**Complete Implementation: User History Management System**

**Introduction**

This document provides the complete implementation for the library database history feature, including design rationale, SQL code, and testing approach. The system creates history tables for both users and loans, ensuring that when a user is removed from the system, their information is preserved along with all their associated loan records.

**Implementation Code**

**History Tables Creation**

-- Create history table for users

CREATE TABLE history\_users (

USER\_ID CHAR(10),

ID\_CARD CHAR(17) NOT NULL,

NAME VARCHAR2(80) NOT NULL,

SURNAME1 VARCHAR2(80) NOT NULL,

SURNAME2 VARCHAR2(80),

BIRTHDATE DATE NOT NULL,

TOWN VARCHAR2(50) NOT NULL,

PROVINCE VARCHAR2(22) NOT NULL,

ADDRESS VARCHAR2(150) NOT NULL,

EMAIL VARCHAR2(100),

PHONE NUMBER(9) NOT NULL,

TYPE CHAR(1) NOT NULL,

BAN\_UP2 DATE,

DELETION\_DATE DATE DEFAULT SYSDATE NOT NULL,

CONSTRAINT pk\_history\_users PRIMARY KEY(USER\_ID)

);

-- Create history table for loans

CREATE TABLE history\_loans (

SIGNATURE CHAR(5),

USER\_ID CHAR(10),

STOPDATE DATE,

TOWN VARCHAR2(50) NOT NULL,

PROVINCE VARCHAR2(22) NOT NULL,

TYPE CHAR(1) NOT NULL,

TIME NUMBER(5) NOT NULL,

RETURN DATE,

MOVED\_TO\_HISTORY DATE DEFAULT SYSDATE NOT NULL,

CONSTRAINT pk\_history\_loans PRIMARY KEY(SIGNATURE, USER\_ID, STOPDATE)

);

**User Deletion Trigger**

CREATE OR REPLACE TRIGGER trg\_user\_delete

BEFORE DELETE ON users

FOR EACH ROW

DECLARE

v\_posts\_count NUMBER;

BEGIN

INSERT INTO history\_users (

USER\_ID, ID\_CARD, NAME, SURNAME1, SURNAME2, BIRTHDATE,

TOWN, PROVINCE, ADDRESS, EMAIL, PHONE, TYPE, BAN\_UP2

) VALUES (

:OLD.USER\_ID, :OLD.ID\_CARD, :OLD.NAME, :OLD.SURNAME1, :OLD.SURNAME2, :OLD.BIRTHDATE,

:OLD.TOWN, :OLD.PROVINCE, :OLD.ADDRESS, :OLD.EMAIL, :OLD.PHONE, :OLD.TYPE, :OLD.BAN\_UP2

);

SELECT COUNT(\*) INTO v\_posts\_count

FROM posts p

WHERE p.USER\_ID = :OLD.USER\_ID;

IF v\_posts\_count > 0 THEN

DELETE FROM posts

WHERE USER\_ID = :OLD.USER\_ID;

END IF;

INSERT INTO history\_loans (

SIGNATURE, USER\_ID, STOPDATE, TOWN, PROVINCE, TYPE, TIME, RETURN

)

SELECT

SIGNATURE, USER\_ID, STOPDATE, TOWN, PROVINCE, TYPE, TIME, RETURN

FROM loans

WHERE USER\_ID = :OLD.USER\_ID;

DELETE FROM loans

WHERE USER\_ID = :OLD.USER\_ID;

END;

/

**Delete User Procedure**

CREATE OR REPLACE PROCEDURE delete\_user(p\_user\_id IN CHAR)

AS

BEGIN

DELETE FROM users

WHERE USER\_ID = p\_user\_id;

COMMIT;

DBMS\_OUTPUT.PUT\_LINE('User ' || p\_user\_id || ' has been deleted and moved to history.');

EXCEPTION

WHEN NO\_DATA\_FOUND THEN

DBMS\_OUTPUT.PUT\_LINE('User ' || p\_user\_id || ' not found.');

WHEN OTHERS THEN

ROLLBACK;

DBMS\_OUTPUT.PUT\_LINE('Error deleting user: ' || SQLERRM);

END delete\_user;

/

**Testing Instructions**

**Test 1: Deleting a User**

SQL> SET SERVEROUTPUT ON;

SQL> EXEC delete\_user('USER000001');

User USER000001 has been deleted and moved to history.

PL/SQL procedure successfully completed.

SQL> SELECT USER\_ID, NAME || ' ' || SURNAME1 AS NAME, TO\_CHAR(DELETION\_DATE, 'DD-MM-YYYY HH24:MI:SS') AS DELETED\_ON FROM history\_users;

USER\_ID

----------

NAME

--------------------------------------------------------------------------------

DELETED\_ON

-------------------

USER000001

Test User

04-04-2025 16:22:22

**Test 2: Deleting Another User**

SELECT \* FROM loans WHERE user\_id = '8631945440';

SIGNA USER\_ID STOPDATE TOWN

----- ---------- --------- --------------------------------------------------

PROVINCE T TIME RETURN

---------------------- - ---------- ---------

NG009 8631945440 21-NOV-24 Prados del Lago

Ciudad Real L 670 05-DEC-24

SQL> EXEC delete\_user('8631945440');

User 8631945440 has been deleted and moved to history.

PL/SQL procedure successfully completed.

SQL> SELECT USER\_ID, NAME || ' ' || SURNAME1 AS NAME, TO\_CHAR(DELETION\_DATE, 'DD-MM-YYYY HH24:MI:SS') AS DELETED\_ON FROM history\_users;

USER\_ID

----------

NAME

--------------------------------------------------------------------------------

DELETED\_ON

-------------------

USER000001

Test User

04-04-2025 16:22:22

8631945440

Alejandro Ochoa

04-04-2025 16:38:39

USER\_ID

----------

NAME

--------------------------------------------------------------------------------

DELETED\_ON

-------------------

SQL> SELECT \* FROM history\_loans WHERE user\_id = '8631945440';

SIGNA USER\_ID STOPDATE TOWN

----- ---------- --------- --------------------------------------------------

PROVINCE T TIME RETURN MOVED\_TO\_

---------------------- - ---------- --------- ---------

NG009 8631945440 21-NOV-24 Prados del Lago

Ciudad Real L 670 05-DEC-24 04-APR-25

SQL> SELECT \* FROM history\_loans

2 ;

SIGNA USER\_ID STOPDATE TOWN

----- ---------- --------- --------------------------------------------------

PROVINCE T TIME RETURN MOVED\_TO\_

---------------------- - ---------- --------- ---------

SIG03 USER000001 04-APR-25 Testville

Testland L 0 04-APR-25

SIG04 USER000001 04-APR-25 Testville

Testland L 0 04-APR-25

SIG05 USER000001 04-APR-25 Testville

Testland L 0 04-APR-25

SIGNA USER\_ID STOPDATE TOWN

----- ---------- --------- --------------------------------------------------

PROVINCE T TIME RETURN MOVED\_TO\_

---------------------- - ---------- --------- ---------

SIG01 USER000001 05-APR-25 Othertown

Otherprov R 0 04-APR-25

SIG01 USER000001 04-APR-25 Testville

Testland L 0 04-APR-25 04-APR-25

SIG02 USER000001 04-APR-25 Testville

Testland L 0 04-APR-25

SIGNA USER\_ID STOPDATE TOWN

----- ---------- --------- --------------------------------------------------

PROVINCE T TIME RETURN MOVED\_TO\_

---------------------- - ---------- --------- ---------

NG009 8631945440 21-NOV-24 Prados del Lago

Ciudad Real L 670 05-DEC-24 04-APR-25

**Conclusion**

This implementation provides a robust solution for maintaining historical data about deleted users and their loan activities. The design ensures data integrity while giving administrators the tools they need to manage and query historical information.