```sql
CREATE TABLE Resources (
    resource_id INT PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    quantity INT CHECK (quantity >= 0),
    unit VARCHAR(20),
    description TEXT,
    last_updated DATETIME
);

INSERT INTO Resources (resource_id, name, quantity, unit, description, last_updated)
VALUES
(1, 'Rice', 500, 'kg', 'Basmati, expiry: 2025', '2025-07-20 09:00:00'),
(2, 'Blankets', 200, 'pcs', 'Winter, double-layered', '2025-07-28 16:22:00'),
(3, 'Medicine', 150, 'boxes', 'First aid kits', '2025-07-27 10:12:00');

-- Update quantity of Medicine to 180
UPDATE Resources
SET quantity = 180
WHERE resource_id = 3;

-- Add a new column for supplier name
ALTER TABLE Resources
ADD supplier VARCHAR(100);

-- Add UNIQUE constraint to resource name
ALTER TABLE Resources
ADD CONSTRAINT unique_resource_name UNIQUE (name);

-- Add NOT NULL constraint to unit
ALTER TABLE Resources
MODIFY unit VARCHAR(20) NOT NULL;

-- Select all records
SELECT * FROM Resources;

-- Resources updated after July 25, 2025
SELECT * FROM Resources
WHERE last_updated > '2025-07-25';

-- Show only name and quantity of all resources
SELECT name, quantity FROM Resources;

-- List all resource names and units in one column
SELECT name AS Info FROM Resources
```

```sql
UNION
SELECT unit AS Info FROM Resources;

-- Find values that are both names and units (for demo)
SELECT name FROM Resources
INTERSECT
SELECT unit FROM Resources;

-- Show pairs of resources that share the same unit
SELECT A.name AS Resource1, B.name AS Resource2, A.unit
FROM Resources A
JOIN Resources B ON A.unit = B.unit AND A.resource_id <> B.resource_id;
```