```sql
-- ============================================================
-- TCS ILP PROJECT: Online Grocery Ordering System
-- MYSQL VERSION (Final)
-- ============================================================

-- 1. CLEANUP
-- MySQL allows disabling FK checks to drop tables easily
SET FOREIGN_KEY_CHECKS = 0;
DROP TABLE IF EXISTS admin_details;
DROP TABLE IF EXISTS transactions;
DROP TABLE IF EXISTS product;
DROP TABLE IF EXISTS login_details;
DROP TABLE IF EXISTS registration;
SET FOREIGN_KEY_CHECKS = 1;


-- ============================================================
-- US_SQL_002 : REGISTRATION TABLE
-- ============================================================
CREATE TABLE registration (
    customer_id     INT PRIMARY KEY,
    customer_name   VARCHAR(100) NOT NULL,
    email           VARCHAR(100) NOT NULL,
    password        VARCHAR(50)  NOT NULL,
    address         VARCHAR(200) NOT NULL,
    contact_number  BIGINT       NOT NULL
);

INSERT INTO registration VALUES
(101,'Sam','sam@gmail.com','Pass123','RZ/12-4, Mahavir Enclave, New Delhi,
India',9876543210),
(102,'John','john@yahoo.com','Pass456','123 Apple St, New York, US',8877665544),
(103,'Priya','priya@tcs.com','Pass789','Tech Park, Bangalore, India',9123456780),
(104,'Mike','mike@usmail.com','Pass000','456 Orange Ave, California, US',7766554433);


-- ============================================================
-- US_SQL_001 : LOGIN TABLE
-- ============================================================
CREATE TABLE login_details (
    customer_id       INT NOT NULL,
    last_login        DATETIME NOT NULL,
    last_logout       DATETIME NOT NULL,
    updated_password   VARCHAR(50) NOT NULL,
    old_password      VARCHAR(50) NOT NULL,
    is_now_logged_in   VARCHAR(1),
```

```
    CONSTRAINT chk_login CHECK (is_now_logged_in IN ('Y','N')),
    CONSTRAINT fk_login_cust FOREIGN KEY (customer_id) REFERENCES
registration(customer_id)
);

-- Note: MySQL uses NOW() instead of SYSTIMESTAMP
INSERT INTO login_details VALUES
(101, NOW(), NOW() - INTERVAL 1 DAY, 'Pass123', 'Old1', 'Y'),
(102, NOW(), NOW() - INTERVAL 1 DAY, 'Pass456', 'Old2', 'Y'),
(103, NOW(), NOW() - INTERVAL 1 DAY, 'Pass789', 'Old3', 'Y'),
(104, NOW(), NOW() - INTERVAL 1 DAY, 'Pass000', 'Old4', 'N');

-- ============================================================
-- US_SQL_003 : PRODUCT TABLE
-- ============================================================
CREATE TABLE product (
    product_id    INT PRIMARY KEY,
    product_name  VARCHAR(100) NOT NULL,
    price         DECIMAL(10,2) NOT NULL,
    quantity      INT NOT NULL,
    reserved      VARCHAR(1),
    customer_id   INT NOT NULL,
    CONSTRAINT chk_res CHECK (reserved IN ('Y','N')),
    CONSTRAINT fk_prod_cust FOREIGN KEY (customer_id) REFERENCES
registration(customer_id)
);

INSERT INTO product VALUES
(5001,'Milk',60,2,'Y',101),
(5002,'Bread',40,1,'Y',101),
(5003,'Butter',55,5,'Y',102),
(5004,'Rice',100,3,'Y',103),
(5005,'Wheat',200,1,'Y',104);

-- ============================================================
-- US_SQL_004 : TRANSACTION TABLE
-- ============================================================
CREATE TABLE transactions (
    transaction_id INT PRIMARY KEY,
    customer_id    INT NOT NULL,
    product_id     INT NOT NULL,
    total_amount   DECIMAL(10,2) NOT NULL,
    no_of_items    INT NOT NULL,
```

```sql
    CONSTRAINT fk_trx_cust FOREIGN KEY (customer_id) REFERENCES
registration(customer_id),
    CONSTRAINT fk_trx_prod FOREIGN KEY (product_id) REFERENCES product(product_id)
);

INSERT INTO transactions VALUES
(9001,101,5001,5000,5),
(9002,102,5003,3000,3),
(9003,103,5004,7000,7), -- Highest
(9004,101,5002,6000,4), -- Second Highest
(9005,104,5005,2000,1);


-- ============================================================
-- US_SQL_005 : ADMIN TABLE
-- ============================================================
CREATE TABLE admin_details (
    admin_id      INT PRIMARY KEY,
    admin_password VARCHAR(50) NOT NULL,
    customer_id   INT NOT NULL,
    product_id    INT NOT NULL,
    transaction_id INT NOT NULL,
    CONSTRAINT fk_adm_cust FOREIGN KEY (customer_id) REFERENCES
registration(customer_id),
    CONSTRAINT fk_adm_prod FOREIGN KEY (product_id) REFERENCES
product(product_id),
    CONSTRAINT fk_adm_trx FOREIGN KEY (transaction_id) REFERENCES
transactions(transaction_id)
);

INSERT INTO admin_details VALUES
(1,'Admin@123',101,5001,9001),
(2,'Admin@456',103,5004,9003);


-- ============================================================
-- REPORTS / QUERIES (MySQL Compatible)
-- ============================================================

-- US_SQL_001 : First 50% Logged-in Customers
-- Logic: Uses Window Function (Standard in MySQL 8.0)
SELECT customer_id
FROM (
    SELECT customer_id,
        ROW_NUMBER() OVER (ORDER BY last_login) as rn,
        COUNT(*) OVER () as total_count
```

```sql
    FROM login_details
    WHERE is_now_logged_in = 'Y'
) AS subquery
WHERE rn <= CEIL(total_count / 2);

-- US_SQL_002 : Update Customer Name Based on Country
-- Logic: MySQL uses CONCAT(), not ||
UPDATE registration
SET customer_name = CASE
    WHEN address LIKE '%US%' THEN CONCAT('US_', customer_name)
    WHEN address LIKE '%India%' THEN CONCAT('IN_', customer_name)
    ELSE customer_name
END;

-- Check Update
SELECT * FROM registration;

-- US_SQL_003 : Distinct Products
SELECT DISTINCT product_name FROM product;

-- US_SQL_003 : Customer–Product Mapping
SELECT r.customer_name, p.product_name
FROM registration r
JOIN product p ON r.customer_id = p.customer_id;

-- US_SQL_004 : Second Highest Transaction
-- Logic: Order Descending, Skip 1, Take 1
SELECT * FROM transactions
ORDER BY total_amount DESC
LIMIT 1 OFFSET 1;

-- US_SQL_005 : Final Admin Report
SELECT r.customer_id,
    r.customer_name,
    r.email,
    t.transaction_id,
    p.product_name
FROM registration r
JOIN transactions t ON r.customer_id = t.customer_id
JOIN product p ON t.product_id = p.product_id;
```