

پروژه ی نهایی پایگاه داده

فروشگاه آنلاین

مهسان اکبری 9531004

زهره گنجی 9531802

موجودیت ها:

1) مشتری:

مشتریان به دو دسته تقسیم میشوند: مشتریانی که قبلا در سامانه ثبت نام کرده اند و مشتریان جدید. پس 2 گروه تخصصی از customer یعنی new\_customer و old\_customer را ایجاد میکنیم. تخصصی شدن customer به ما اجازه میدهد بین موجودیت های مشتری برحسب این که از قبل ثبت نام کرده اند یا جدید هستند تمایز قایل شویم. روابط و صفات در مجموعه موجودیت customer توسط دو مجموعه موجودیت old\_customer و new\_customer به ارث برده می شود. ما در این پروژه صفت username را برای موجودیت customer در نظر گرفتیم که کلید اصلی در این رابطه است و username در new\_customer و old\_customer کلید خارجی است که ارجاع میدهد به رابطه ی customer. هریک از مجموعه موجودیت های new\_customer و old\_customer علاوه بر صفت username دارای صفات دیگری نیز هستند و این صفات باعث تمایز آن ها از یک دیگر می شود.

چون موجودیت های مشتری حداکثر میتوانند در یک گروه تخصصی حضور داشته باشند، این تخصصی شدن را تخصصی شدن جدا از هم (disjoint specialization) می نامیم و در نمودار E-R آن را با یک فلش نشان میدهیم.

• مشتری: (customer)



#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	username	char(20)	latin1_swedish_ci		No	None			Change Drop More

- مشتریانی که از قبل ثبت نام کرده اند: (old\_customer)  
چون هر کاربر میتواند چند آدرس و شماره تلفن داشته باشد، صفت address و phone number صفت چند مقداری هستند و برای این صفت ها شمای رابطه ای جداگانه ای را ایجاد می کنیم و جدول های جدیدی میسازیم. کد پستی نیز با آدرس تغییر میکند پس جزو صفت چند مقداری است.

Server: 127.0.0.1 » Database: store » Table: old\_customer

Table structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	username	char(20)	latin1_swedish_ci		Yes	NULL			Change Drop More
2	password1	varchar(42)	latin1_swedish_ci		Yes	NULL			Change Drop More
3	type	int(11)			No	1			Change Drop More
4	email	varchar(40)	latin1_swedish_ci		Yes	NULL			Change Drop More
5	firstName	varchar(20)	latin1_swedish_ci		Yes	NULL			Change Drop More
6	lastName	varchar(30)	latin1_swedish_ci		Yes	NULL			Change Drop More
7	gender	set('female', 'male')	latin1_swedish_ci		Yes	NULL			Change Drop More
8	credit	int(11)			Yes	NULL			Change Drop More

Check all With selected: Browse Change Drop Primary Unique Index Fulltext Add to central columns

## Customer\_address

Server: 127.0.0.1 » Database: store » Table: customer\_address

Table structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	username	char(20)	latin1_swedish_ci		Yes	NULL			Change Drop More
2	address	varchar(100)	latin1_swedish_ci		Yes	NULL			Change Drop More
3	postalCode	int(11)			Yes	NULL			Change Drop More

Check all With selected: Browse Change Drop Primary Unique Index Fulltext Add to central columns Remove from central columns

## Customer\_phone

Server: 127.0.0.1 » Database: store » Table: customer\_phone

Table structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	username	char(20)	latin1_swedish_ci		Yes	NULL			Change Drop More
2	phoneNo	int(11)			Yes	NULL			Change Drop More

Check all With selected: Browse Change Drop Primary Unique Index Fulltext Add to central columns Remove from central columns

- مشتریان جدید: (new\_customer)

Server: 127.0.0.1 » Database: store » Table: new\_customer

Table structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	username	char(20)	latin1_swedish_ci		Yes	NULL			Change Drop More
2	type	int(11)			Yes	0			Change Drop More

Check all With selected: Browse Change Drop Primary Unique Index Fulltext Add to central columns  
Remove from central columns

Print Propose table structure Track table Move columns Normalize

## 2) فروشگاه: (shop)

کلید اصلی را شناسه ی فروشگاه (sh\_id) قرار می دهیم.

Server: 127.0.0.1 » Database: store » Table: shop

Table structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	sh_id	char(20)	latin1_swedish_ci		No	None			Change Drop More
2	title	varchar(20)	latin1_swedish_ci		Yes	NULL			Change Drop More
3	city	varchar(20)	latin1_swedish_ci		Yes	NULL			Change Drop More
4	address	varchar(100)	latin1_swedish_ci		Yes	NULL			Change Drop More
5	phone_no	int(11)			Yes	NULL			Change Drop More
6	manager_name	varchar(30)	latin1_swedish_ci		Yes	NULL			Change Drop More
7	openTime	time			Yes	NULL			Change Drop More
8	closeTime	time			Yes	NULL			Change Drop More
9	credit	int(11)			Yes	0			Change Drop More

## 3) عوامل عملیات: (operator)

کلید اصلی را شناسه ی عوامل عملیات (op\_id) قرار می دهیم.

Server: 127.0.0.1 » Database: store » Table: operator

Table structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	op_id	char(20)	latin1_swedish_ci		No	None			Change Drop More
2	firstName	varchar(20)	latin1_swedish_ci		Yes	NULL			Change Drop More
3	lastName	varchar(30)	latin1_swedish_ci		Yes	NULL			Change Drop More

Check all With selected: Browse Change Drop Primary Unique Index Fulltext Add to central columns

## 4) عوامل پیک: (deliveryman)

کلید اصلی را شناسه ی پیک (d\_id) قرار می دهیم.

Server: 127.0.0.1 » Database: store » Table: deliveryman

Table structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	d_id	char(20)	latin1_swedish_ci		No	None			Change Drop More
2	firstName	varchar(20)	latin1_swedish_ci		Yes	NULL			Change Drop More
3	lastName	varchar(30)	latin1_swedish_ci		Yes	NULL			Change Drop More
4	phone_no	int(11)			Yes	NULL			Change Drop More
5	state	set('free', 'sending')	latin1_swedish_ci		Yes	NULL			Change Drop More
6	credit	int(11)			Yes	NULL			Change Drop More

## 5) سفارش: (order1)

هر فروشگاه دارای تعدادی سفارش است. پس هر سفارش یک صفت به نام sh\_id دارد که id فروشگاه را نشان می دهد. و قید کلید خارجی روی sh\_id اعمال می شود و آن را ارجاع می دهد به صفت id در موجودیت فروشگاه.

هر مشتری تعدادی سفارش را میدهد پس بین مجموعه موجودیت مشتری و سفارش یک ارتباط یک به چند داریم و در رابطه ی order1 قید کلید خارجی را روی username اعمال می کنیم که به شناسه ی کاربری در customer ارجاع می دهد.

کلید اصلی را شناسه ی سفارش (or\_id) قرار می دهیم.

Server: 127.0.0.1 » Database: store » Table: order1

Table structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	or_id	char(20)	latin1_swedish_ci		No	None			Change Drop More
2	sh_id	char(20)	latin1_swedish_ci		Yes	NULL			Change Drop More
3	username	char(20)	latin1_swedish_ci		Yes	NULL			Change Drop More
4	deliveryman_id	char(20)	latin1_swedish_ci		Yes	NULL			Change Drop More
5	orderState	set('registered', 'sent', 'completed', 'failed')	latin1_swedish_ci		Yes	registered			Change Drop More
6	payment_type	set('by credit', 'via bank portal')	latin1_swedish_ci		Yes	NULL			Change Drop More
7	orderDate	date			Yes	NULL			Change Drop More
8	orderAddress	varchar(100)	latin1_swedish_ci		Yes	NULL			Change Drop More
9	orderTime	time			Yes	NULL			Change Drop More
10	flag	int(11)			No	0			Change Drop More

## 6) محصول: (product)

هر فروشگاه دارای تعدادی کالا است. پس هر کالا یک صفت به نام sh\_id دارد که id فروشگاه را نشان می دهد. و قید کلید خارجی روی sh\_id اعمال می شود و آن را ارجاع می دهد به صفت id در موجودیت فروشگاه.

کلید اصلی در مجموعه موجودیت محصول، شناسه ی محصول (p\_id) است.

Server: 127.0.0.1 » Database: store » Table: product

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	p_id	char(20)	latin1_swedish_ci		No	None			Change Drop More
2	p_title	varchar(20)	latin1_swedish_ci		Yes	NULL			Change Drop More
3	sh_id	char(20)	latin1_swedish_ci		Yes	NULL			Change Drop More
4	price	int(11)			Yes	NULL			Change Drop More
5	discount	int(11)			Yes	NULL			Change Drop More
6	amount	int(11)			Yes	NULL			Change Drop More

Check all With selected: Browse Change Drop Primary Unique Index Fulltext Add to central columns

## 7) حساب بانک:(bank\_account)

هر کاربر میتواند درگاه بانک را داشته باشد پس در این مجموعه موجودیت، یک صفت به نام username داریم که قید کلید خارجی را روی آن اعمال می کنیم و ارجاع می دهد به username در رابطه ی مشتری.

Server: 127.0.0.1 » Database: store » Table: bank\_account

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	username	char(20)	latin1_swedish_ci		Yes	NULL			Change Drop More
2	bankCredit	int(11)			Yes	NULL			Change Drop More

Check all With selected: Browse Change Drop Primary Unique Index Fulltext Add to central columns

## روابط:

### 1) رابطه ی بین مشتری و سفارش:(customer\_product)

یک رابطه ی یک به چند از مشتری به سفارشات است. چون هر مشتری می تواند چند سفارش داشته باشد و مشارکت هر سفارش در این رابطه کامل است چون هر سفارش حداقل باید توسط یک مشتری ثبت شود.

### 2) رابطه ی بین مشتری و عوامل پشتیبان:(conversation)

عوامل پشتیبان وظیفه ی ارتباط با مشتریان را بر عهده دارند. این رابطه ، یک رابطه ی یک به چند است. چون هر عامل پشتیبان می تواند با چند مشتری ارتباط داشته باشد .

Server: 127.0.0.1 » Database: store » Table: conversation

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	customer_username	char(20)	latin1_swedish_ci		Yes	NULL			Change Drop More
2	order_id	char(20)	latin1_swedish_ci		Yes	NULL			Change Drop More
3	message	longtext	latin1_swedish_ci		Yes	NULL			Change Drop More

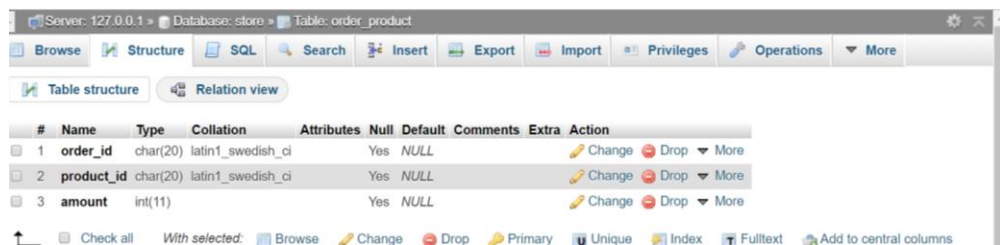
Check all With selected: Browse Change Drop Primary Unique Index Fulltext Add to central columns

### 3) رابطه ی بین سفارش و محصول:(order\_product)

یک رابطه ی چند به چند از سفارشات به محصول است.مشارکت سفارش در این ارتباط کامل است چون همه ی سفارشات باید حداقل شامل یک محصول باشند.

برای این رابطه یک جدول در پایگاه داده ایجاد کرده ایم و شمای رابطه ی آن دارای کلید های اصلی دو مجموعه موجودیت سفارش و محصول است که قید کلید خارجی را روی آن ها اعمال می کنیم.

این رابطه دارای یک صفت دیگر به نام amunt نیز است که مقدار محصول سفارش شده را مشخص میکند.



The screenshot shows the MySQL Table Structure for the 'order\_product' table. The table has three columns: 'order\_id' (char(20), latin1\_swedish\_ci, Yes, NULL), 'product\_id' (char(20), latin1\_swedish\_ci, Yes, NULL), and 'amount' (int(11), Yes, NULL). The 'order\_id' and 'product\_id' columns are marked as primary keys. The interface includes tabs for 'Table structure' and 'Relation view', and a toolbar with options like 'Check all', 'Browse', 'Change', 'Drop', 'Primary', 'Unique', 'Index', 'Fulltext', and 'Add to central columns'.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	order_id	char(20)	latin1_swedish_ci		Yes	NULL			Change Drop More
2	product_id	char(20)	latin1_swedish_ci		Yes	NULL			Change Drop More
3	amount	int(11)			Yes	NULL			Change Drop More

### 4) رابطه ی بین سفارش و فروشگاه:(shop\_order1)

این رابطه یک رابطه ی یک به چند از فروشگاه به سفارش است چون فروشگاه میتواند سفارش های متعدد داشته باشد اما هر سفارش مختص یک فروشگاه است.و مشارکت سفارش دراین رابطه کامل است چون هر سفارش باید حداقل به یک فروشگاه مرتبط باشد.

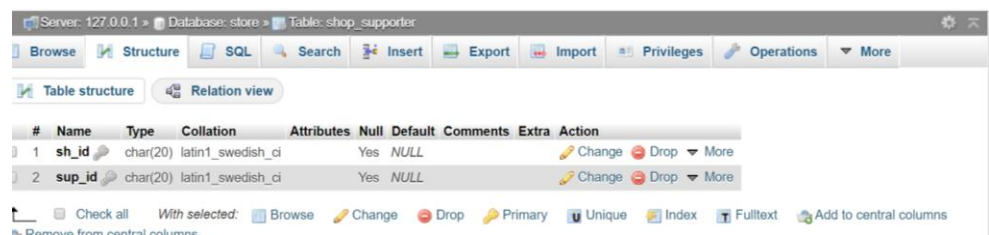
### 5) رابطه ی بین فروشگاه و محصول:(shop\_product)

این رابطه یک رابطه ی یک به چند از فروشگاه به محصول است چون فروشگاه میتواند محصول های متعدد داشته باشد اما هر محصول مختص یک فروشگاه است و مشارکت محصول در آن کامل است چون هر محصول باید حداقل به یک فروشگاه مرتبط باشد.یعنی باید حداقل متعلق به یک فروشگاه باشد.

### 6) رابطه ی بین فروشگاه و عوامل پشتیبان:(shop\_supporter)

ای رابطه، یک رابطه ی یک به چند از فروشگاه به عوامل پشتیبان است و مشارکت عوامل پشتیبان در آن کامل است چون هر عامل پشتیبان حداقل باید به یک فروشگاه تعلق داشته باشد و ارتباط داشته باشد.

برای این رابطه یک جدول در پایگاه داده ایجاد کرده ایم و شمای رابطه ی آن دارای کلید های اصلی دو مجموعه موجودیت فروشگاه و عوامل پشتیبان است که قید کلید خارجی را روی آن ها اعمال می کنیم.



The screenshot shows the MySQL Table Structure for the 'shop\_supporter' table. The table has two columns: 'sh\_id' (char(20), latin1\_swedish\_ci, Yes, NULL) and 'sup\_id' (char(20), latin1\_swedish\_ci, Yes, NULL). Both columns are marked as primary keys. The interface includes tabs for 'Table structure' and 'Relation view', and a toolbar with options like 'Check all', 'Browse', 'Change', 'Drop', 'Primary', 'Unique', 'Index', 'Fulltext', and 'Add to central columns'.

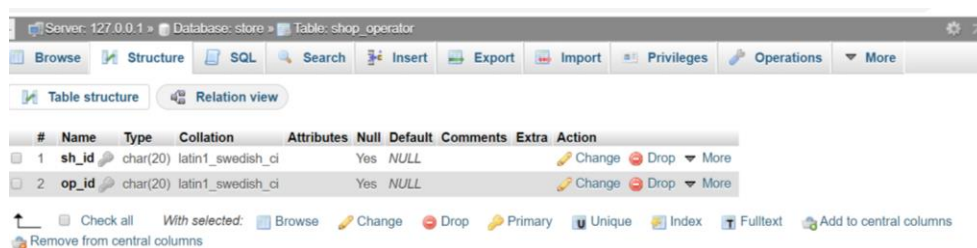
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	sh_id	char(20)	latin1_swedish_ci		Yes	NULL			Change Drop More
2	sup_id	char(20)	latin1_swedish_ci		Yes	NULL			Change Drop More



### 7) رابطه ی بین فروشگاه و عوامل عملیات:(shop\_operator)

ای رابطه، یک رابطه ی یک به چند از فروشگاه به عوامل عملیات است و مشارکت عوامل عملیات در آن کامل است چون هر عامل عملیات حداقل باید به یک فروشگاه تعلق داشته باشد و ارتباط داشته باشد.

برای این رابطه یک جدول در پایگاه داده ایجاد کرده ایم و شمای رابطه ی آن دارای کلید های اصلی دو مجموعه موجودیت فروشگاه و عوامل عملیات است که قید کلید خارجی را روی آن ها اعمال می کنیم.



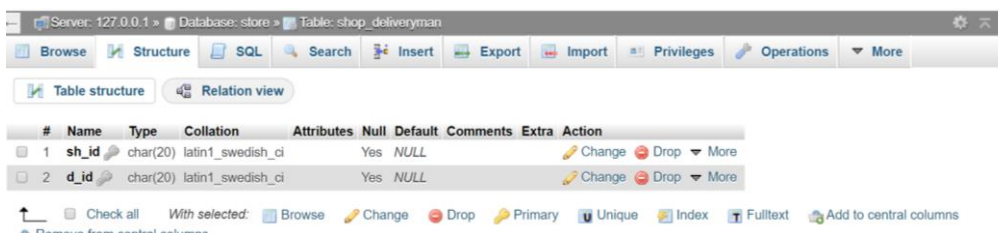
The screenshot shows the MySQL Table Structure window for the 'shop\_operator' table. The table has two columns: 'sh\_id' and 'op\_id', both of type 'char(20)' with 'latin1\_swedish\_ci' collation. Both columns are marked as 'Yes' for 'Attributes' and 'Null', and 'NULL' for 'Default'. The 'Extra' column is empty for both. The 'Action' column shows 'Change', 'Drop', and 'More' options for each column. The window title is 'Server: 127.0.0.1 » Database: store » Table: shop\_operator'.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	sh_id	char(20)	latin1_swedish_ci	Yes	NULL				Change Drop More
2	op_id	char(20)	latin1_swedish_ci	Yes	NULL				Change Drop More

### 8) رابطه ی بین فروشگاه و عوامل پیک:(shop\_deliveryman)

ای رابطه، یک رابطه ی یک به چند از فروشگاه به عوامل پیک است و مشارکت عوامل پیک در آن کامل است چون هر عامل پیک حداقل باید به یک فروشگاه تعلق داشته باشد و ارتباط داشته باشد.

برای این رابطه یک جدول در پایگاه داده ایجاد کرده ایم و شمای رابطه ی آن دارای کلید های اصلی دو مجموعه موجودیت فروشگاه و عوامل پیک است که قید کلید خارجی را روی آن ها اعمال می کنیم.



The screenshot shows the MySQL Table Structure window for the 'shop\_deliveryman' table. The table has two columns: 'sh\_id' and 'd\_id', both of type 'char(20)' with 'latin1\_swedish\_ci' collation. Both columns are marked as 'Yes' for 'Attributes' and 'Null', and 'NULL' for 'Default'. The 'Extra' column is empty for both. The 'Action' column shows 'Change', 'Drop', and 'More' options for each column. The window title is 'Server: 127.0.0.1 » Database: store » Table: shop\_deliveryman'.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	sh_id	char(20)	latin1_swedish_ci	Yes	NULL				Change Drop More
2	d_id	char(20)	latin1_swedish_ci	Yes	NULL				Change Drop More

### 9) رابطه ی بین مشتری و حساب بانک:(charging\_credit)

این رابطه یک رابطه ی یک به یک است برای این که مشتری بتواند به درگاه بانک دسترسی داشته باشد. مشتری و حساب بانک در این رابطه مشارکت کامل دارند. همه ی مشتریان باید حداقل یک حساب بانکی را داشته باشند و همه ی حساب های بانکی متعلق به مشتریان هستند یعنی حساب بانکی ای نداریم که مشتری نداشته باشد. هیچ دو مشتری ای حساب بانکی یکسان ندارند یعنی حتما بین حساب های بانکی تفاوت باید باشد برای مشتریان مختلف و هیچ دو حساب بانکی ای متعلق به یک مشتری نیستند.

برای این رابطه یک جدول در پایگاه داده ایجاد کرده ایم.

Server: 127.0.0.1 » Database: store » Table: charging\_credit

Table structure Relation view

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	username	char(20)	latin1_swedish_ci		Yes	NULL			Change Drop More
2	chargingAmount	int(11)			Yes	NULL			Change Drop More

Check all With selected: Browse Change Drop Primary Unique Index Fulltext Add to central columns Remove from central columns

## Log:

Log ها برای ثبت تغییرات در سیستم استفاده می شوند.

Log ها را با تریگر پیاده سازی می کنیم و برای ثبت log ها تغییرات را در جدول هایی ذخیره می کنیم.

### 1) تغییر وضعیت پیک:

ابتدا جدول deliverymanstatechangelog را می سازیم.

Server: 127.0.0.1 » Database: store » Table: deliverymanstatechangelog

Table structure Relation view

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	deliveryman_id	char(20)	latin1_swedish_ci		No	None			Change Drop More
2	oldState	set('free', 'sending')	latin1_swedish_ci		Yes	NULL			Change Drop More
3	newState	set('free', 'sending')	latin1_swedish_ci		Yes	NULL			Change Drop More
4	changeDate	datetime			Yes	NULL			Change Drop More

Check all With selected: Browse Change Drop Primary Unique Index Fulltext Add to central columns Remove from central columns

سپس تریگر زیر را روی جدول deliveryman بعد از به هنگام سازی این جدول اجرا می کنیم.



**Edit trigger**

**Details**

Trigger name: deliveryman\_state\_change\_LOG

Table: deliveryman

Time: AFTER

Event: UPDATE

**Definition**

```

1 BEGIN
2 INSERT INTO
3 deliverymanstatechangelog(deliveryman_id,oldState,newState,changeDate) VALUES
4 (NEW.d_id ,OLD.state,NEW.state, NOW());
5 END

```

Definer: root@localhost

Go Close

2) تغییر وضعیت سفارش:

ابتدا جدول orderstatechangelog را می سازیم.

Server: 127.0.0.1 » Database: store » Table: orderstatechangelog

Table structure Relation view

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	order_id	char(20)	latin1_swedish_ci		No	None			Change Drop More
2	oldState	set('registered', 'sent', 'completed', 'failed')	latin1_swedish_ci		Yes	NULL			Change Drop More
3	newState	set('registered', 'sent', 'completed', 'failed')	latin1_swedish_ci		Yes	NULL			Change Drop More
4	changeDate	datetime			Yes	NULL			Change Drop More

Check all With selected: Browse Change Drop Primary Unique Index Fulltext Add to central columns

Remove from central columns

Print Propose table structure Track table Move columns Normalize

Add 1 column(s) after changeDate Go

سپس تریگر زیر را روی جدول order1 بعد از به هنگام سازی این جدول اجرا می کنیم.

**Edit trigger**

**Details**

Trigger name: orderState\_update

Table: order1

Time: AFTER

Event: UPDATE

**Definition**

```

1 BEGIN
2 IF NEW.orderState<>OLD.orderState THEN
3 INSERT INTO orderstatechangelog
4 (order_id,oldState,newState,changeDate)VALUES(OLD.or_id,OLD.orderState,NEW.o
5 rderState,NOW());
6 End IF;
7 IF (NEW.orderState<>OLD.orderState and NEW.orderState='sent')THEN
8 UPDATE product P
9 set amount=amount-(SELECT order_product.amount
10 FROM order1,order_product
11 WHERE order1.or_id=NEW.or_id and
12 order1.or_id=order_product.order_id and order_product.product_id=P.p_id);
13 End IF;
14 END

```

Definer: root@localhost

XAMPP Control Panel v3.2.2 [Compiled: Nov 12th 2015]

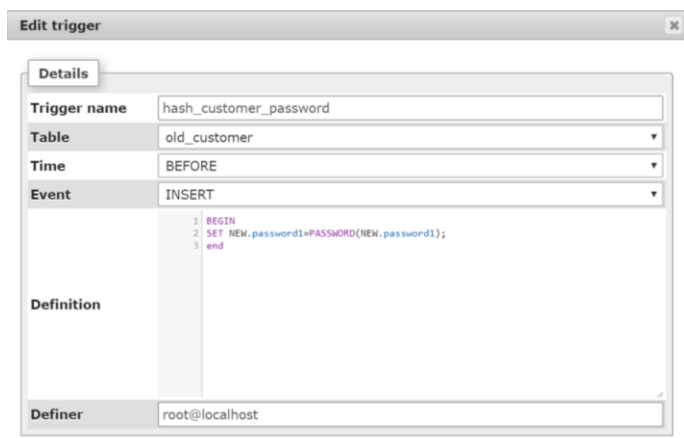
Go Close

## تریگر ها:

### hash\_customer\_password(1)

رمز عبور به صورت درهم سازی شده باید در پایگاه داده ذخیره شود.

برای این امر یک تریگر روی جدول old\_customer ایجاد میکنیم. این تریگر قبل از اضافه کردن تاپلی در جدول old\_customer اجرا می شود و رمز عبور آن را به صورت درهم سازی شده ذخیره میکند.



Edit trigger	
Trigger name	hash_customer_password
Table	old_customer
Time	BEFORE
Event	INSERT
Definition	<pre>1 BEGIN 2 SET NEW.password1=PASSWORD(NEW.password1); 3 end</pre>
Definer	root@localhost

### orderState\_modify(2)

این تریگر برای قسمت های زیر اجرا میشود:

تغییر وضعیت سفارش که در حالت های زیر تغییر می کند:

- هر فروشگاه فقط در بازه های زمانی تعیین شده میتواند سرویس دهی داشته باشد و اگر در زمانی که فروشگاه بسته است سفارشی ثبت شود، سفارش رد میشود .
- مسئولین عملیات سفارش ثبت شده را بررسی میکنند و در صورت نبود مشکل، بررسی میکنند که اگر پیکی آزاد بود، این سفارش را به پیک مربوطه اختصاص میدهد و وضعیت سفارش را به در حال ارسال تغییر میدهد. در صورتی که هیچ پیکی در آن فروشگاه آزاد نبود، سفارش را رد میکند. ( سفارشی بدون مشکل است که تمامی کالاهای سفارش داده از سمت مشتری در فروشگاه موجودی داشته باشد. اگر یکی از محصولات در فروشگاه موجودی کافی نداشته باشد، سفارش رد میشود.)

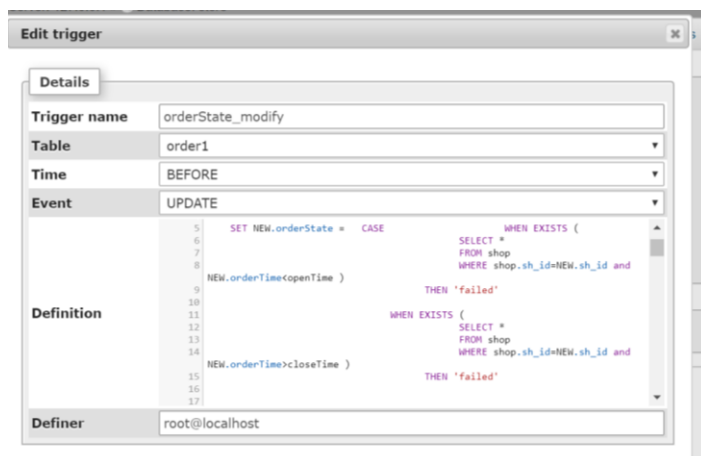
در این تریگر، در صورتی که سفارش در وضعیت رد شدن باشد، تاپلی متناسب با آن سفارش و یوزرنیم مشتری و دلیل رد شدن آن سفارش به جدول conversation اضافه میشود.

کاربر میتواند با اعتبار حساب کاربری خود سفارشهای خود را ثبت کند و مبلغ سفارش از آن کم شود.

زمانی که سفارش رد نشود، اگر مشتری، مشتری ای باشد که از قبل ثبت نام کرده است و نوع پرداخت با حساب کاربری باشد، `old_customer` را بهنگام میکنیم و مقدار `credit` آن را بهنگام می کنیم (یعنی مبلغ سفارش را از آن کم میکنیم)، در غیر این صورت یا مشتری یک مشتری جدید است یا یک مشتری قدیمی است ولی نوع پرداختش از طریق درگاه بانک است، برای این دو حالت مبلغ سفارش را از `bank_account` برای آن مشتری کسر می کند.

و در قسمت `update_shop` 95 درصد از هزینه ی سفارش به حساب فروشگاه می که از آن خرید می شود منتقل می شود .

برای این امر یک تریگر روی جدول `order1` ایجاد می کنیم. که قبل از بهنگام سازی این جدول اجرا شود .



### orderState\_update(3)

این تریگر برای قسمت هر سفارش که انجام ، 5 درصد مبلغ سفارش به اعتبار حساب پیک افزوده میشود اجرا میشود.

این تریگر بعد از بهنگام سازی جدول `order1` اجرا می شود. در این تریگر موارد زیر هندل میشوند:

1. LOG گرفتن از تغییر وضعیت سفارش
2. کاهش موجودی کالاهای سفارش در وضعیت ارسال از فروشگاه مربوطه
3. زمانی که وضعیت سفارش به تکمیل شده تغییر میکند، وضعیت پیک مربوط به آن سفارش از در حال ارسال به آزاد تغییر میکند.
4. مرحله روم از تخصیص پیک به سفارش که رد نشده است.
5. انتقال 5 درصد از مبلغ سفارش به اعتبار پیک

**Edit trigger**

**Details**

Trigger name: orderState\_update

Table: order1

Time: AFTER

Event: UPDATE

**Definition**

```

1 BEGIN
2 IF NEW.orderState<>OLD.orderState THEN
3 INSERT INTO orderstatechangelog
4 (order_id,oldState,newState,changeDate)VALUES(OLD.or_id,OLD.orderState,NEW.o
5 rderState,NOW());
6 End IF;
7 IF (NEW.orderState<>OLD.orderState and NEW.orderState='sent')THEN
8 UPDATE product P
9 set amount=amount-(SELECT order_product.amount
10 FROM order1,order_product
11 WHERE order1.or_id=NEW.or_id and
12 order1.or_id=order_product.order_id and order_product.product_id=P.p_id);
13 End IF;

```

Definer: root@localhost

Go Close

#### charging<sup>4</sup>

این تریگر برای قسمت کاربر میتواند با درگاه بانک، حساب خود را شارژ کند، اجرا می شود.

این تریگر بعد از insert در جدول charging\_credit اجرا می شود.

**Edit trigger**

**Details**

Trigger name: charging

Table: charging\_credit

Time: AFTER

Event: INSERT

**Definition**

```

1 BEGIN
2 UPDATE bank_account
3 SET bankCredit=bankCredit+NEW.chargingAmount
4 WHERE bank_account.username=NEW.username;
5
6 UPDATE old_customer
7 SET credit=credit+NEW.chargingAmount
8 WHERE old_customer.username=NEW.username;
9
10 END

```

Definer: root@localhost

#### deliveryman\_state\_change\_LOG (5)

تریگریست برای log گرفتن از تغییر وضعیت پیک