

# Práctica: 12-2

Agustín Alejandro Mota Hinojosa

## Terminología

- Modifica filas existentes en una tabla: **update**
- Recupera información de una tabla y utiliza esta información para actualizar otra tabla: subconsulta con **update**
- Se asegura de que los datos cumplen un juego de reglas predefinido: restricción
- Suprime la información en una tabla enlazada en función de lo que se ha suprimido en la otra tabla: subconsulta con **delete**
- Elimina filas existentes en una tabla: **delete**

## Inténtelo/Resuévalo

1. Monique Tuttle, la jefa de Global Fast Foods, ha enviado una circular solicitando un cambio inmediato de los precios. El precio de un batido de fresa subirá de 3,59 a 3,75 \$, y el precio de las patatas fritas subirá a 1,20 \$. Realice los siguientes cambios en la tabla `copy_f_food_items`.

```
create table copy_f_food_items
as
select * from f_food_items;

-- fila del batido de fresa
update
  copy_f_food_items
set
  price = 3.75
where
  food_item_number = 93;

-- fila de las patatas fritas
update
  copy_f_food_items
set
  price = 1.20
where
  food_item_number = 90;
```

2. Bob Miller y Sue Doe han sido empleados destacados de Global Fast Foods. La dirección ha decidido recompensarlos aumentando su paga por horas extra. Bob Miller recibirá 0,75 \$ adicionales por hora y Sue Doe recibirá 0,85 \$ adicionales por hora. Actualice la tabla `copy_f_staffs` con estos nuevos valores. (Nota: Bob Miller actualmente no recibe paga por horas extra. ¿Qué función debe utilizar para convertir un valor nulo en 0?)

```
create table copy_f_staffs
as
select * from f_staffs;

-- Salario de bob
-- Para convertir un valor nulo se utiliza la función NVL()
update
  copy_f_staffs empleado
set
  empleado.overtime_rate = (
select
```

```

    nvl(empleado.overtime_rate, 0) + 0.75
from
    copy_f_staffs sub_empleado
where
    empleado.id = sub_empleado.id
)
where
    id = 9;

-- Salario de Sue Doe
update
    copy_f_staffs
set
    overtime_rate = overtime_rate + 85
where
    id = 12;

```

3. Agregue los pedidos que se muestran a continuación a la tabla `copy_f_orders` de Global Fast Foods:

```

create copy_f_orders
as
select * from f_orders;

insert into copy_f_orders (
    order_number,order_date,order_total,
    cust_id,staff_id
)
values (
    5680,
    to_date('June 12, 2004','MM-DD-YYYY'),
    159.78,
    145, 9
);

insert into copy_f_orders (
    order_number,order_date,order_total,
    cust_id,staff_id
)
values (
    5691,
    to_date('09-23-2004','MM-DD-YYYY'),
    145.98,
    225,
    12
);

insert into copy_f_orders (
    order_number,order_date,order_total,
    cust_id,staff_id
)
values (
    5701,
    to_date('July 4, 2004','MM-DD-YYYY'),
    229.31,
    230,
    12
);

```

4. Agregue los nuevos clientes que se muestran a continuación a la tabla `copy_f_customers`. Es posible que ya haya agregado a Katie Hernandez. ¿Podrá agregar todos estos registros correctamente?

```

create table copy_f_customers

```

```

as
select * from f_customers;

insert into copy_f_customers (
    id,first_name,last_name,
    address,city,state,
    zip,phone_number
)
values (
    145,
    'Katie',
    'Hernandez',
    '92 Chico Way',
    'Los Angeles',
    'CA',
    98008,
    '8586667641'
);

insert into copy_f_customers (
    id,first_name,last_name,
    address,city,state,
    zip,phone_number
)
values (
    225,
    'Daniel',
    'Spode',
    1923,
    'Silverado',
    'Denver',
    'CO',
    80219,
    '7193343523'
);

insert into copy_f_customers (
    id,first_name,last_name,
    address,city,state,
    zip,phone_number
)
values (
    230,
    'Adam',
    'Zurn',
    '5 Admiral Way',
    'Seattle',
    'WA',
    null,
    4258879009
);

```

**RE:** No, el problema está en el último insert ( $id = 230$ ) donde no existe un valor para la columna ZIP, y según un constraint de la tabla no se pueden ingresar valores nulos en esa columna.

5. Sue Doe ha sido un miembro destacado del personal de Global Fast Foods y se le ha concedido un aumento salarial. Ahora recibirá la misma paga que Bob Miller. Actualice su registro en `copy_f_staffs`.

```

update copy_f_staffs
set salary = (
    select salary

```

```

        from copy_f_staffs
    where id = 9
)
where id = 12;

```

6. Global Fast Foods está ampliando su personal. La jefa, Monique Tuttle, ha contratado a Kai Kim. No está disponible toda la información en este momento, pero agregue la información que se muestra aquí.

```

insert into copy_f_staffs (
    id,first_name,last_name,
    birthdate,salary,staff_type
)
values (
    25,
    'Kai',
    'Kim',
    to_date('3-Nov-1988','DD-MM-YYYY'),
    6.75,
    'Order Taker'
);

```

7. Ahora que está disponible toda la información de Kai Kim, actualice su registro en Global Fast Foods para incluir lo siguiente: Kai tendrá el mismo jefe que Sue Doe. No podrá trabajar horas extra. Deje los valores de formación, presupuesto del jefe y objetivo del jefe como null.

```

update copy_f_staffs
set manager_id = (
    select manager_id
    from copy_f_staffs
    where id = 12
)
where id = 25;

```

8. Ejecute la siguiente sentencia SQL. Registre los resultados.

```

delete from departments
where department_id = 60;

```

**RE:** el elemento con id = 60 de la tabla departments fue eliminado:

1 row(s) deleted.

0.01 seconds

9. Kim Kai ha decidido volver a la universidad y no tiene tiempo para trabajar y asistir a las clases. Suprímalo de la plantilla de Global Fast Foods. Verifique que el cambio ha sido correcto.

```

delete from copy_f_staffs
where id = 25;

```

10. Cree una copia de la tabla de empleados y denomínela `lesson7_employees`; Una vez creada esta tabla, escriba una sentencia delete correlacionada que suprima cualquier empleado de la tabla `lesson7_employees` que también exista en la tabla `job_history`.

```

create table lesson7_employees
as select * from employees;

delete from lesson7_employees
where employee_id in (
    select employee_id
    from job_history
);

```