|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **《数据库系统原理》实验报告1** | | | | | |
| **题目：**DDL语言实验 | | | | | |
| 学号 |  | 姓名 |  | 日期 | 2025年4月11日 |
| **实验环境：5.7.25-OceanBase\_CE-v4.2.1.10** | | | | | |
| **实验步骤及结果截图：**   1. **表结构设计**   CREATE TABLE Customers (      CustomerID INT AUTO\_INCREMENT PRIMARY KEY NOT NULL,      Name VARCHAR(50) NOT NULL,      Email VARCHAR(100) NOT NULL UNIQUE CHECK (Email REGEXP '^[A-Za-z0-9.\_%+-]+@[A-Za-z0-9.-]+\.[A-Za-z]{2,4}$'),      Phone VARCHAR(20) CHECK (Phone REGEXP '^((13[0-9]|14[01456879]|15[0-35-9]|16[2567]|17[0-8]|18[0-9]|19[0-35-9])[0-9]{8})|((0[0-9]{2,3})-?([0-9]{7,8}))$'),      RegistrationDate DATE NOT NULL  );  CREATE TABLE Categories (      CategoryID INT AUTO\_INCREMENT PRIMARY KEY NOT NULL,      CategoryName VARCHAR(50) NOT NULL UNIQUE  );  CREATE TABLE Products (      ProductID INT AUTO\_INCREMENT PRIMARY KEY NOT NULL,      ProductName VARCHAR(100) NOT NULL,      Price DECIMAL(10,2) NOT NULL CHECK (Price > 0),      StockQuantity INT NOT NULL CHECK (StockQuantity >= 0),      CategoryID INT NOT NULL,      FOREIGN KEY (CategoryID) REFERENCES Categories(CategoryID)  );  CREATE TABLE Orders (      OrderID INT AUTO\_INCREMENT PRIMARY KEY NOT NULL,      CustomerID INT NOT NULL,      OrderDate DATETIME NOT NULL DEFAULT CURRENT\_TIMESTAMP,      TotalAmount DECIMAL(10,2) NOT NULL CHECK (TotalAmount > 0),      FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID)  );  CREATE TABLE OrderItems (      OrderItemID INT AUTO\_INCREMENT PRIMARY KEY NOT NULL,      OrderID INT NOT NULL,      ProductID INT NOT NULL,      Quantity INT NOT NULL CHECK (Quantity > 0),      Subtotal DECIMAL(10,2) NOT NULL CHECK (Subtotal > 0),      FOREIGN KEY (OrderID) REFERENCES Orders(OrderID),      FOREIGN KEY (ProductID) REFERENCES Products(ProductID)  );  -- 计算Subtotal的触发器  DELIMITER $$  CREATE TRIGGER CalculateSubtotal  BEFORE INSERT ON OrderItems  FOR EACH ROW  BEGIN      DECLARE product\_price DECIMAL(10,2);      SELECT Price INTO product\_price FROM Products WHERE ProductID = NEW.ProductID;      SET NEW.Subtotal = product\_price \* NEW.Quantity;  END$$  DELIMITER ;    **2.表结构查询**  DESC Customers;  DESC Categories;  DESC Products;  DESC Orders;  DESC OrderItems;    **3.使⽤以下语句查询建⽴的约束:**  SELECT table\_name, constraint\_name, constraint\_type  FROM information\_schema.TABLE\_CONSTRAINTS  WHERE table\_name='Customers';  SELECT table\_name, constraint\_name, constraint\_type  FROM information\_schema.TABLE\_CONSTRAINTS  WHERE table\_name='Categories';  SELECT table\_name, constraint\_name, constraint\_type  FROM information\_schema.TABLE\_CONSTRAINTS  WHERE table\_name='Products';  SELECT table\_name, constraint\_name, constraint\_type  FROM information\_schema.TABLE\_CONSTRAINTS  WHERE table\_name='Orders';  SELECT table\_name, constraint\_name, constraint\_type  FROM information\_schema.TABLE\_CONSTRAINTS  WHERE table\_name='OrderItems';    **4.表结构修改与操作.**  **(1)**  ALTER TABLE Customers ADD COLUMN LastLoginDate DATETIME;  DELIMITER $$  CREATE PROCEDURE CustomerLogin(IN customer\_id INT)  BEGIN      UPDATE Customers      SET LastLoginDate = CURRENT\_TIMESTAMP      WHERE CustomerID = customer\_id;  END$$  DELIMITER ;  只需调用CALL CustomerLogin(客户ID);即可更新对应客户的登录时间。    **(2)**  ALTER TABLE Customers MODIFY COLUMN Name CHAR(15) NOT NULL;    **(3)**  ALTER TABLE Products ADD COLUMN IsFeatured BOOLEAN DEFAULT FALSE;  CREATE INDEX idx\_featured\_products ON Products(IsFeatured);    **(4)**  DELIMITER $$  CREATE PROCEDURE DeleteUnusedCategories()  BEGIN      DELETE FROM Categories      WHERE CategoryID NOT IN (          SELECT DISTINCT CategoryID FROM Products      );        SELECT CONCAT('Deleted ', ROW\_COUNT(), ' unused categories.') AS Result;  END$$  DELIMITER ;  -- 调用存储过程  CALL DeleteUnusedCategories();  CALL DeleteUnusedCategories();即可进行删除 | | | | | |
| **出现的问题：**  **1. DEFAULT CURRENT\_DATE() 语法错误**  在创建 Customers 表时，使用 DEFAULT CURRENT\_DATE() 设置默认值时出现语法错误：    **2. 正则表达式兼容性问题**  OceanBase 数据库对正则表达式的支持可能与标准 MySQL 有所不同，导致 Email 和 Phone 字段的 CHECK 约束创建困难。  **3. 触发器语法问题**  OceanBase 对触发器的语法要求与标准 MySQL 略有不同，特别是在 DELIMITER 的使用和触发器函数体的定义上。  **4. 外键约束关系处理**  表创建顺序问题：因为存在外键依赖关系，需要严格按照正确的顺序创建表（先创建被引用的表）。 | | | | | |
| **解决方案：**   1. DEFAULT CURRENT\_DATE() 语法错误   使用触发器代替默认值  -- 移除默认值语法  RegistrationDate DATE NOT NULL  -- 使用触发器设置默认值  DELIMITER $$  CREATE TRIGGER set\_registration\_date  BEFORE INSERT ON Customers  FOR EACH ROW  BEGIN      IF NEW.RegistrationDate IS NULL THEN          SET NEW.RegistrationDate = CURRENT\_DATE;      END IF;  END$$  DELIMITER ;   1. 正则表达式兼容性问题   解决方法：简化正则表达式  对于复杂验证，可以考虑创建一个函数来验证数据有效性，而不是直接使用 CHECK 约束   1. 触发器语法问题   解决方法：确保 DELIMITER 语句正确使用，特别是在 OceanBase 客户端中执行时  DELIMITER $$  CREATE TRIGGER [触发器名]  [触发时机] [触发事件] ON [表名]  FOR EACH ROW  BEGIN      -- 触发器逻辑  END$$  DELIMITER ;   1. 外键约束关系处理   解决方法：按照正确的顺序创建表和约束  Categories → Customers → Products → Orders → OrderItems  确保在创建引用表之前，被引用的表已经创建完成 | | | | | |