

REPORTE PRACTICA 23

EVIDENCIAS INSTRUCCION 5

```
banco=# CREATE PROCEDURE ctenuevo(a integer, b varchar(15), c varchar(15),d
varchar(12))
banco=# LANGUAGE SQL
banco=# AS $$
banco$$ INSERT INTO cliente VALUES(a,b,c,d);
banco$$ $$;
CREATE PROCEDURE
banco=#
```

EVIDENCIAS INSTRUCCION 8

```
banco=# CALL ctenuevo(31,'Laura','Ramos','443 345 9823');
CALL
banco=# CALL ctenuevo(31,'Miguel','Flores','443 331 1189');
ERROR:  duplicate key value violates unique constraint "cliente_pkey"
DETAIL:  Key (clavecte)=(31) already exists.
CONTEXT:  SQL function "ctenuevo" statement 1
banco=# CALL ctenuevo(52,'Miguel','Flores','443 331 1189');
CALL
banco=# SELECT * FROM CLIENTE;
```

clavecte	nombre	apellido	teléfono
26	Lucy Dora	Lara	443 342 5287
35	Jaime	Nava	443 312 4194
45	Dora	Bustos	433 312 1570
59	Lucy Dora	Borja	443 314 5729
41	Rita	Luna	443 276 3197
12	Jorge	Gómez	443 312 5720
86	Max	Smart	443 107 1664
99	Gina	Torre	443 376 5693
31	Laura	Ramos	443 345 9823
52	Miguel	Flores	443 331 1189

(10 rows)

EVIDENCIA INSTRUCCION 10

```
banco=# CREATE OR REPLACE PROCEDURE Creacuenta(num INTEGER, cte INTEGER, suc
INTEGER)
banco=# LANGUAGE plpgsql
banco=# AS $$
banco$$ BEGIN
banco$$ INSERT INTO cuentas VALUES (num,cte,suc,0);
banco$$ EXCEPTION WHEN OTHERS THEN
banco$$ RAISE NOTICE 'Algun dato es erróneo';
banco$$ END;
banco$$ $$;
CREATE PROCEDURE
banco=#
```

EVIDENCIA INSTRUCCION 14

```
banco=# SELECT * FROM cuentas;
 numero | clavecte | clavesuc | saldo
-----+-----+-----+-----
 3491   |      35  |      256 | 1270.75
 6331   |      12  |      275 | 498.04
 4323   |      45  |      341 | 786.69
 5179   |      41  |      341 | 5875.35
 6993   |      59  |      341 | 550
(5 rows)
```

```
banco=# CALL Creacuenta(6127,52,183);
CALL
banco=# CALL Creacuenta(34911,31,341);
CALL
banco=# CALL Creacuenta(34911,31,341);
NOTICE: Algun dato es erróneo
CALL
```

```
banco=# SELECT * FROM CUENTAS;
 numero | clavecte | clavesuc | saldo
-----+-----+-----+-----
 3491   |      35  |      256 | 1270.75
 6331   |      12  |      275 | 498.04
 4323   |      45  |      341 | 786.69
 5179   |      41  |      341 | 5875.35
 6993   |      59  |      341 | 550
 6127   |      52  |      183 | 0
 34911  |      31  |      341 | 0
(7 rows)
```

EVIDENCIA INSTRUCCION 17

```
banco=# CREATE TABLE empleado(clave INTEGER PRIMARY KEY, nombre VARCHAR(25),
apellido_pat VARCHAR(25), apellido_mat VARCHAR(25), "teléfono" VARCHAR(12),
sueldo MONEY);
CREATE TABLE
banco=# \d empleado
```

Column	Type	Collation	Nullable	Default
clave	integer		not null	
nombre	character varying(25)			
apellido_pat	character varying(25)			
apellido_mat	character varying(25)			
teléfono	character varying(12)			
sueldo	money			

```
Indexes:
    "empleado_pkey" PRIMARY KEY, btree (clave)
```

EVIDENCIA INSTRUCCION 19

```
banco=# CREATE PROCEDURE emp_alta(cve INTEGER, nom varchar(25), ApP VARCHAR(
25), ApM VARCHAR(25), tel VARCHAR(12), Sue INTEGER)
banco=# LANGUAGE SQL
banco=# AS $$
banco$$ INSERT INTO empleado VALUES(cve,nom,ApP,ApM,tel,Sue);
banco$$ $$;
CREATE PROCEDURE
```

EVIDENCIA INSTRUCCION 22

```

banco=# CALL emp_alta(443,'Lucas','Montes','Torres','443 312 1754',5500)
banco=# ;
CALL
banco=# CALL emp_alta(443,'Martha','Mares','Rosas','443 324 1689',6700);
ERROR: duplicate key value violates unique constraint "empleado_pkey"
DETAIL: Key (clave)=(443) already exists.
CONTEXT: SQL function "emp_alta" statement 1
banco=# CALL emp_alta(512,'Martha','Mares','Rosas','443 324 1689',6700);
CALL
banco=# CALL emp_alta(517,'Karla','Cuevas','Mina','443 212 4317',8200);
CALL
banco=# CALL emp_alta(597,'Alberto','Vidal','Luna','443 331 6536',7700);
CALL
banco=# SELECT * FROMM empleado;
ERROR: syntax error at or near "FROMM"
LINE 1: SELECT * FROMM empleado;
                ^
banco=# SELECT * FROM empleado;
 clave | nombre | apellido_pat | apellido_mat | teléfono | sueldo
-----+-----+-----+-----+-----+-----
  443  | Lucas  | Montes      | Torres       | 443 312 1754 | $5,500.00
  512  | Martha | Mares       | Rosas        | 443 324 1689 | $6,700.00
  517  | Karla  | Cuevas      | Mina          | 443 212 4317 | $8,200.00
  597  | Alberto | Vidal       | Luna         | 443 331 6536 | $7,700.00
(4 rows)

```

EVIDENCIA INSTRUCCION 27

```

banco=# CREATE TABLE movimientos (folio SERIAL PRIMARY KEY, fecha DATE, cuenta INTEGER, empleado INTEGER, importe NUMERIC, tipo CHAR(1));
CREATE TABLE
banco=# ALTER TABLE movimientos ADD CONSTRAINT fk_cuenta FOREIGN KEY (Cuenta) REFERENCES cuentas(Numero) ON UPDATE CASCADE;
ALTER TABLE
banco=# ALTER TABLE movimientos ADD CONSTRAINT fk_empleado FOREIGN KEY (empleado) REFERENCES empleado(clave) ON UPDATE CASCADE;
ALTER TABLE
banco=# \d movimientos

```

Column	Type	Collation	Nullable	Default
folio	integer		not null	nextval('movimientos_folio_seq'::regclass)
fecha	date			
cuenta	integer			
empleado	integer			
importe	numeric			
tipo	character(1)			

```

Indexes:
    "movimientos_pkey" PRIMARY KEY, btree (folio)
Foreign-key constraints:
    "fk_cuenta" FOREIGN KEY (cuenta) REFERENCES cuentas(numero) ON UPDATE CASCADE
    "fk_empleado" FOREIGN KEY (empleado) REFERENCES empleado(clave) ON UPDATE CASCADE

```

EVIDENCIA INSTRUCCION 29

```
banco=# CREATE OR REPLACE PROCEDURE depósito(cta INTEGER, emp INTEGER, imp NUMERIC)
banco=# LANGUAGE plpgsql
banco=# AS $$
banco$# BEGIN
banco$# UPDATE cuentas SET Saldo=Saldo+imp WHERE numero=cta;
banco$# INSERT INTO movimientos(Fecha, Cuenta, Empleado, Importe, Tipo) VALUES(current_date,cta,emp,imp,'D');
banco$# END;
banco$# $$;
CREATE PROCEDURE
banco=# |
```

EVIDENCIA INSTRUCCION 32

```
banco=# CALL depósito(6127,443,1500);
CALL
banco=# CALL depósito(6993,512,2450);
CALL
banco=# SELECT * FROM movimientos;
 folio |   fecha   | cuenta | empleado | importe | tipo
-----+-----+-----+-----+-----+-----
      1 | 2025-04-15 |   6127 |       443 |    1500 | D
      2 | 2025-04-15 |   6993 |       512 |    2450 | D
(2 rows)

banco=# select * FROM cuentas;
 numero | clavecte | clavesuc | saldo
-----+-----+-----+-----
    3491 |        35 |        256 | 1270.75
    6331 |        12 |        275 |  498.04
    4323 |        45 |        341 |  786.69
    5179 |        41 |        341 | 5875.35
   34911 |        31 |        341 |      0
    6127 |        52 |        183 |    1500
    6993 |        59 |        341 |    3000
(7 rows)
```

EVIDENCIA INSTRUCCION 34

```
banco=# CREATE OR REPLACE PROCEDURE retiro (cta INTEGER, emp INTEGER, imp NUMERIC)
banco=# LANGUAGE plpgsql
banco=# AS $$
banco$$ DECLARE
banco$$ s NUMERIC;
banco$$ BEGIN
banco$$ SELECT Saldo INTO s FROM cuentas WHERE numero=cta;
banco$$ IF s<imp THEN
banco$$ RAISE EXCEPTION 'Datos proporcionados son erróneos';
banco$$ END IF;
banco$$ UPDATE cuentas SET saldo=saldo-imp WHERE numero=cta;
banco$$ INSERT INTO movimientos.fecha,cuenta,empleado,importe, tipo) VALUES(
current_date,cta,emp, imp, 'R');
banco$$ END;
banco$$ $$;
CREATE PROCEDURE
banco=# |
```

EVIDENCIA INSTRUCCION 37

```
banco=# CALL retiro(6127,517,1000);
CALL
banco=# CALL retiro(6993,597,2000);
CALL
banco=# CALL retiro(6127,517,1000);
ERROR: Datos proporcionados son erróneos
CONTEXT: PL/pgSQL function retiro(integer,integer,numeric) line 7 at RAISE
banco=# SELECT * FROM movimientos;
```

folio	fecha	cuenta	empleado	importe	tipo
1	2025-04-15	6127	443	1500	D
2	2025-04-15	6993	512	2450	D
3	2025-04-15	6127	517	1000	R
4	2025-04-15	6993	597	2000	R

(4 rows)

```
banco=# SELECT * FROM cuentas;
```

numero	clavecte	clavesuc	saldo
3491	35	256	1270.75
6331	12	275	498.04
4323	45	341	786.69
5179	41	341	5875.35
34911	31	341	0
6127	52	183	500
6993	59	341	1000

(7 rows)