

گزارشکار درس پایگاه داده

استاد:

دکتر شجاع

دانشگاه ارومیه

- ▶ این گزارشکار درمورد الحاق شدن چندین رستوران ذنجیره ای برای تشکیل یک رستوران بزرگ تر است.
- ▶ تعدادی رستوران هر یک با یک نوع منو متفاوت و تعدادی شعبه های متفاوت به گونه ای به یکدیگر متصل میشوند که با وجود اتصال اطلاعات مختص خود را حفظ می کنند.
- ▶ کد تشکیل هر table در تصاویر آورده شده و شرح داده میشود. برای هر یک تعدادی رکورد آزمایشی اضافه شده و چندین پرمسمان نیز به منظور آزمایش دیتا بیس انجام شده و نتایج آن مشخص است.
- ▶ درنهایت مدل ER نیز رسم و توضیح داده میشود.
- ▶ اطلاعاتی که در مثال آورده شده است به صورت دستی و گرافیکی وارد شده است.

در تصاویر زیر مراحل ساخت جداول شعبه و رستوران دیده میشود.
جدول رستوران دارای ID به عنوان کلید اصلی است.(عبارت identity برای auto generate کردن استفاده میشود.
جدول رستوران همچنین شامل ارزش سهام رستوران و آدرس و شماره تلفن مرکز رستوران است.
جدول شعبه شامل اطلاعاتی چون branch_number به عنوان کلید اصلی و اطلاعاتی دیگر همچون شماره تلفن و آدرس شعبه هست.
Restaurant_id یک کلید خارجی است که هر شعبه را به یک رستوران مرتبط میکند.

```
create table resturant(  
ID int not null identity(10000,1),  
resturant_name varchar(20),  
share_worth int,  
HQ_address varchar(100),  
HQ_number varchar(20),  
primary key(ID));
```

```
create table branch(  
branch_number int not null identity(100000,3),  
branch_address varchar(50),  
resturant_id int,  
phone_number varchar(20),  
primary key(branch_number),  
foreign key(resturant_id) references resturant);
```

مثال هایی از رستوران و شعبه

	branch_nu...	branch_ad...	resturant_id	phone_nu...
▶	100000	tabriz valias...	10000	465465794
	100003	tehran shem...	10000	789875454
	100006	tabriz park l...	10001	456454414
	100009	tehran borj...	10001	456418898
*	NULL	NULL	NULL	NULL

	ID	resturant_n...	share_worth	HQ_address	HQ_number
▶	10000	fastfood	2000000	sahand stre...	0443xxx2257
	10001	kebab	1500000	olive street	0443xxx3489
*	NULL	NULL	NULL	NULL	NULL

جدول employee با کلید اصلی ID شامل نام و نام خانوادگی و شماره تلفن و حقوق هریک از کارکنان است. این جدول توسط branch_id به شعبه به منظور مطلع شدن از اینکه هر یک از کارکنان در کدام شعبه مشغول به کارند. restaurant_id نیز این جدول را به یک رستوران متصل می کند تا بدانیم هر کارکن متعلق به کدام رستوران است.

مدیر شامل ایدی یک کارکن و یک شعبه و به همراه حقوق ان کارمند است. در واقع مدیر خود یک رابطه بین جداول شعبه و کارکن است.

```
create table employee(  
ID int not null identity(100001,1),  
firstname varchar(20),  
lastname varchar(20),  
phone_number varchar(20),  
salary int,  
branch_id int,  
restaurant_id int,  
primary key (ID),  
foreign key (restaurant_id) references restaurant,  
foreign key (branch_id) references branch );
```

```
create table manager(  
salary int,  
employee_id int,  
branch_id int,  
foreign key (employee_id) references employee,  
foreign key (branch_id) references branch);
```


مثال هایی از کارکنان و مدیران

	ID	firstname	lastname	phone_nu...	salary	branch_id	resturant_id
▶	100001	ali	hosseini	654845188	10000	100000	10000
	100002	sasan	amiri	494894598	10000	100000	10000
	100003	reza	hamidi	654654654	15000	100000	10000
	100004	hossein	ghasemi	456445645	18000	100000	10000
	100005	saman	hatami	564646556	30000	100000	10000
	100006	farzad	rezayi	465465465	12000	100003	10000
	100007	mohamd	mohamdi	141561156	11000	100003	10000
	100008	nasrin	fatemi	165115612	11000	100003	10000
	100009	maryam	ghorani	874561654	13000	100003	10000
	100010	amir	ahvazi	165156655	20000	100003	10000
	100011	pouya	tehrani	145194984	25000	100006	10001
	100012	zahra	hasheni	198464894	13000	100006	10001
	100013	fateme	valaiy	165461565	14000	100006	10001
	100014	kimia	najimy	615651561	18000	100006	10001
	100015	atena	saheri	446862158	17000	100006	10001
	100016	zari	zarani	547798257	12000	100009	10001
	100017	morteza	qasemi	877885474	10000	100009	10001
	100018	jafar	jafarzade	746541891	11000	100009	10001
	100019	ali	asghari	185789789	16000	100009	10001
	100020	mohamad	shojazade	698784544	60000	100009	10001
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

	salary	employee_id	branch_id
▶	30000	100005	100000
	20000	100010	100003
	25000	100011	100006
	60000	100020	100009
*	NULL	NULL	NULL

جدول غذا شامل یک ID غذا به عنوان کلید اصلی و نام و توضیحات غذا است. جدول منو تعدادی از غذا را به یک رستوران مرتبط میکند. توجه داشته باشیم که رستوران های متفاوت می توانند غذا های مشابهی سرو کنند از این رو دو کلید خارجی `restaurant_id` و `food_id` با هم کلید اصلی هر منو هستند.

جدول مشتری شامل ID به عنوان کلید اصلی و اطلاعاتی مثل نام و شماره تلفن است. از آنجاکه در پروژه پیک غذا در نظر گرفته نشده است به ادرس مشتری یا دیگر اطلاعات آن نیاز نداریم.

```
create table food(  
  food_id int not null identity(101,1) primary key,  
  food_name varchar(20),  
  food_description varchar(100)  
);
```

```
create table menu(  
  restaurant_id int ,  
  food_id int,  
  primary key(restaurant_id, food_id),  
  foreign key (restaurant_id) references restaurant,  
  foreign key (food_id) references food);
```

```
create table customer(  
  ID int not null identity(1000001,1) primary key,  
  customer_name varchar(20),  
  phone_number varchar(20));
```

مثال هایی از منو و غذا و مشتری

	ID	customer_n...	phone_nu...
▶	1000001	mohmad ha...	44894567
	1000002	ali rezayi	49874878
	1000003	morteza za...	47888787
	1000004	shakib shaki...	87987898
	1000005	hamed ham...	87897798
*	NULL	NULL	NULL

	resturant_id	food_id
▶	10000	101
	10000	102
	10000	103
	10000	104
	10000	105
	10001	106
	10001	107
	10001	108
	10001	109
	10001	110
*	NULL	NULL

	food_id	food_name	food_descr...
▶	101	hamburger	adsasdadad...
	102	pizza	fsdfsfsfsdfs
	103	cheeseburger	adsdadsada...
	104	pasta	pklh[hfg[kh...
	105	morghsokh...	kdkpogkjfp...
	106	kabab kubide	dgfdfgdghf...
	107	kabab jooje	4654654656...
	108	kabab negini	jkldjgfkjdfj...
	109	kabab bargh	lgjdjgoijgfi...
	110	kabab lobn...	fuojdhdsfoj...
*	NULL	NULL	NULL

جدول orderinfo با کلید اصلی ID و دیگر اطلاعات یک سفارش مثل زمان و تاریخ سفارش و شعبه اس که سفارش در آن انجام شده و مشتری که سفارش متعلق به اوست ذخیره شده است.

جدول orderItem نشان دهنده این است که هر سفارش شامل چه غذایی است و هر غذا چه تعدادی دارد و هزینه مربوط به آن غذا در سفارش مربوطه چقدر است. اگر هزینه هر غذا در یک سفارش محاسبه شود هزینه آن سفارش مشخص میشود.

```
create table orderinfo(  
  ID int not null identity(1000001,1) primary key,  
  order_date date,  
  order_time time,  
  customer_id int,  
  branch_id int,  
  foreign key (branch_id) references branch,  
  foreign key (customer_id) references customer);
```

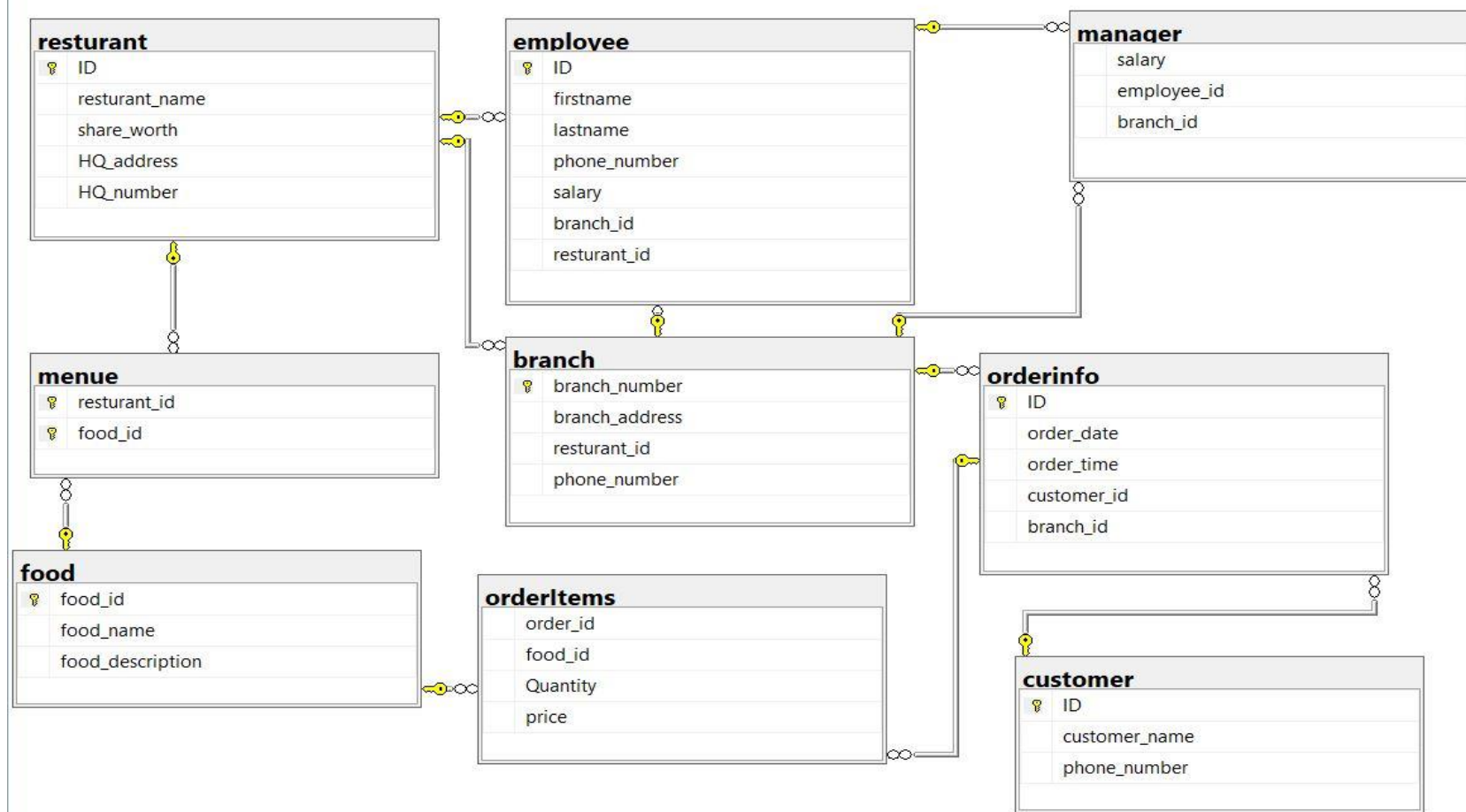
```
create table orderItems(  
  order_id int,  
  food_id int,  
  Quantity smallint,  
  price int,  
  foreign key (food_id) references food,  
  foreign key (order_id) references orderinfo);
```

مثالهایی از orderItem و orderinfo

	ID	order_date	order_time	customer_id	branch_id
▶	1000001	2023-06-28	12:25:36	1000001	100000
	1000002	2023-06-28	13:35:10	1000002	100003
	1000003	2023-05-13	14:23:58	1000003	100006
	1000004	2023-06-20	17:54:21	1000004	100009
	1000005	2023-08-21	19:52:12	1000005	100003
	1000006	2023-02-23	17:52:10	1000001	100003
	1000007	2022-03-10	14:10:56	1000001	100006
	1000008	2022-03-15	19:32:21	1000001	100009
	1000009	2022-08-10	13:23:12	1000002	100003
*	NULL	NULL	NULL	NULL	NULL

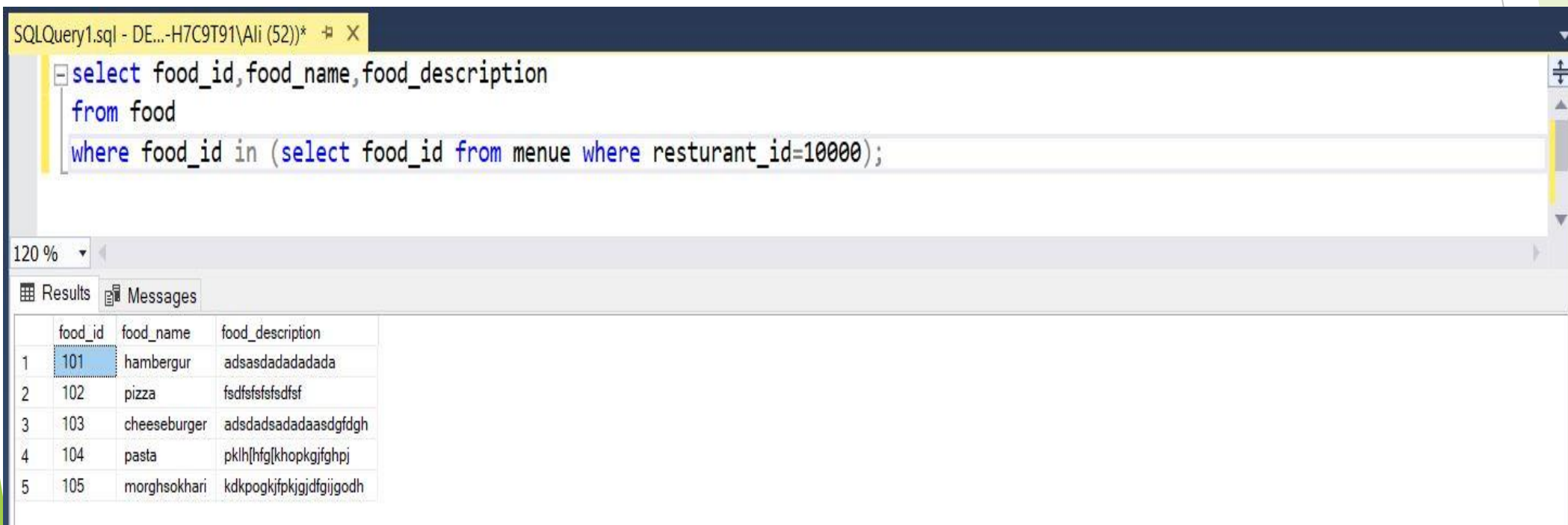
	order_id	food_id	Quantity	price
▶	1000001	101	2	120
	1000001	102	3	600
	1000001	103	1	150
	1000002	101	1	60
	1000002	104	2	600
	1000003	105	2	400
	1000004	106	3	900
	1000004	107	2	300
	1000005	106	1	300
	1000005	107	1	150
	1000005	108	2	500
	1000005	109	4	1600
	1000005	110	5	1000
	1000006	106	3	900
	1000006	107	3	450
	1000007	101	10	600
	1000008	104	5	1500
	1000008	102	4	800
	1000009	110	3	600
*	NULL	NULL	NULL	NULL

جداول و ارتباط های آنان



چند پرسمان

- این پرسمان ساده برای این است که مشخص شود که منو یک رستوران به خصوص شامل کدام غذا هاست.




The screenshot shows a SQL query editor window titled "SQLQuery1.sql - DE...-H7C9T91\Ali (52)". The query is as follows:

```
select food_id, food_name, food_description
from food
where food_id in (select food_id from menue where resturant_id=10000);
```

The query is executed, and the results are displayed in a table with 5 rows and 3 columns: food_id, food_name, and food_description. The first row is highlighted.

	food_id	food_name	food_description
1	101	hambegur	adsasdadadadada
2	102	pizza	fsdfsfsfsdfs
3	103	cheeseburger	adsdadsadadaasdgfdgh
4	104	pasta	pklh[hfg[khopkgjfgjhpj
5	105	morghsokhari	kdkpogkjfpkjgjdfigjgodh

پرسمان زیر برای پیدا کردن مشتری هست که از تمام شعبه ها خریداری کرده است به طوری که با یک زیرپرسمان ابتدا چک می کند که آیا سفارشی برای همه شعبه ها وجود ندارد مگر اینکه یک مشتری از انرا سفارش داده است.



```
select A.ID, A.customer_name
from customer as A
where not exists(
select B.branch_number
from branch B
where not exists (
select C.ID
from orderinfo as C
where C.customer_id = A.Id
and C.branch_id = B.branch_number));
```

120 %

Results Messages

	ID	customer_name
1	1000001	mohmad hamed

پرسمان زیر ابتدا جداول مشتری و orderinfo را بهم وصل کرده و ان را بر اساس ایدی مشتری و نام مشتری وایدی شعبه مرتب میکند. عبارت having نتایج را به ان محدود میکند که مشتریان در یک شعبه بیشتر از یک بار سفارش دادند. عبارت order_count تعداد دفعات سفارش را نشان میدهد.

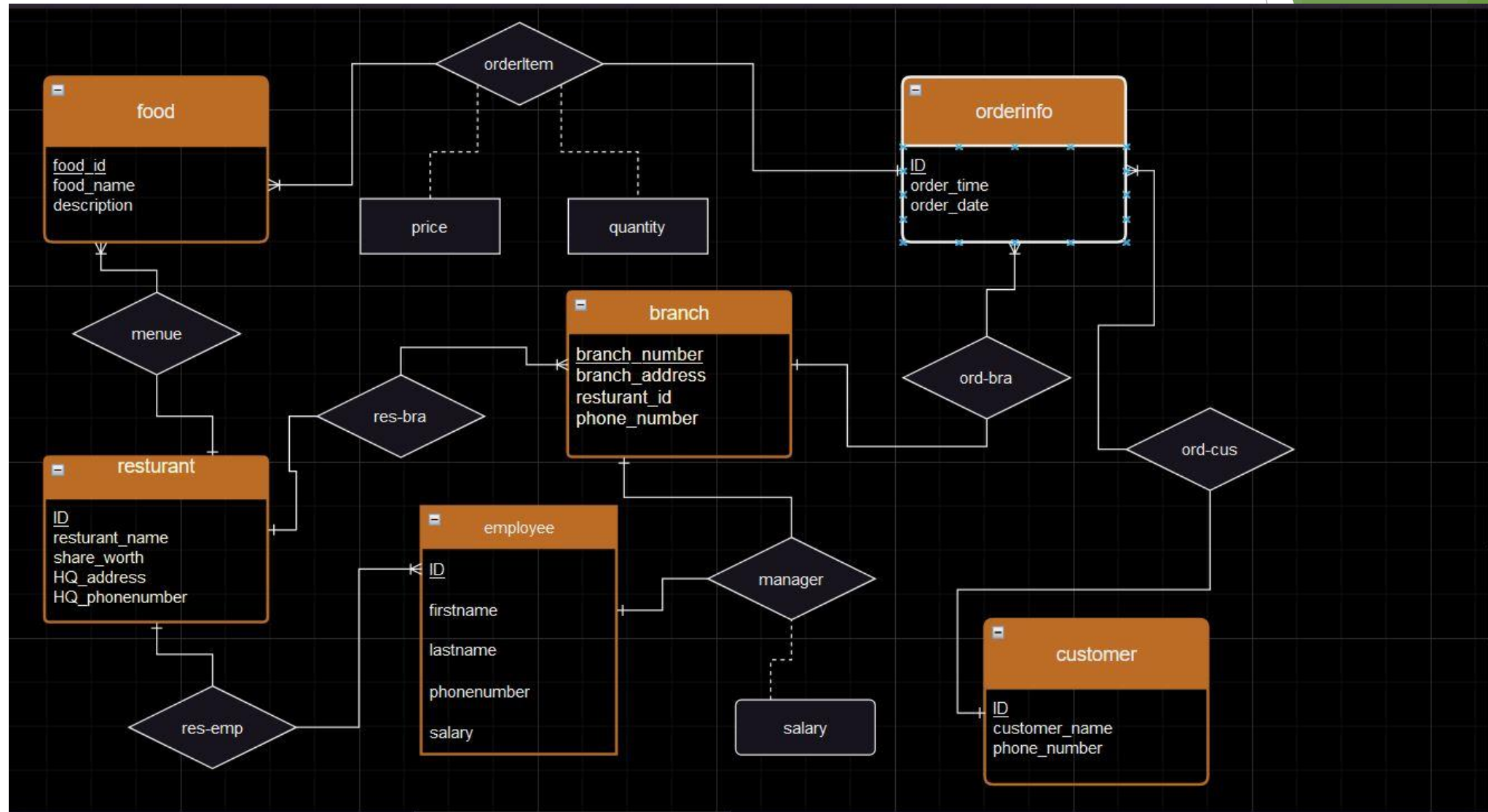
```
DESKTOP-H7C9T91\...rants - Diagram_0  SQLQuery1.sql - DE...-H7C9T91\Ali (52)) * X
select X.ID, X.customer_name, O.branch_id, count(O.ID) as order_count
from customer X join orderinfo O on X.ID = O.customer_id
group by X.ID, X.customer_name, O.branch_id
having count(O.ID)>1;
```

120 %

Results Messages

	ID	customer_name	branch_id	order_count
1	1000002	ali rezayi	100003	2

مدل ER



توضیحات مدل ER

- ▶ بین رستوران و غذا رابطه one to many وجود دارد چون یک رستوران شامل چند نوع غذا هست. این رابطه نامش menue است همانطور که در مثال ها مشاهده میشود در جدول ایدی یک رستوران همراه با چند غذا می تواند بیاید.
- ▶ بین رستوران و کارکن نیز رابطه one to many وجود دارد چون چند کارکن تنها می توانند در یک رستوران کار کنند. اگرچه ما می توانیم چند رستوران داشته باشیم اما هم چون مثال کتاب (رابطه department با student) رابطه one to many خواهد بود.
- ▶ همچنین رابطه one to many در رابطه رستوران و شعبه صدق میکند زیرا واضح است که هر رستوران چند شعبه می تواند داشته باشد.
- ▶ رابطه manager رابطه ای one to one میان شعبه و کارکن ایجاد میکند واضح است که هر شعبه تنها یک مدیر می تواند داشته باشد. salary به عنوان ویژگی مدیر آماده تا به هنگام تغییر در آن موجب راحتی شود.

- ▶ رابطه میان مشتری و سفارش (orderinfo) one to many هست زیرا هر مشتری می توان به تعداد زیادی سفارش داشته باشد.
- ▶ رابطه میان سفارش و شعبه نیز one to many است زیرا یک شعبه میتواند چندین سفارش داشته باشد ولی یک سفارش نمی تواند از چند شعبه باشد.
- ▶ رابطه food و orderinfo و رابطه orderItem نام دارد که شامل غذاهای حاضر در هر سفارش است. هم چنین ویژگی price و quantity نیز در آن قرار دارد که قبلا توضیح داده شد. این رابطه one to many هست چون هر سفارش میتواند چندین نوع غذا را شامل شود.