과제 내용

➢ 상품을 주문하는 주문 시스템을 구축하려 한다.

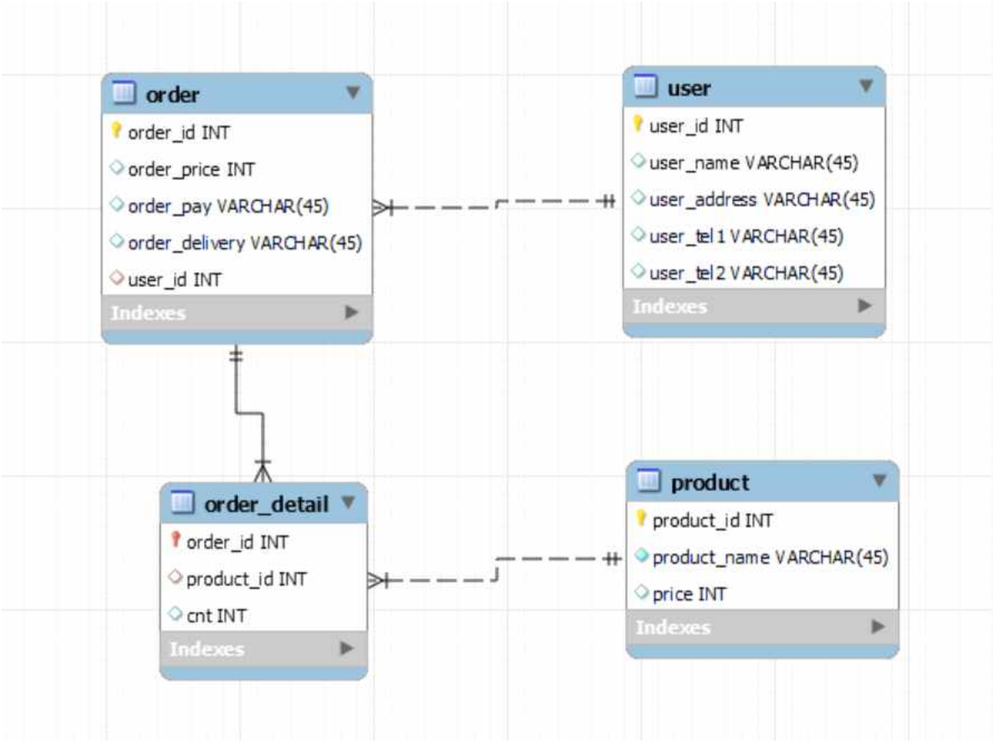
한 번 주문 시 여러 상품을 한꺼번에 주문하여 결제가능하다.

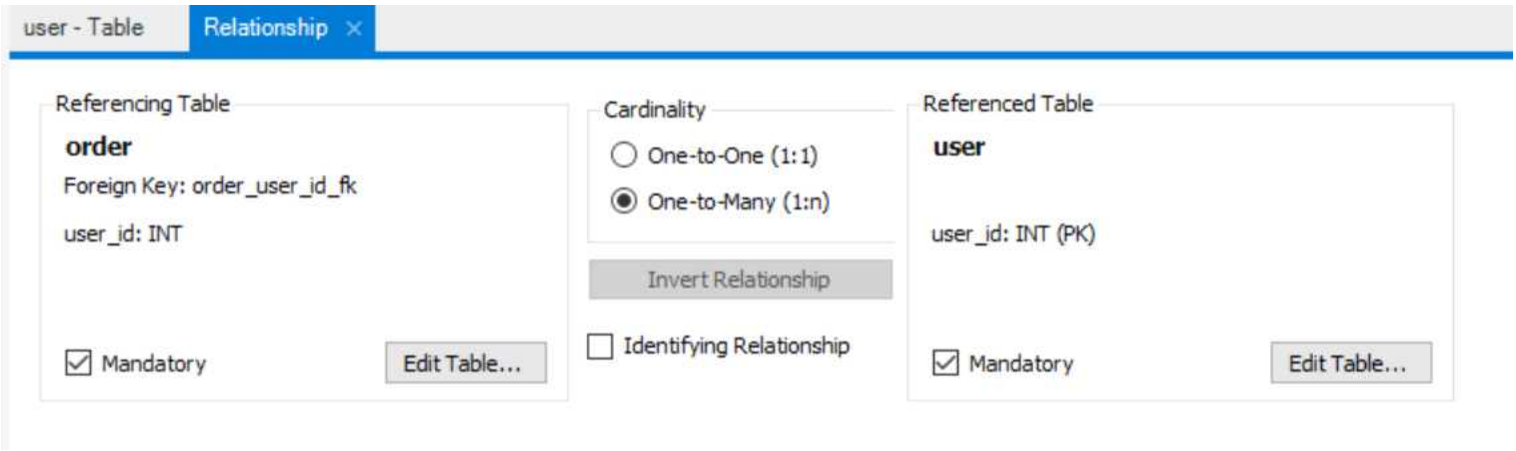
아래와 같은 정보를 저장하고 관리하고자 할 때 정규화를 적용하여 테이블 설계를 구현하세요.

필요하다면 제시된 정보 이외에 필요한 데이터항목을 자유롭게 추가하여 설계한다.

주문번호, 주문금액, 결제여부, 배송여부, 상품코드, 상품명, 가격, 수량, 고객번호, 고객명, 고객주소, 고객 연락처 1, 고객연락처 2

```
-- MySQL Workbench Forward Engineering
SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0;
SET @OLD_SQL_MODE=@@SQL_MODE,
SQL_MODE='ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR_FOR_DIVISION
_BY_ZERO,NO_ENGINE_SUBSTITUTION';
-- -----------------------------------------------------
-- Schema OrderSystem
-- -----------------------------------------------------
-- -----------------------------------------------------
-- Schema OrderSystem
-- -----------------------------------------------------
CREATE SCHEMA IF NOT EXISTS `OrderSystem` DEFAULT CHARACTER SET utf8 ;
USE `OrderSystem` ;
-- -----------------------------------------------------
-- Table `OrderSystem`.`user`
-- -----------------------------------------------------
DROP TABLE IF EXISTS `OrderSystem`.`user` ;
CREATE TABLE IF NOT EXISTS `OrderSystem`.`user` (
  `user_id` INT NOT NULL,
  `user_name` VARCHAR(45) NULL,
  `user_address` VARCHAR(45) NULL,
  `user_tel1` VARCHAR(45) NULL,
  `user_tel2` VARCHAR(45) NULL,
  PRIMARY KEY (`user_id`))
ENGINE = InnoDB;
-- -----------------------------------------------------
-- Table `OrderSystem`.`order`
-- -----------------------------------------------------
DROP TABLE IF EXISTS `OrderSystem`.`order` ;
CREATE TABLE IF NOT EXISTS `OrderSystem`.`order` (
  `order_id` INT NOT NULL AUTO_INCREMENT,
  `order_price` INT NULL,
  `order_pay` VARCHAR(45) NULL,
  `order_delivery` VARCHAR(45) NULL DEFAULT 'N',
  `user_id` INT NULL,
  PRIMARY KEY (`order_id`),
  INDEX `order_user_id_fk_idx` (`user_id` ASC) VISIBLE,
  CONSTRAINT `order_user_id_fk`
    FOREIGN KEY (`user_id`)
    REFERENCES `OrderSystem`.`user` (`user_id`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;
-- -----------------------------------------------------
-- Table `OrderSystem`.`product`
-- -----------------------------------------------------
DROP TABLE IF EXISTS `OrderSystem`.`product` ;
CREATE TABLE IF NOT EXISTS `OrderSystem`.`product` (
  `product_id` INT NOT NULL,
  `product_name` VARCHAR(45) NOT NULL,
  `price` INT NULL,
  PRIMARY KEY (`product_id`))
ENGINE = InnoDB;
-- -----------------------------------------------------
-- Table `OrderSystem`.`order_detail`
-- -----------------------------------------------------
DROP TABLE IF EXISTS `OrderSystem`.`order_detail` ;
CREATE TABLE IF NOT EXISTS `OrderSystem`.`order_detail` (
  `order_id` INT NOT NULL,
  `product_id` INT NULL,
  `cnt` INT NULL,
  PRIMARY KEY (`order_id`),
  INDEX `order_detail_product_id_fk_idx` (`product_id` ASC) VISIBLE,
  CONSTRAINT `order_detail_product_id_fk`
    FOREIGN KEY (`product_id`)
    REFERENCES `OrderSystem`.`product` (`product_id`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `order_detail_order_id_fk`
    FOREIGN KEY (`order_id`)
    REFERENCES `OrderSystem`.`order` (`order_id`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;
SET SQL_MODE=@OLD_SQL_MODE;
SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;
SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;
```





Referencing Table

order

Foreign Key: order_user_id_fk

user_id: INT

☑ Mandatory

Cardinality

○ One-to-One (1:1)

● One-to-Many (1:n)

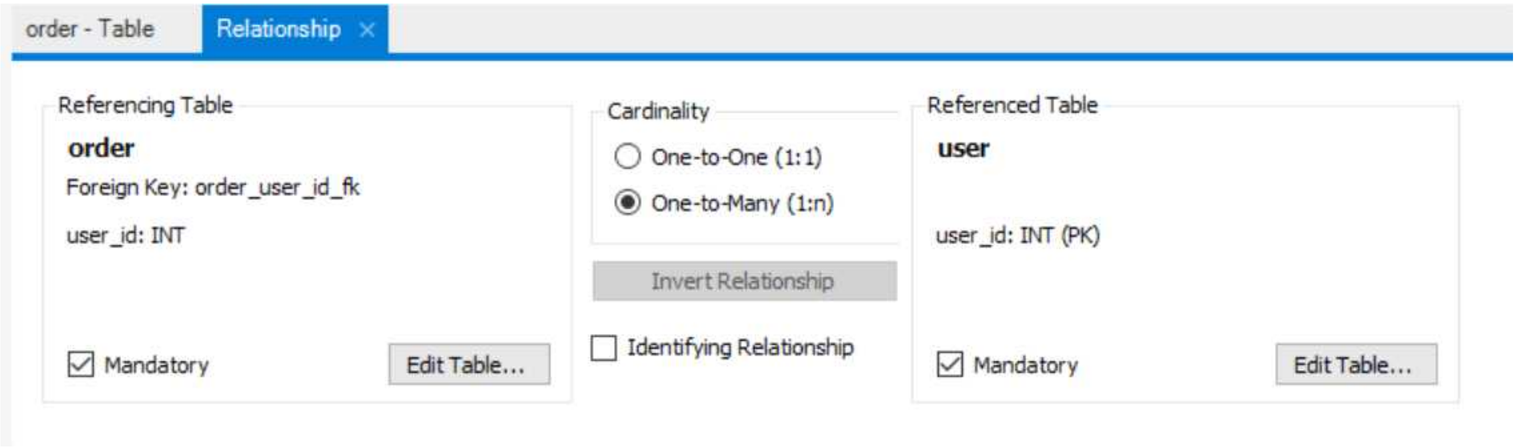
□ Identifying Relationship

Referenced Table

user

user_id: INT (PK)

☑ Mandatory



Referencing Table

order

Foreign Key: order_user_id_fk

user_id: INT

☑ Mandatory

Cardinality

○ One-to-One (1:1)

● One-to-Many (1:n)

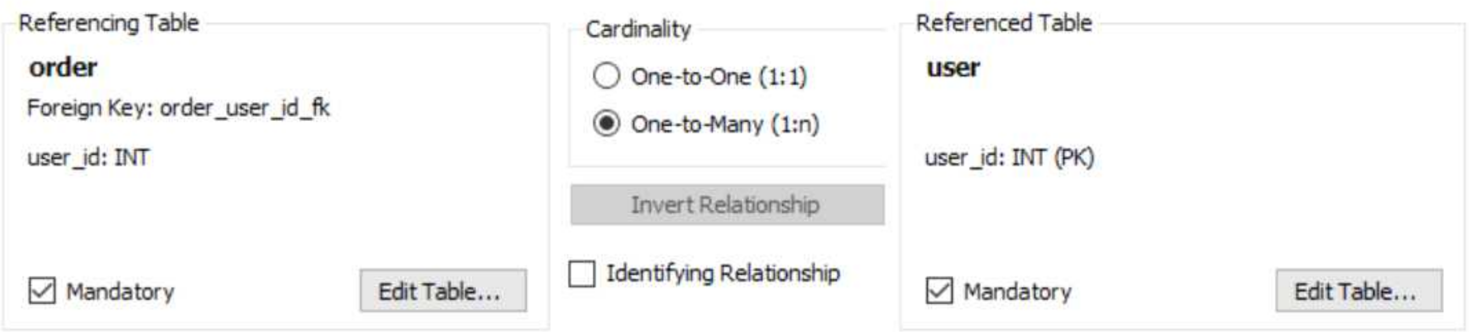
□ Identifying Relationship

Referenced Table

user

user_id: INT (PK)

☑ Mandatory


Referencing Table

order

Foreign Key: order_user_id_fk

user_id: INT

☑ Mandatory

Cardinality

○ One-to-One (1:1)

● One-to-Many (1:n)

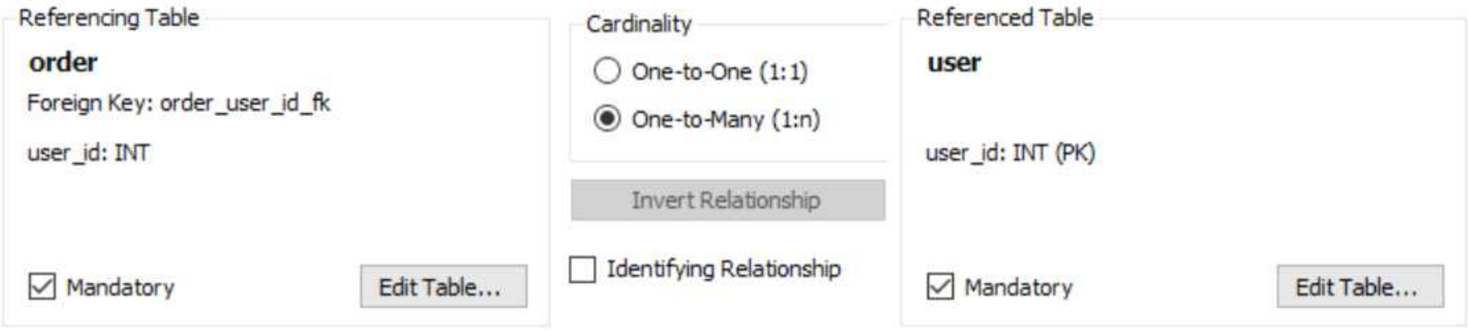
□ Identifying Relationship

Referenced Table

user

user_id: INT (PK)

☑ Mandatory



Referencing Table

order

Foreign Key: order_user_id_fk

user_id: INT

☑ Mandatory

Cardinality

○ One-to-One (1:1)

● One-to-Many (1:n)

□ Identifying Relationship

Referenced Table

user

user_id: INT (PK)

☑ Mandatory