# **Physical Data Model**

## **Customer**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên cột** | **Kiểu dữ liệu** | **Khoá** | **Diễn giải** |
| 1 | customer\_id | Varchar(20) | PK | Mã khách hàng |
| 2 | name | Varchar(100) |  | Tên khách hàng |
| 3 | phone\_number | Varchar(12) |  | Số điện thoại |
| 4 | address | Varchar(255) |  | Địa chỉ |
| 5 | email | Varchar(100) |  | email |

## **Bill**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên cột** | **Kiểu dữ liệu** | **Khoá** | **Diễn giải** |
| 1 | bill\_id | Varchar(20) | PK | Mã hoá đơn |
| 2 | customer\_id | Varchar(20) | FK | Mã khách hàng ( có thể null) |
| 4 | CreateDate | Datetime |  | Ngày lập |
| 5 | UpdateDate | Datetime |  | Cập nhật lần cuối |
| 6 | Discount | Decimal(15,2) |  | Tổng giá trị chiết khấu. |
|  | Total | Int(10) |  | Tổng giá trị bill phải trả sau khi tính qua tất cả các khoảng. |
| 7 | Status | Varchar(100) |  | Tình trạng hoá đơn.  Khoảng giá trị: **“pending, success, cancel”** |

## **BillItem**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên cột** | **Kiểu dữ liệu** | **Khoá** | **Diễn giải** |
| 1 | BillId | Varchar(20) | PK,FK | Mã hoá đơn |
| 2 | ShopSku | Varchar(50) | PK,FK | Mã stock keeping unit do hệ thống laz tự động đặt. Khóa ShopSku là duy nhất |
|  | SellerSku | Varchar(50) |  | Mã stock keeping unit do người dùng đặt hoặc hệ thống tự tạo khi người dùng không đặt. Mã SellerSku là duy nhất |
| 3 | Number | Int(10) |  | Số lượng |
| 4 | Price | Decimal(15,2) |  | Giá tiền sản phẩm |
| 5 | PaidPrice | Decimal(15,2) |  | Số tiền phải trả, đã qua chiết khấu |

## **Order**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên cột** | **Kiểu dữ liệu** | **Khoá** | **Diễn giải** |
| 1 | OrderID | Varchar(20) | PK | Mã đơn hàng. Lấy từ lazop |
| 4 | CreateDate | Datetime |  | Ngày lập. |
| 5 | UpdateDate | Varchar(100) |  | Cập nhật lần cuối |
| 6 | PaymentMethod | Varchar(100) |  | Phương thức thanh toán. |
| 7 | ShippingFee | Decimal(15,2) |  | Tổng phí ship của đơn hàng còn phải trả. |
| 8 | Price | Int(10) |  | Tổng giá trị đơn hàng |
| 9 | Status | Varchar(20) |  | Trạng thái order. Khoảng giá trị **“unpaid, pending, canceled, ready\_to\_ship, delivered, returned, shipped, failed”** |

## **OrderItem**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên cột** | **Kiểu dữ liệu** | **Khoá** | **Diễn giải** |
| 1 | OrderID | Varchar(20) | PK,FK | Mã đơn hàng |
| 2 | ShopSku | Varchar(50) | PK,FK | Mã stock keeping unit do hệ thống laz tự động đặt. Khóa ShopSku là duy nhất |
|  | SellerSku | Varchar(50) |  | Mã stock keeping unit do người dùng đặt hoặc hệ thống tự tạo khi người dùng không đặt. Mã SellerSku là duy nhất |
|  | Name | Varchar(100) |  | Tên sản phẩm. Tránh việc sản phẩm cùng id đó update tên sau này. |
|  | Variation | Varchar(255) |  | Thể hiện tên của item. Dữ liệu mẫu: “Nhóm màu: màu hồng” |
|  | ItemPrice | Decimal(15,2) |  | Giá sản phẩm chưa qua xử lý. |
|  | PaidPrice | Decimal(15,2) |  | Đơn giá. Giá tiền người mua phải trả sau khi đi qua tất cả giảm giá. |
|  | Quantity | Int(10) |  | Số lượng. |

## **GoodsReceivedNote**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên cột** | **Kiểu dữ liệu** | **Khoá** | **Diễn giải** |
| 1 | GRNId | Varchar(20) | PK | Mã nhập hàng |
|  | Description | Text |  | Chi tiết, nội dung, lý do của đơn nhập hàng. |
| 2 | DateReceived | Datetime |  | Ngày nhận hàng |
| 3 | Provider | Varchar(50) |  | Tên đơn vị cung cấp. |
|  | Telephone | Varchar(12) |  | Số điện thoại |
|  | Address | Varchar(255) |  | Địa chỉ nơi cung cấp |
|  | Status | Varchar(20) |  | Trạng thái phếu nhập.  Khoảng giá trị: **“pending, success, canceled”** |

## **GoodsReceivedNoteItem**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên cột** | **Kiểu dữ liệu** | **Khoá** | **Diễn giải** |
|  | GRNId | Varchar(20) | PK,FK | Mã nhập hàng |
|  | ShopSku | Varchar(50) | PK,FK | Mã stock keeping unit do hệ thống laz tự động đặt. Khóa ShopSku là duy nhất |
|  | SellerSku | Varchar(50) |  | Mã stock keeping unit do người dùng đặt hoặc hệ thống tự tạo khi người dùng không đặt. Mã SellerSku là duy nhất |
|  | Name | Varchar(255) |  | Tên sản phẩm |
|  | Color | Varchar(255) |  | Màu |
|  | Size | Varchar(20) |  | Size giày. Mẫu dữ liệu: EU:39 |
|  | Quantity | Int(10) |  | Số lượng |
|  | Price | Decimal(15,2) |  | Giá nhập của sản phẩm |

## **Category**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên cột** | **Kiểu dữ liệu** | **Khoá** | **Diễn giải** |
| 1 | CategoryId | Varchar(20) | PK | Mã loại. Tương đương primary\_category trong lazop. |
| 2 | CategoryName | Varchar(100) |  | Tên loại giày |

## **Product**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên cột** | **Kiểu dữ liệu** | **Khoá** | **Diễn giải** |
|  | ProductId | Varchar(20) | PK | Mã giày |
|  | CategoryId | Varchar(20) | FK | Mã loại |
|  | Brand | Varchar(50) |  | Tên nhãn hiệu. Mặc định là “No Brand” |
|  | ShoesName | Varchar(255) |  | Tên giày |
|  | ShortDescription | Text |  | Mô tả ngắn về sản phẩm |
|  | Description | Text |  | Mô tả sản phẩm |
|  | Status | Varchar(20) |  | Khoảng giá trị:  **“active, inactive, deleted**  **”** |

## **SKU**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên cột** | **Kiểu dữ liệu** | **Khoá** | **Diễn giải** |
|  | ShopSku | Varchar(50) | PK | Mã stock keeping unit do hệ thống laz tự động đặt. Khóa ShopSku là duy nhất |
|  | ProductId | Varchar(20) | FK | Mã giày |
|  | SellerSku | Varchar(50) |  | Mã stock keeping unit do người dùng đặt hoặc hệ thống tự tạo khi người dùng không đặt. Mã SellerSku là duy nhất |
|  | Available | Int(10) |  | Số lượng **có** **thể bán được** còn lại của stock keeping unit. |
|  | Quantity | Int(10) |  | Số lượng của stock keeping unit. |
|  | CorlorFamily | Varchar(50) |  | Màu chính của sản phẩm, do người dùng nhập. Mẫu dữ liệu: Màu trắng |
|  | Size | Varchar(20) |  | Size giày. Mẫu dữ liệu: EU:39 |
|  | Height | Varchar(20) |  | Chiều cao cục hàng sau khi đóng gói. Tính theo cm. |
|  | Width | Varchar(20) |  | Chiều rộng gói hàng sau khi đóng gói. Tính theo cm. |
|  | Length | Varchar(20) |  | Chiều dài gói hàng sau khi đóng gói. Tính theo cm. |
|  | Weight | Varchar(20) |  | Cân nặng gói hàng sau khi đóng gói. Tính theo kg. Mẫu dữ liệu: 1 |
|  | Price | Decimal(15,2) |  | Giá gốc |
|  | SpecialPrice | Decimal(15,2) |  | Giá bán |
|  | SpecialFromTime | Datetime |  | Ngày bắt đầu áp dụng giá bán. (giá bán đuuợc giảm từ giá gốc) |
|  | SpecialToTime | Datetime |  | Ngày kết thúc áp dụng giá bán |
|  | Status | Varchar(20) |  | Khoảng giá trị:  **“active, inactive, deleted**  **”** |

## **ImageProduct**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Tên cột** | **Kiểu dữ liệu** | **Khoá** | **Diễn giải** |
|  | ShopSku | Varchar(50) | PK, FK | Mã stock keeping unit do hệ thống laz tự động đặt. Khóa ShopSku là duy nhất |
|  | Image | Varchar(255) | PK, FK | Url của hình ảnh sau khi uploadlên laz. Laz sẽ trả về 1 url của hình đó. Sử dụng hình ảnh nay để thêm vào SKU |

## **Image**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Tên cột** | **Kiểu dữ liệu** | **Khoá** | **Diễn giải** |
|  | Image | Varchar(255) | PK | Url của hình ảnh |

# **Quy ước**

## **Kiểu dữ liệu MySQL**

### Kiểu số

|  |  |
| --- | --- |
| **Kiểu dữ liệu** | **Mô tả** |
| TINYINT(size) | Lưu trữ một số nguyên có giá trị từ -128 đến -127 hoặc 0 đến 255 |
| SMALLINT(size) | Lưu trữ một số nguyên có giá trị từ -32768 đến 32767 hoặc 0 đến 65535 |
| MEDIUMINT(size) | Lưu trữ một số nguyên có giá trị từ -8388608 đến 8388607 hoặc 0 đến 16777215 |
| INT(size) | Lưu trữ một số nguyên có giá trị từ -2147483648 đến 2147483647 hoặc 0 đến 4294967295 |
| BIGINT(size) | Lưu trữ một số nguyên có giá trị từ -9223372036854775808 đến 9223372036854775807 hoặc 0 đến 18446744073709551615. |
| FLOAT(size,d) | Lưu trữ một số thập phân loại nhỏ (Ví dụ: 567.25). Tham số “size” dùng để xác định kích thước tối đa của phần nguyên (nằm bên trái dấu chấm). Tham số “d” dùng để xác định kích thước tối đa của phần thập phân (nằm bên phải dấu chấm). |
| DOUBLE(size,d) | Lưu trữ một số thập phân loại lớn. Tham số “size” dùng để xác định kích thước tối đa của phần nguyên (nằm bên trái dấu chấm). Tham số “d” dùng để xác định kích thước tối đa của phần thập phân (nằm bên phải dấu chấm). |
| DECIMAL(size,d) | Lưu trữ như một chuỗi, cho phép một dấu thập phân cố định. Tham số “size” dùng để xác định kích thước tối đa của phần nguyên (nằm bên trái dấu chấm). Tham số “d” dùng để xác định kích thước tối đa của phần thập phân (nằm bên phải dấu chấm). |

### Kiểu văn bản

|  |  |
| --- | --- |
| **Kiểu dữ liệu** | **Mô tả** |
| CHAR(size) | Dùng để lưu trữ một chuỗi ký tự có chiều dài tối đa là 255 ký tự |
| VARCHAR(size) | Dùng để lưu trữ một chuỗi ký tự có chiều dài tối đa là 255 ký tự. Nếu đặt “size” lớn hơn 255 thì nó sẽ chuyển sang kiểu TEXT |
| TINYTEXT | Dùng để lưu trữ một chuỗi ký tự có chiều dài tối đa là 255 ký tự |
| TEXT | Dùng để lưu trữ một chuỗi ký tự có chiều dài tối đa là 65,535 ký tự |
| BLOB | Dùng để lưu trữ dữ liệu nhị phân tối đa là 65,535 byte |
| MEDIUMTEXT | Dùng để lưu trữ một chuỗi ký tự có chiều dài tối đa là 16,777,215 ký tự |
| MEDIUMBLOB | Dùng để lưu trữ dữ liệu nhị phân tối đa là 16,777,215 byte |
| LONGTEXT | Dùng để lưu trữ một chuỗi ký tự có chiều dài tối đa là 4,294,967,295 ký tự |
| LONGBLOB | Dùng để lưu trữ dữ liệu nhị phân tối đa là 4,294,967,295 byte |

### Kiểu ngày tháng

|  |  |
| --- | --- |
| **Kiểu dữ liệu** | **Mô tả** |
| DATE() | Lưu trữ một ngày theo định dạng YYYY-MM-DD (Ví dụ: 2016-09-12 tức là lưu ngày 12 tháng 9 năm 2016) |
| TIME() | Lưu trữ thời gian theo định dạng HH:MI:SS (Ví dụ 17:25:36 tức là lưu 17 giờ 25 phút 36 giây) |
| YEAR() | Lưu trữ một năm theo định dạng hai số hoặc bốn số |
| DATETIME() | Lưu trữ một ngày cùng với thời gian theo định dạng YYYY-MM-DD HH:MI:SS (Ví dụ: 2016-09-12 17:25:36 tức là lưu ngày 12 tháng 9 năm 2016 lúc 17 giờ 25 phút 36 giây) |

## **Đặt tên**

1. Sử dụng một kí tự alphabet ở đầu tên, không sử dụng các kí tự “\_” gạch dưới, hoặc số ở đầu tên biến.

2. Giới hạn việc viết tắt quá nhiều trong tên biến hoặc đối tượng. Điều này dễ gây nhầm lẫn.

3. Giới hạn việc sử dụng các từ viết tắt hai lần trong tên.

4. Đặt tên mang ý nghĩa đúng với nội dung và tính chất của biến hoặc đối tượng.

5. Không sử dụng khoảng trắng(space) hoặc các kí tự đặc biệt trong tên như \*,$,#…

6. Tên bảng thường không cần bắt đầu bằng prefix “tbl” hay “tb” và tên bảng nên để ở dạng số ít như “Student” thay vì “Students”. Tên bảng nên sử dụng Pascal Case tức là viết hoa chữ cái đầu của mỗi từ trong tên như SaleDepartment.

7. Nguyên tắc đặt tên cột.

\* Nên đặt tên cột theo kiểu Pascal, hạn chế dùng theo kiểu camelCase( kiểu lạc đà – tức là chữ đầu không viết hoa mà viết thường còn các chữ sau có thể viết hoa chữ cái đầu của mỗi từ như myDocument).

\* Đặt tên khóa chính nên có hậu tố Id thường thì công thức sẽ là tên bảng + “Id”.

\* Đối với khóa ngoại thì nên có tên giống với khóa chính mà nó tham chiếu tới. Ví dụ khóa chính của bảng Staff là “StaffId” và bảng [Class] có khóa ngoại đến bảng Staff thì nó có tên là “StaffId”.

\* Không nên sử dụng các tiến tố trước tên cột.

\* Tên nên chỉ ra loại dữ liệu mà nó chứa ví dụ bạn có 1 trường dữ liệu lưu việc sinh viên có đi học vào ngày nào đó hay không thì thay vì đặt tên biến là Absent bạn có thể đặt là IsAbsent với kiểu dữ liệu là bit.

\* Tên cột phải có ý nghĩa và không nên viết tắt quá nhiều nhưng bạn phải lưu ý đến độ dài của tên cột nên nằm trong khoảng dưới 30 chars là tốt nhất nhưng giới hạn của nó là 50 chars.

\* Tên cột có thể chứa số nhưng không chứa các kí tự đặc biệt.

8. Đặt tên cho index thì nên bắt đầu bằng “IX\_”, đặt tên cho unique column thì nên bắt đầu là “UN\_” hoặc “U/N”

9. Đặt tên cho ràng buộc(Constraint)

\* Công thức: loại ràng buộc+ tên bảng + “\_”+tên cột

\* Một số prefix cho các loại ràng buộc.

o. Primary Key: Pk ví dụ PkProduct\_Id.

o. Foreign Key: Fk ví dụ FkOrder\_ProductId.

o.  Check: Ck ví dụ CkStudent\_Age.

o. Unique: Un ví dụ UnContact\_Phone.

## **Chuyển dữ liệu từ MySQL sang Java**

MySQL Type Java Type

---------- ---------

CHAR String

VARCHAR String

LONGVARCHAR String

NUMERIC java.math.BigDecimal

DECIMAL java.math.BigDecimal

BIT boolean

TINYINT byte

SMALLINT short

INTEGER int

BIGINT long

REAL float

FLOAT double

DOUBLE double

BINARY byte[]

VARBINARY byte[]

LONGVARBINARY byte[]

DATE java.sql.Date

TIME java.sql.Time

TIMESTAMP java.sql.Tiimestamp