Payrollmanagementsystem-capstone project (database)

-- Create the database if it doesn't exist

CREATE DATABASE IF NOT EXISTS imsdb CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;

-- Use the database

USE imsdb;

-- Table for Users

CREATE TABLE IF NOT EXISTS user (

   id BIGINT AUTO_INCREMENT PRIMARY KEY,

   username VARCHAR(255) NOT NULL UNIQUE,

   password VARCHAR(255) NOT NULL,

   role VARCHAR(50) NOT NULL DEFAULT 'SALES' -- ADMIN, MANAGER, SALES

);

-- Table for Suppliers

CREATE TABLE IF NOT EXISTS supplier (

   id BIGINT AUTO_INCREMENT PRIMARY KEY,

   name VARCHAR(255) NOT NULL,

   contact VARCHAR(255),

   address TEXT,

   created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,

   updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP

);

```sql
-- Table for Products
CREATE TABLE IF NOT EXISTS product (
    id BIGINT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(255) NOT NULL,
    description TEXT,
    price DECIMAL(10, 2) NOT NULL,
    stock_quantity INT NOT NULL DEFAULT 0,
    min_stock_level INT NOT NULL DEFAULT 5,
    supplier_id BIGINT,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
    FOREIGN KEY (supplier_id) REFERENCES supplier(id) ON DELETE SET NULL,
    INDEX idx_product_name (name),
    INDEX idx_product_supplier (supplier_id)
);

-- Table for Purchase Orders
CREATE TABLE IF NOT EXISTS purchase_order (
    id BIGINT AUTO_INCREMENT PRIMARY KEY,
    supplier_id BIGINT,
    order_date TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP,
    status ENUM('PENDING', 'APPROVED', 'REJECTED', 'RECEIVED', 'CANCELLED') NOT NULL DEFAULT 'PENDING',
    total_amount DECIMAL(10, 2) DEFAULT 0.00,
    created_by BIGINT,
    approved_by BIGINT,
```

```sql
    received_by BIGINT,

    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,

    updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE
CURRENT_TIMESTAMP,

    FOREIGN KEY (supplier_id) REFERENCES supplier(id) ON DELETE SET NULL,

    FOREIGN KEY (created_by) REFERENCES user(id) ON DELETE SET NULL,

    FOREIGN KEY (approved_by) REFERENCES user(id) ON DELETE SET NULL,

    FOREIGN KEY (received_by) REFERENCES user(id) ON DELETE SET NULL,

    INDEX idx_po_status (status),

    INDEX idx_po_supplier (supplier_id)
);


-- Table for Purchase Order Items
CREATE TABLE IF NOT EXISTS purchase_order_item (
    id BIGINT AUTO_INCREMENT PRIMARY KEY,

    purchase_order_id BIGINT NOT NULL,

    product_id BIGINT,

    quantity INT NOT NULL,

    unit_price DECIMAL(10, 2) NOT NULL,

    total_price DECIMAL(10, 2) NOT NULL,

    FOREIGN KEY (purchase_order_id) REFERENCES purchase_order(id) ON DELETE
CASCADE,

    FOREIGN KEY (product_id) REFERENCES product(id) ON DELETE SET NULL,

    INDEX idx_po_item_po (purchase_order_id),

    INDEX idx_po_item_product (product_id)
);


-- Table for Sales
```

```sql
CREATE TABLE IF NOT EXISTS sale (

    id BIGINT AUTO_INCREMENT PRIMARY KEY,

    sale_date TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP,

    total_amount DECIMAL(10, 2) NOT NULL,

    customer_name VARCHAR(255),

    customer_contact VARCHAR(255),

    created_by BIGINT,

    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,

    updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE
CURRENT_TIMESTAMP,

    FOREIGN KEY (created_by) REFERENCES user(id) ON DELETE SET NULL,

    INDEX idx_sale_date (sale_date)

);


-- Table for Sale Items
CREATE TABLE IF NOT EXISTS sale_item (

    id BIGINT AUTO_INCREMENT PRIMARY KEY,

    sale_id BIGINT NOT NULL,

    product_id BIGINT,

    quantity INT NOT NULL,

    price DECIMAL(10, 2) NOT NULL,

    total DECIMAL(10, 2) NOT NULL,

    FOREIGN KEY (sale_id) REFERENCES sale(id) ON DELETE CASCADE,

    FOREIGN KEY (product_id) REFERENCES product(id) ON DELETE SET NULL,

    INDEX idx_sale_item_sale (sale_id),

    INDEX idx_sale_item_product (product_id)

);
```

```sql
-- Table for inventory audit log
CREATE TABLE IF NOT EXISTS inventory_audit (
    id BIGINT AUTO_INCREMENT PRIMARY KEY,
    product_id BIGINT NOT NULL,
    old_stock INT NOT NULL,
    new_stock INT NOT NULL,
    change_type ENUM('SALE', 'RESTOCK', 'ADJUSTMENT', 'RETURN') NOT NULL,
    reference_id BIGINT, -- Can be sale_id or purchase_order_id
    notes TEXT,
    created_by BIGINT,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (product_id) REFERENCES product(id) ON DELETE CASCADE,
    FOREIGN KEY (created_by) REFERENCES user(id) ON DELETE SET NULL,
    INDEX idx_audit_product (product_id),
    INDEX idx_audit_date (created_at)
);


-- Insert default users
INSERT INTO user (username, password, role) VALUES
('admin', '$2a$10$rOzZJkcij/hp7k6vVHd.7.FYvW6qQaQ1nZ1Jz1Z1Z1Z1Z1Z1Z1Z1Z1Z1', 'ADMIN'), -- password: admin123
('manager', '$2a$10$rOzZJkcij/hp7k6vVHd.7.FYvW6qQaQ1nZ1Jz1Z1Z1Z1Z1Z1Z1Z1Z1Z1', 'MANAGER'), -- password: manager123
('sales', '$2a$10$rOzZJkcij/hp7k6vVHd.7.FYvW6qQaQ1nZ1Jz1Z1Z1Z1Z1Z1Z1Z1Z1Z1', 'SALES'); -- password: sales123


-- Insert sample suppliers
```

```sql
INSERT INTO supplier (name, contact, address) VALUES

('Tech Supplies Inc.', 'contact@techsupplies.com', '123 Tech Street, Tech City'),

('Office World Ltd.', 'info@officeworld.com', '456 Office Road, Business District'),

('Electro Gadgets Co.', 'sales@electrogadgets.com', '789 Electronics Avenue, Innovation Park');


-- Insert sample products

INSERT INTO product (name, description, price, stock_quantity, min_stock_level, supplier_id) VALUES

('Laptop Dell XPS 13', '13-inch laptop with Intel i7 processor, 16GB RAM, 512GB SSD', 1299.99, 15, 5, 1),

('Wireless Mouse', 'Ergonomic wireless mouse with 2.4GHz connectivity', 29.99, 45, 10, 2),

('Mechanical Keyboard', 'RGB mechanical keyboard with blue switches', 89.99, 22, 8, 1),

('27-inch Monitor', 'IPS 4K monitor with HDR support', 399.99, 8, 5, 3),

('Webcam HD 1080p', 'Full HD webcam with built-in microphone', 59.99, 30, 12, 2),

('USB-C Hub', '7-in-1 USB-C hub with HDMI, USB, and Ethernet ports', 49.99, 18, 8, 3);


-- Display confirmation message

SELECT 'Database and tables created successfully!' AS Status;


ALTER TABLE purchase_order_item MODIFY total_price DECIMAL(10, 2) NULL;


-- Update existing products with NULL stock_quantity to 0

UPDATE product SET stock_quantity = 0 WHERE stock_quantity IS NULL;


-- Add NOT NULL constraint if not already there

ALTER TABLE product MODIFY stock_quantity INT NOT NULL DEFAULT 0;


-- Default admin user
```

```sql
INSERT INTO users (username, email, password, role)

VALUES ('admin', 'admin@payroll.com',
'$2a$10$rOzZJXW6jG.7kKjQ7qZzE.BVTDn6Ih6kY8Qk8jqJZJZJZJZJZJZJZJZ', 'ADMIN');


-- Default departments

INSERT INTO departments (department_name, description) VALUES

('HR', 'Human Resources Department'),

('IT', 'Information Technology Department'),

('Finance', 'Finance and Accounting Department'),

('Operations', 'Operations Department');


-- Default job roles

INSERT INTO job_roles (job_title, base_salary, description) VALUES

('HR Manager', 75000.00, 'Human Resources Manager'),

('Software Developer', 65000.00, 'Software Developer'),

('Accountant', 55000.00, 'Finance Accountant'),

('Operations Manager', 70000.00, 'Operations Department Manager');


-- Default admin employee (links to user_id = 1)

INSERT INTO employees (user_id, first_name, last_name, date_of_birth, phone_number,
address, hire_date, department_id, job_id)

VALUES (1, 'Admin', 'User', '1990-01-01', '1234567890', 'Admin Address', CURDATE(), 1, 1);


USE payroll_management_system;


ALTER TABLE employees DROP FOREIGN KEY fk_employee_user;

ALTER TABLE employees DROP FOREIGN KEY fk_employee_department;
```

ALTER TABLE employees DROP FOREIGN KEY fk_employee_job;

ALTER TABLE leave_requests DROP FOREIGN KEY fk_leave_employee;

ALTER TABLE leave_requests DROP FOREIGN KEY fk_leave_processed;

ALTER TABLE payroll DROP FOREIGN KEY fk_payroll_employee;

ALTER TABLE salary_history DROP FOREIGN KEY fk_salary_employee;

ALTER TABLE salary_history DROP FOREIGN KEY fk_salary_changedby;

user-- Default admin user

INSERT INTO users (username, email, password, role)

VALUES ('admin', 'admin@payroll.com',
'$2a$10$rOzZJXW6jG.7kKjQ7qZzE.BVTDn6Ih6kY8Qk8jqJZJZJZJZJZJZJZJZ', 'ADMIN');

-- Default departments

INSERT INTO departments (department_name, description) VALUES

('HR', 'Human Resources Department'),

('IT', 'Information Technology Department'),

('Finance', 'Finance and Accounting Department'),

('Operations', 'Operations Department');

-- Default job roles

INSERT INTO job_roles (job_title, base_salary, description) VALUES

('HR Manager', 75000.00, 'Human Resources Manager'),

('Software Developer', 65000.00, 'Software Developer'),

('Accountant', 55000.00, 'Finance Accountant'),

('Operations Manager', 70000.00, 'Operations Department Manager');

-- Default admin employee (links to user_id = 1)

INSERT INTO employees (user_id, first_name, last_name, date_of_birth, phone_number, address, hire_date, department_id, job_id)

VALUES (1, 'Admin', 'User', '1990-01-01', '1234567890', 'Admin Address', CURDATE(), 1, 1);

USE payroll_management_system;

ALTER TABLE employees DROP FOREIGN KEY fk_employee_user;

ALTER TABLE employees DROP FOREIGN KEY fk_employee_department;

ALTER TABLE employees DROP FOREIGN KEY fk_employee_job;

ALTER TABLE leave_requests DROP FOREIGN KEY fk_leave_employee;

ALTER TABLE leave_requests DROP FOREIGN KEY fk_leave_processed;

ALTER TABLE payroll DROP FOREIGN KEY fk_payroll_employee;

ALTER TABLE salary_history DROP FOREIGN KEY fk_salary_employee;

ALTER TABLE salary_history DROP FOREIGN KEY fk_salary_changedby;