1) Realiza un depósito de 300.00 en la cuenta de Juan

UPDATE Cuentas SET saldo = saldo + 300.00 WHERE titular = 'Juan';

2) Realiza un retiro de 200.00 de la cuenta de Maria.

UPDATE Cuentas SET saldo = saldo - 200.00 WHERE titular = 'Maria';

3) Realiza una transferencia de 150.00 de la cuenta de Pedro a la cuenta de Juan

UPDATE Cuentas SET saldo = saldo - 150.00 WHERE titular = 'Pedro';

UPDATE Cuentas SET saldo = saldo + 150.00 WHERE titular = 'Juan';



4) Realiza un depósito de 100.00 en la cuenta de Maria, pero simula un error en el proceso y observa cómo se maneja la transacción

START TRANSACTION;

UPDATE Cuentas SET saldo = saldo + 100.00 WHERE titular = 'Maria';

INSERT INTO Cuentas (titular, saldo) VALUES ('Maria', 100.00);

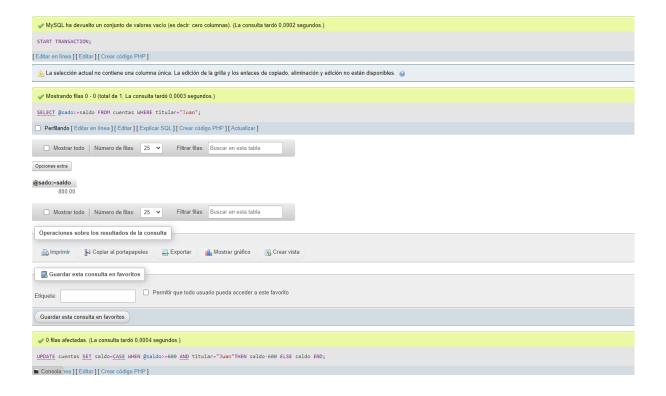
ROLLBACK;



5) Realiza un retiro de 600.00 de la cuenta de Juan, pero verifica si hay suficiente saldo antes de realizar el retiro.

START TRANSACTION; SELECT @sado:=saldo FROM cuentas WHERE titular="Juan";

UPDATE cuentas
SET saldo=CASE
WHEN @saldo>=600 AND titular="Juan"THEN saldo-600
ELSE saldo
END;



6) Realiza una transferencia de 1000.00 de la cuenta de Maria a la cuenta de Pedro, pero verifica si hay suficiente saldo antes de realizar la transferencia.

START TRANSACTION;
SELECT @sado:=saldo FROM cuentas WHERE titular="Maria";

UPDATE cuentas
SET saldo=CASE
WHEN @saldo>=1000 AND titular="Maria"THEN saldo-1000
WHEN @saldo>=1000 AND titular="Pedro"THEN saldo+1000
ELSE saldo
END:

7) <u>Crear una nueva cuenta para 'Carlos' con un saldo inicial de 500.00 y luego</u> realizar un depósito de 250.00.

INSERT INTO Cuentas (titular, saldo) VALUES ('Carlos', 500.00);

UPDATE Cuentas SET saldo = saldo + 250.00 WHERE titular = 'Carlos';



8) Cerrar la cuenta de 'Pedro' y transferir su saldo a la cuenta de 'Maria'

```
START TRANSACTION;

SELECT @saldo_pedro := saldo
FROM Cuentas
WHERE titular = 'Pedro';

UPDATE Cuentas
SET saldo = saldo + @saldo_pedro
WHERE titular = 'Maria';

DELETE FROM Cuentas
WHERE titular = 'Pedro';

COMMIT:
```



9) <u>Crear una nueva cuenta para 'Lucia' con un saldo inicial de 1000.00, pero simular un error en el proceso de creación de la cuenta.</u>

START TRANSACTION;

INSERT INTO cuentas (titular, saldo) VALUES ('Lucia', 1000.00); INSERT INTO cuentas (titular, saldo) VALUES ('Lucia', 500.00);

ROLLBACK;

```
✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0001 segundos.)

START TRANSACTION;

[Editar en línea] [Editar] [Crear código PHP]

✓ 1 fila insertada.

La Id de la fila insertada es: 6 (La consulta tardó 0,0005 segundos.)

INSERT INTO cuentas (titular, saldo) VALUES ('Lucia', 1000.00);

[Editar en línea] [Editar] [Crear código PHP]

✓ 1 fila insertada.

La Id de la fila insertada es: 7 (La consulta tardó 0,0018 segundos.)

INSERT INTO cuentas (titular, saldo) VALUES ('Lucia', 500.00);

[Editar en línea] [Editar] [Crear código PHP]

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0011 segundos.)

ROLLBACK;

[Editar en línea] [Editar] [Crear código PHP]

[Editar en línea] [Editar] [Editar] [Editar] [Editar] [Edit
```

10) Realiza una transferencia de 300.00 de la cuenta de Maria a la cuenta de Juan, pero simula un error en el proceso de transferencia después de retirar el dinero de la cuenta de Maria.

```
START TRANSACTION;
SELECT @sado:=saldo FROM cuentas WHERE titular="Maria";

UPDATE cuentas
SET saldo=CASE
WHEN @saldo>=300 AND titular="Maria"THEN saldo-300
WHEN @saldo>=300 AND titular="Juan"THEN saldo+300
ELSE saldo
END;

ROLLBACK;
```

