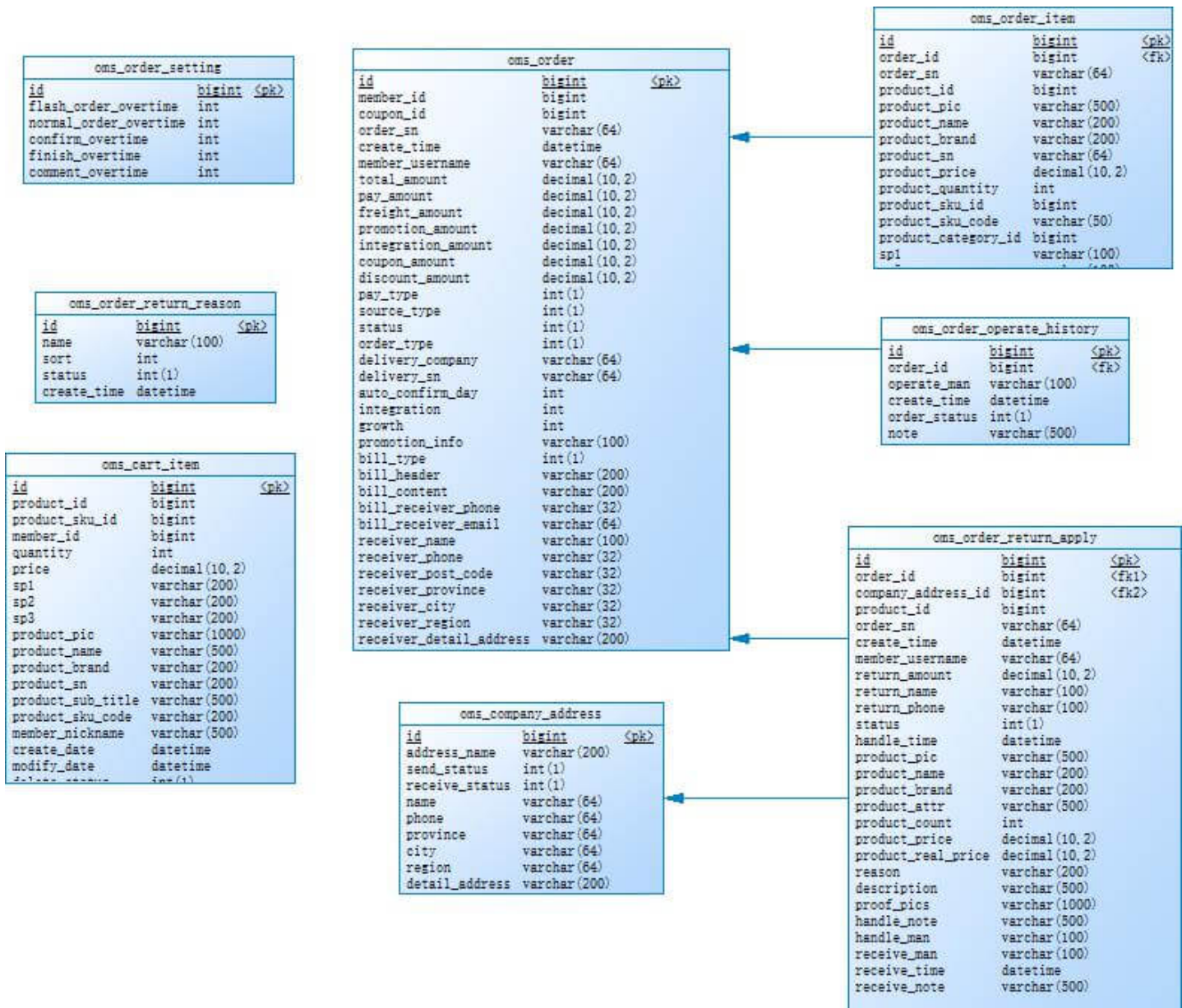


Data Base - 06/15/2022 17:00pm PDT

MySQL - Relational Database

1.1 Practice DB Relationship



1.2 表清单

#	数据表	名称	备注说明
1	oms_order	订单表	
2	oms_order_item	订单中所包含的商品	
3	oms_order_operate_history	订单操作记录表	
4	oms_order_setting	订单设置表	
5	oms_order_return_apply	订单退货申请	
6	oms_company_address	公司收发货地址表	
7	oms_order_return_reason	退货原因表	
8	oms_cart_item	购物车表	

1.3 表字段明细

1.3.1 oms_order [订单表]

#	字段	名称	数据类型	主键	非空	默认值	备注说明
1	id	订单id	bigint	√			
2	member_id	会员id	bigint				
4	order_sn	订单编号	varchar(64)				
5	create_time	提交时间	DATETIME				
6	member_username	用户帐号	varchar(64)				
7	total_amount	订单总金额	decimal(10,2)				
8	pay_amount	应付金额（实际支付金额）	decimal(10,2)				
9	freight_amount	运费金额	decimal(10,2)				
14	pay_type	支付方式：0->未支付；1->支付宝；2->微信	int(1)				
15	source_type	订单来源：0->PC订单；1->app订单	int(1)				
16	status	订单状态：0->待付款；1->待发货；2->已发货；3->已完成；4->已关闭；5->无效订单	int(1)				
17	order_type	订单类型：0->正常订单；1->秒杀订单	int(1)				
18	delivery_company	物流公司(配送方式)	varchar(64)				
19	delivery_sn	物流单号	varchar(64)				

20	auto_confirm_day	自动确认时间（天）	INT				
24	bill_type	发票类型：0->不开发票；1->电子发票；2->纸质发票	int(1)				
29	receiver_name	收货人姓名	varchar(100)				
30	receiver_phone	收货人电话	VARCHAR(32)				
31	receiver_post_code	收货人邮编	VARCHAR(32)				
32	receiver_province	省份/直辖市	VARCHAR(32)				
33	receiver_city	城市	VARCHAR(32)				
34	receiver_region	区	VARCHAR(32)				
35	receiver_detail_address	详细地址	varchar(200)				
36	note	订单备注	varchar(500)				
37	confirm_status	确认收货状态：0->未确认；1->已确认	int(1)				
38	delete_status	删除状态：0->未删除；1->已删除	int(1)				
40	payment_time	支付时间	DATETIME				
41	delivery_time	发货时间	DATETIME				
42	receive_time	确认收货时间	DATETIME				
43	comment_time	评价时间	DATETIME				
44	modify_time	修改时间	DATETIME				

1.3.2 oms_order_item [订单中所包含的商品]

#	字段	名称	数据类型	主键	非空	默认值	备注说明
1	id	id	bigint	√			
2	order_id	订单id	bigint				
3	order_sn	订单编号	varchar(64)				
4	product_id	商品id	bigint				
5	product_pic	商品图片	varchar(500)				
6	product_name	商品名称	varchar(200)				
7	product_brand	商品品牌	varchar(200)				
8	product_sn	商品条码	varchar(64)				
9	product_price	销售价格	decimal(10,2)				
10	product_quantity	购买数量	INT				
11	product_sku_id	商品sku编号	bigint				
12	product_sku_code	商品sku条码	varchar(50)				
13	product_category_id	商品分类id	bigint				
14	sp1	商品的销售属性1	varchar(100)				
15	sp2	商品的销售属性2	varchar(100)				
16	sp3	商品的销售属性3	varchar(100)				
24	product_attr	商品销售属性:[{"key":"颜色"	varchar(500)				"value":"颜色"},{"key":"容量","value":"4G"}]

1.3.3 oms_order_operate_history [订单操作记录表]

#	字段	名称	数据类型	主键	非空	默认值	备注说明
1	id	id	bigint	√			
2	order_id	订单id	bigint				
3	operate_man	操作人：用户；系统；后台管理员	varchar(100)				
4	create_time	操作时间	DATETIME				
5	order_status	订单状态：0->待付款；1->待发货；2->已发货；3->已完成；4->已关闭；5->无效订单	int(1)				
6	note	备注	varchar(500)				

1.3.4 oms_order_setting [订单设置表]

#	字段	名称	数据类型	主键	非空	默认值	备注说明
1	id	id	bigint	√			
3	normal_order_overtime	正常订单超时时间(分)	INT				
4	confirm_overtime	发货后自动确认收货时间（天）	INT				
5	finish_overtime	自动完成交易时间	INT				不能申请售后（天）
6	comment_overtime	订单完成后自动好评时间（天）	INT				

1.3.5 oms_order_return_apply [订单退货申请]

#	字段	名称	数据类型	主键	非空	默认值	备注说明
1	id	id	bigint	√			
2	order_id	订单id	bigint				
3	company_address_id	收货地址表id	bigint				
4	product_id	退货商品id	bigint				
5	order_sn	订单编号	varchar(64)				
6	create_time	申请时间	DATETIME				
7	member_username	会员用户名	varchar(64)				
8	return_amount	退款金额	decimal(10,2)				
9	return_name	退货人姓名	varchar(100)				
10	return_phone	退货人电话	varchar(100)				
11	status	申请状态：0->待处理；1->退货中；2->已完成；3->已拒绝	int(1)				
12	handle_time	处理时间	DATETIME				
13	product_pic	商品图片	varchar(500)				
14	product_name	商品名称	varchar(200)				
15	product_brand	商品品牌	varchar(200)				
16	product_attr	商品销售属性：颜色：红色；尺码：xl	varchar(500)				
17	product_count	退货数量	INT				
18	product_price	商品单价	decimal(10,2)				
19	product_real_price	商品实际支付单价	decimal(10,2)				
20	reason	原因	varchar(200)				
21	description	描述	varchar(500)				
22	proof_pics	凭证图片	varchar(1000)				以逗号隔开
23	handle_note	处理备注	varchar(500)				
24	handle_man	处理人员	varchar(100)				
25	receive_man	收货人	varchar(100)				
26	receive_time	收货时间	DATETIME				
27	receive_note	收货备注	varchar(500)				

1.3.6 oms_company_address [公司收发货地址表]

#	字段	名称	数据类型	主键	非空	默认值	备注说明
1	id	id	bigint	√			
2	address_name	地址名称	varchar(200)				
3	send_status	默认发货地址：0->否；1->是	int(1)				
4	receive_status	是否默认收货地址：0->否；1->是	int(1)				
5	name	收发货人姓名	varchar(64)				
6	phone	收货人电话	varchar(64)				
7	province	省/直辖市	varchar(64)				
8	city	市	varchar(64)				
9	region	区	varchar(64)				
10	detail_address	详细地址	varchar(200)				

1.3.7 oms_order_return_reason [退货原因表]

#	字段	名称	数据类型	主键	非空	默认值	备注说明
1	id	id	bigint	√			
2	name	退货类型	varchar(100)				
3	sort	sort	INT				
4	status	状态：0->不启用；1->启用	int(1)				
5	create_time	添加时间	DATETIME				

1.3.8 oms_cart_item [购物车表]

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

#	字段	名称	数据类型	主键	非空	默认值	备注说明
1	id	id	bigint	√			
2	product_id	商品的id	bigint				
3	product_sku_id	商品sku的id	bigint				
4	member_id	会员id	bigint				
5	quantity	购买数量	INT				
6	price	添加到购物车的价格	decimal(10,2)				
7	sp1	销售属性1	varchar(200)				
8	sp2	销售属性2	varchar(200)				
9	sp3	销售属性3	varchar(200)				
10	product_pic	商品主图	varchar(1000)				
11	product_name	商品名称	varchar(500)				
12	product_brand	商品品牌	varchar(200)				
13	product_sn	商品的条码	varchar(200)				
14	product_sub_title	商品副标题（卖点）	varchar(500)				
15	product_sku_code	商品sku条码	varchar(200)				
16	member_nickname	会员昵称	varchar(500)				
17	create_date	创建时间	DATETIME				
18	modify_date	修改时间	DATETIME				
19	delete_status	是否删除	int(1)				
20	product_category_id	商品的分类	bigint				
21	product_attr	商品销售属性:[{"key":"颜色"	varchar(500)				"value":"银色"},{"key":"容量","value":"4G"}]

Exercise

1. Create oms_company_address table
2. Insert some random data to oms_company_address table
3. Write a SQL query to fetch all data from oms_company_address `table
4. Write a SQL query to fetch top 3 records from oms_company_address table
5. Update oms_company_address table to set all phone to 666-6666-8888
6. Delete one entry from oms_company_address table
7. (Optional) You can also try to create other tables that listed above

MongoDB - Non-SQL Database

SQL概念	MongoDB概念	解释/说明
database	database	数据库
table	collection	数据库表/集合
row	document	数据记录行/文档
column	field	数据字段/域
index	index	索引
primary key	primary key	主键,MongoDB自动将_id字段设置为主键

操作	格式	SQL中的类似语句
等于	<code>{<key>:<value>}</code>	<code>where title = 'MongoDB 教程'</code>
小于	<code>{<key>:{\$lt:<value>}}</code>	<code>where likes < 50</code>
小于或等于	<code>{<key>:{\$lte:<value>}}</code>	<code>where likes <= 50</code>
大于	<code>{<key>:{\$gt:<value>}}</code>	<code>where likes > 50</code>
大于或等于	<code>{<key>:{\$gte:<value>}}</code>	<code>where likes >= 50</code>
不等于	<code>{<key>:{\$ne:<value>}}</code>	<code>where likes != 50</code>

Exercise

1. Create `test` DB
2. Create `oms_company_address` collection (method: `createCollection()`)
3. Insert few random entries to `oms_company_address` collection (method: `insert()`)
4. Read one entry from `oms_company_address` collection (method: `find()`)
5. Read all entries from `oms_company_address` collection (method: `find()`)
6. Update one entry from `oms_company_address` collection (method: `update()` or `save()`)
7. Remove one entry from `oms_company_address` collection (method: `remove()`)
8. (Optional) You can also try to create other tables that listed above