

授業内容（講義と演習）

1. スキーマ設計例その5：実装用のスキーマ
2. 演習：スキーマの設計、関係（表）の作成と ERD の描画

今までの概念設計と論理設計を経て、正規化した wineShop スキーマを物理設計段階で実装に相応しいものに調整していく必要がある。

この授業に使われた HI 科の linux サーバーやデータベースサーバーはローカルサーバで、今までの案は実装できる。ただし、昨今、クラウドを利用し仮想サーバー上でのデータベースの構築と利用は多くなりました。これにより、Linux または WEB サーバーはデータベースサーバーと別々に設置され、オンラインにおいて所定の phpMyAdmin や MySQL workbench など使って、データベースの定義と操作を行われなければならない。

phpMyAdmin や MySQL workbench によるデータベースやテーブルの作成には特定のルールがある。それに従わないと、データベースの実装はできない。その中の一つは、すべてのテーブルには必ず主キーがあるとのこと。今の wineShop の中にある、setDetail, orderWine, orderSetなどを、ローカルまたはクラウドに対応できるように調整してみる。ここでは、顧客管理の案③を採用する

1. スキーマ

A) 顧客管理(修正なし)

idpw(メールアドレス, パスワード, パスワード更新日時)

顧客(顧客 ID, firstname, lastname, メールアドレス, 電話番号, 郵便番号, 住所, 状態, 登録日時)

idpw(email, password, pwUpdateTime)

customer(customerID, firstName, lastName, email, phone, postcode, address, status, regDateTime)

B) ワインとワインセット管理（ワインセット詳細が修正あり）

生産地 (産地 ID, 産地名, 国名)

ワイン状態 (状態 ID, 状態, 対象)

ワイン (ワイン ID, 名前, 産地 ID, 品種, 色, ビンテージ, 価格, 画像ファイル名, コメント, 状態 ID)

ワインセット (セット ID, 名前, コメント, 状態 ID)

ワインセット詳細 (tableid, セット ID, ワイン ID, ワインの本数) (修正あり)

locality(locaID, locality, country)

wstatus(stID, status, forwhat)

wine(wineID, name, locaID, kind, color, vintage, price, picture, comment, stID)

wineSet(setID, name, comment, stID)

setDetail(id, setID, wineID, quantity) 修正あり

C) 顧客注文管理

cusOrder(注文 ID, 客 ID, 注文日時, 配達日, 配達先郵便番号, 配達先住所, 受取人名前, 状態)

orderWine(tableid, 注文 ID, ワイン ID, 数量) 修正あり

orderWineSet(tableid, 注文 ID, ワインセット ID, 数量) 修正あり

cusOrder(orderID, customerID, orderTime, deliveryDay, deliveryPostcode, deliveryAddr, receiverName, status)

orderWine(id, ordID, wID, qty)

orderWineSet(id, ordID, winesetID, qty)

D) 顧客請求管理 (修正なし)

bill(請求 ID, 顧客 ID, 注文 ID, 請求日, 支払期限日, 総額, 状態)

bill(billID, cID, ordID, reqDay, dealine, payAmount, status)

2. MySQL によるテーブルの作成

```
create table locality(
    locaID int primary key not null auto_increment,
    locality varchar(100),
    country varchar(30));
```

```
create table wstatus(
    stID int primary key not null auto_increment,
    status varchar(20),
    forwhat varchar(20));
```

```
create table wine (
```

```
wineID int auto_increment not null primary key,  
name varchar(100),  
locaID int,  
kind varchar(32),  
color varchar(8),  
vintage int,  
price int,  
picture varchar(150),  
comment text,  
stID int,  
foreign key(locaID) references locality(locaID),  
foreign key(stID) references wstatus(stID));  
  
create table wineSet(  
    setID varchar(20) not null primary key,  
    name varchar(32),  
    comment text,  
    stID int,  
    foreign key(stID) references wstatus(stID));  
  
create table setDetail(  
    id int primary key auto_increment,  
    setID varchar(20),  
    wineID int,  
    quantity int,  
    foreign key(setID) references wineSet(setID),  
    foreign key(wineID) references wine(wineID));  
  
create table idpw(  
    email varchar(150) primary key not NULL,  
    password varchar(20),  
    updateTime datetime);  
  
create table customer(  
    customerID varchar(10) primary key not NULL,  
    firstName varchar(30),
```

```
        lastName varchar(16),
        email varchar(150),
        phone varchar(20),
        postcode char(8),
        address varchar(300),
        status char(10),
        regDateTime datetime,
        foreign key (email) references idpw(email));
create table cusOrder(
    orderID int primary key auto_increment,
    customerID varchar(10),
    orderTime datetime,
    deliveryDay date,
    deliveryPostcode char(8),
    deliveryAddr varchar(300),
    receiverName varchar(60),
    status char(10),
    foreign key (customerID) references customer(customerID));

create table orderWine(
    id int primary key auto_increment,
    ordID int,
    wID int,
    qty int,
    foreign key (ordID) references cusOrder(orderID),
    foreign key (wID) references wine(wineID));

create table orderWineSet(
    id int primary key auto_increment,
    ordID int,
    setID varchar(20),
    qty int,
    foreign key (ordID) references cusOrder(orderID),
    foreign key (setID) references wineSet(setID));

create table bill(
```

