

Primera parte

DELIMITER //

CREATE PROCEDURE RegistrarAbono(

IN p_idprestamo INT,

IN p_montoabono FLOAT

)

BEGIN

DECLARE v_saldoactual FLOAT;

DECLARE v_estado VARCHAR(9);

-- Verificar si el préstamo existe y está activo

SELECT saldoactual, estado INTO v_saldoactual, v_estado

FROM prestamo

WHERE idprestamo = p_idprestamo;

IF v_estado IS NULL THEN

SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'El préstamo no existe.';

ELSEIF v_estado <> 'activo' THEN

SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'El préstamo no está activo.';

ELSEIF p_montoabono > v_saldoactual THEN

SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'El monto del abono no puede ser mayor que el saldo actual.';

ELSE

-- Insertar el abono en la tabla abono

INSERT INTO abono (montoabono, fechaabono, idprestamo)

VALUES (p_montoabono, CURDATE(), p_idprestamo);

END IF;

```
END //
```

```
DELIMITER ;
```

Disparadores

```
DELIMITER //
```

```
CREATE TRIGGER trg_actualizar_saldo
```

```
AFTER INSERT ON abono
```

```
FOR EACH ROW
```

```
BEGIN
```

```
    DECLARE v_saldoactual FLOAT;
```

```
    -- Obtener el saldo actual del préstamo
```

```
    SELECT saldoactual INTO v_saldoactual
```

```
    FROM prestamo
```

```
    WHERE idprestamo = NEW.idprestamo;
```

```
    -- Actualizar el saldo actual restando el monto abonado
```

```
    SET v_saldoactual = v_saldoactual - NEW.montoabono;
```

```
    -- Actualizar el saldo actual en la tabla prestamo
```

```
    UPDATE prestamo
```

```
    SET saldoactual = v_saldoactual,
```

```
        estado = CASE WHEN v_saldoactual = 0 THEN 'cancelado' ELSE estado END
```

```
    WHERE idprestamo = NEW.idprestamo;
```

```
END //
```

```
DELIMITER ;
```