**Scenario 1: Update Last Modified Date on Customer Record Update**

sql

CopyEdit

CREATE OR REPLACE TRIGGER UpdateCustomerLastModified

BEFORE UPDATE ON Customers

FOR EACH ROW

BEGIN

:NEW.LastModified := SYSDATE;

END UpdateCustomerLastModified;

/

**Expected Output**:  
When a record in the Customers table is updated, the LastModified column is automatically set to the current date.

Example:

sql

CopyEdit

UPDATE Customers SET Name = 'John Doe' WHERE CustomerID = 1;

SELECT LastModified FROM Customers WHERE CustomerID = 1;

Output:

plaintext

CopyEdit

23-JUN-2025 12:34:56

**Scenario 2: Maintain an Audit Log for Transactions**

sql

CopyEdit

CREATE OR REPLACE TRIGGER LogTransaction

AFTER INSERT ON Transactions

FOR EACH ROW

BEGIN

INSERT INTO AuditLog (TransactionID, AccountID, Amount, TransactionDate)

VALUES (:NEW.TransactionID, :NEW.AccountID, :NEW.Amount, SYSDATE);

END LogTransaction;

/

**Expected Output**:  
Whenever a transaction is inserted into the Transactions table, a corresponding record is inserted into the AuditLog table.

Example:

sql

CopyEdit

INSERT INTO Transactions (TransactionID, AccountID, Amount, TransactionDate, TransactionType)

VALUES (1, 101, 500, SYSDATE, 'DEPOSIT');

SELECT \* FROM AuditLog WHERE TransactionID = 1;

Output:

plaintext

CopyEdit

TransactionID AccountID Amount TransactionDate

1 101 500 23-JUN-2025 12:34:56

**Scenario 3: Enforce Business Rules on Transactions**

sql

CopyEdit

CREATE OR REPLACE TRIGGER CheckTransactionRules

BEFORE INSERT ON Transactions

FOR EACH ROW

BEGIN

-- Ensure withdrawals do not exceed balance

IF :NEW.TransactionType = 'WITHDRAWAL' THEN

DECLARE

v\_balance NUMBER;

BEGIN

SELECT Balance INTO v\_balance

FROM Accounts

WHERE AccountID = :NEW.AccountID;

IF v\_balance < :NEW.Amount THEN

RAISE\_APPLICATION\_ERROR(-20001, 'Insufficient funds for withdrawal');

END IF;

END;

END IF;

-- Ensure deposits are positive

IF :NEW.TransactionType = 'DEPOSIT' AND :NEW.Amount <= 0 THEN

RAISE\_APPLICATION\_ERROR(-20002, 'Deposit amount must be positive');

END IF;

END CheckTransactionRules;

/

**Expected Output**:

* If a withdrawal exceeds the account balance, an error will be raised.
* If a deposit amount is non-positive, an error will be raised.

Example:

sql

CopyEdit

-- Attempt to withdraw more than balance

INSERT INTO Transactions (TransactionID, AccountID, Amount, TransactionDate, TransactionType)

VALUES (1, 101, 2000, SYSDATE, 'WITHDRAWAL');

Output:

plaintext

CopyEdit

ORA-20001: Insufficient funds for withdrawal