Exercise 5: Triggers

Scenario 1:

```
CREATE OR REPLACE TRIGGER UpdateCustomerLastModified
BEFORE UPDATE ON customers
FOR EACH ROW
BEGIN
   :NEW.LastModified := SYSDATE;
END UpdateCustomerLastModified;
/
```

Scenario 2:

```
CREATE OR REPLACE TRIGGER LogTransaction
AFTER INSERT ON transactions
FOR EACH ROW
BEGIN
   INSERT INTO AuditLog (log_id, transaction_id, log_date, log_message)
   VALUES (audit_log_seq.NEXTVAL, :NEW.transaction_id, SYSDATE, 'Transaction recorded');
END LogTransaction;
/
```

Scenario 3:

```
CREATE OR REPLACE TRIGGER CheckTransactionRules
BEFORE INSERT ON transactions
FOR EACH ROW
DECLARE
   v_balance NUMBER;
BEGIN
   -- Check if the amount is positive
   IF :NEW.amount <= 0 THEN
      RAISE_APPLICATION_ERROR(-20001, 'Transaction amount must be positive.');
   END IF;

   -- Check if the withdrawal does not exceed the balance
   IF :NEW.transaction_type = 'withdrawal' THEN
      SELECT balance INTO v_balance FROM accounts WHERE account_id = :NEW.account_id;
      IF v_balance < :NEW.amount THEN
```

```
         RAISE_APPLICATION_ERROR(-20002, 'Insufficient balance for withdrawal.');
      END IF;
   END IF;
END CheckTransactionRules;
/
```