| 2차시 |  |  |  | 실습: □ | 과제 : | 평가 : □ |
| --- | --- | --- | --- | --- | --- | --- |
| 과제)    쇼핑몰을 운영하는 데 필요한 기본적인 관계형 데이터베이스 스키마를 설계할 때, 여러 핵심 테이블을 아래 내용을 참고하여 테이블 스키마를 설계하여 보세요.  필드는 자유롭게 구성하세요.  create database weekend\_project;  1. Customers (고객 정보)  customer\_id: 고객의 고유 식별자 (기본 키)  CREATE TABLE Customers  ( customer\_id INT PRIMARY KEY AUTO\_INCREMENT  )default character set utf8;  2. Products (제품 정보)  product\_id: 제품의 고유 식별자 (기본 키)  CREATE TABLE Products  ( product\_id INT PRIMARY KEY AUTO\_INCREMENT,  category\_id INT,  FOREIGN KEY (category\_id) REFERENCES Categories(category\_id)  )default character set utf8;  3. Categories (카테고리 정보)  category\_id: 카테고리의 고유 식별자 (기본 키)  CREATE TABLE Categories  ( category\_id INT PRIMARY KEY AUTO\_INCREMENT  )default character set utf8;  4. Orders (주문 정보)  order\_id: 주문의 고유 식별자 (기본 키)  CREATE TABLE Orders  (order\_id INT PRIMARY KEY AUTO\_INCREMENT,  customer\_id INT,  FOREIGN KEY (customer\_id) REFERENCES Customers(customer\_id)  )default character set utf8;  5. OrderDetails (주문 상세 정보)  order\_detail\_id: 주문 상세 정보의 고유 식별자 (기본 키)  CREATE TABLE OrederDetails  ( order\_detail\_id INT PRIMARY KEY AUTO\_INCREMENT,  order\_id INT,  product\_id INT,  FOREIGN KEY (order\_id) REFERENCES Orders(order\_id),  FOREIGN KEY (product\_id) REFERENCES Products(product\_id)  )default character set utf8;  관계구조:  l Customers와 Orders: 하나의 고객은 여러 주문을 할 수 있습니다 (1관계).  l Orders와 OrderDetails: 하나의 주문은 여러 개의 주문 상세 정보를 가질 수 있습니다 (1관계).  l Products와 OrderDetails: 하나의 제품은 여러 주문 상세 정보에 포함될 수 있습니다 (1관계).  l Categories와 Products: 하나의 카테고리는 여러 제품을 포함할 수 있습니다 (1관계). | | | | | | |