Miller	19-Mar-1979	10	.75	-	Grill Cook	19	-	-
19	Monique	Tuttle	30-Mar-1969	60	-	-	Manager	-	50000	70000

6. Global Fast Foods is expanding their staff. The manager, Monique Tuttle, has hired Kai Kim. Not all information is available at this time, but add the information shown here.

```
1  select * from copy_f_staffs
```

Results | Explain | Describe | Saved SQL | History

ID	FIRST_NAME	LAST_NAME	BIRTHDATE	SALARY	OVERTIME_RATE	TRAINING	STAFF_TYPE	MANAGER_ID	MANAGER_BUDGET	MANAGER_TARGET
25	Kai	Kim	03-Nov-1988	6.75	-	-	Order Taker	-	-	-
12	Sue	Doe	01-Jul-1980	10	11.1	-	Order Taker	19	-	-
9	Bob	Miller	19-Mar-1979	10	.75	-	Grill Cook	19	-	-
19	Monique	Tuttle	30-Mar-1969	60	-	-	Manager	-	50000	70000

7. Now that all the information is available for Kai Kim, update his Global Fast Foods record to include the following: Kai will have the same manager as Sue Doe. He does not qualify for overtime. Leave the values for training, manager budget, and manager target as null.

```
1  UPDATE copy_f_staffs
2  SET
3      MANAGER_ID = (SELECT MANAGER_ID FROM copy_f_staffs WHERE FIRST_NAME = 'Sue' AND LAST_NAME = 'Doe
4      OVERTIME_RATE = NULL,
5      TRAINING = NULL,
6      MANAGER_BUDGET = NULL,
7      MANAGER_TARGET = NULL
8  WHERE
9      ID = 25;
```

Results | Explain | Describe | Saved SQL | History

```
1 row(s) updated.
```

8. Kim Kai has decided to go back to college and does not have the time to work and go to school. Delete him from the Global Fast Foods staff. Verify that the change was made.

```
1  SELECT * FROM copy_f_staffs
2  WHERE ID = 25;
```

Results | Explain | Describe | Saved SQL | History

no data found

9. Create a copy of the employees table and call it lesson7_emp; Once this table exists, write a correlated delete statement that will delete any employees from the lesson7_employees table that also exist in the job_history table.

```
1  SELECT * FROM lesson7_emp;
```

Results | Explain | Describe | Saved SQL | History

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID	BONUS	DATE_OF_BIRTH	OVERTI
100	Steven	King	SKING	515.123.4567	17-Jun-2002	AD_PRES	24000	-	-	90	-	15-Jan-1990	-
205	Shelley	Higgins	SHIGGINS	515.123.8080	07-Jun-2009	AC_MGR	12000	-	101	110	-	-	-
206	William	Gietz	WGIETZ	515.123.8181	07-Jun-2009	AC_ACCOUNT	8300	-	205	110	-	-	-
149	Eleni	Zlotkey	EZLOTKEY	011.44.1344.429018	29-Jan-2015	SA_MAN	10500	.2	100	80	1500	-	-
174	Ellen	Abel	EABEL	011.44.1644.429267	11-May-2011	SA_REP	11000	.3	149	80	1700	-	-
178	Kimberely	Grant	KGRANT	011.44.1644.429263	24-May-2014	SA_REP	7000	.15	149	-	-	-	-
124	Kevin	Mourgos	KMOURGOS	650.123.5234	16-Nov-2014	ST_MAN	5800	-	100	50	-	-	-
141	Trenna	Rajs	TRAJS	650.121.8009	17-Oct-2010	ST_CLERK	3500	-	124	50	-	-	-

(Didn't return any of the 5 deleted rows)

12-3: DEFAULT Values, MERGE, and Multi-Table Inserts Practice Activities

1. When would you want a DEFAULT value?

- For ease and consistency, for instance, setting a created_at column to SYSDATE by default guarantees that each new record receives the most recent timestamp without the user having to supply it.
- When applying business rules: For instance, setting a status column to Active by default when inserting a new record.
- You want to automatically populate a column with a specified value when a new record is inserted, and the value is not explicitly provided.
- To avoid null values in columns that are required or have a logical default.

2. Currently, the Global Foods F_PROMOTIONAL_MENUS table START_DATE column does not have SYSDATE set as DEFAULT. Your manager has decided she would like to be able to set the starting date of promotions to the current day for some entries. This will require three steps:
    a. In your schema, make a copy of the Global Foods F_PROMOTIONAL_MENUS table using the following SQL statement: CREATE TABLE copy_f_promotional_menus AS (SELECT * FROM f_promotional_menus)

```
1  CREATE TABLE copy_f_promotional_menus AS
2  SELECT * FROM f_promotional_menus;
```

Results | Explain | Describe | Saved SQL | History

```
Table created.
```

    b. Alter the current START_DATE column attributes using: ALTER TABLE copy_f_promotional_menus MODIFY(start_date DATE DEFAULT SYSDATE)

```
1  ALTER TABLE copy_f_promotional_menus
2  MODIFY (start_date DATE DEFAULT SYSDATE);
```

Results | Explain | Describe | Saved SQL | History

```
Table altered.
```

    c. INSERT the new information and check to verify the results. INSERT a new row into the copy_f_promotional_menus table for the manager's new promotion. The promotion code is 120. The name of the promotion is 'New Customer.' Enter DEFAULT for the start date and '01-Jun-2005' for the ending date. The giveaway is a 10% discount coupon. What was the correct syntax used?

```
1  Select * from copy_f_promotional_menus
```

Results | Explain | Describe | Saved SQL | History

CODE	NAME	START_DATE	END_DATE	GIVE_AWAY
100	Back to School	01-Sep-2004	30-Sep-2004	ballpen and highlighter
110	Valentines Special	10-Feb-2004	15-Feb-2004	small box of chocolates
120	New Customer	12-Nov-2024	01-Jun-2005	10% discount coupon