

به نام خدا

پایگاه داده رستوران

استاد: سلحشور

دانشجو: فائزه موسوی فیروزکلا (963112111)

دانشگاه صنعتی نوشیروانی بابل

فهرست مطالب:

3	توضیح در مورد محیط عملیاتی داده
4	جداول
4	جدول customers
4	جدول Agreement
5	جدول Order_Agreement
6	جدول Foods
7	جدول Personels
8	جدول Bills
10	جدول Cashiers
10	جدول های Other_bills
12	نمودار ER
15	کد های مربوط به پروژه
15	Selectes
16	Trigger
18	Function & Procedures
19	Transaction
20	Update & Delete

توضیح در مورد محیط عملیاتی داده:

در این سیستم که در رابطه با رستوران است ، اصلی ترین موضوع خریداری و سفارش غذا است. در این سیستم از query هایی جهت گزارش غذاهایی که موجود است ، درآمدی که از این طریق کسب میشود ، قرارداد هایی که منعقد میشوند، و با بررسی فاکتور های فروش میتوان به تعداد مشتریان و رضایت مندی مشتریان پی برد.

در این سیستم اطلاعات مشتریان ، جهت سهولت برای شناسایی و حتی ارسال بسته ثبت میشود. اطلاعات غذاها برای درک و آگاهی مشتریان از غذا ها جهت خرید و سفارش نگه داری میشوند. اطلاعاتی از قبیل قرارداد هایی که بسته میشوند، تاریخ قرارداد و... برای آگاهی از زمان تحویل و همچنین عدم تداخل قراردادها، و علاوه بر آن Query هایی جهت مدیریت این قرارداد ها استفاده میشود.

در این سیستم از صندوق دارانی جهت چاسخ گویی با صورت حساب های دریافتی جهت مدیریت مالی استفاده میشوند. سیستم شامل پرسنل هایی جهت خدمت گزاری است اطلاعات آن ها در سیستم ثبت میشود و به راحتی میتوان به اطلاعات آن ها دست یافت.

جدول:

جدول customers:

جدول مشتری شامل آیدی (Ccode) مخصوص به مشتری و نام (Cname) و نام خانوادگی (Cfamily) و شماره ملی (Ccm) و آدرس (Cadress) و شماره تلفن (Cphone) مشتری است. در این جدول آیدی مشتری (Ccode) کلید اصلی است و در این جدول اطلاعاتی که مربوط به مشتری است ثبت میشود و با جدول Agreement در ارتباط است.

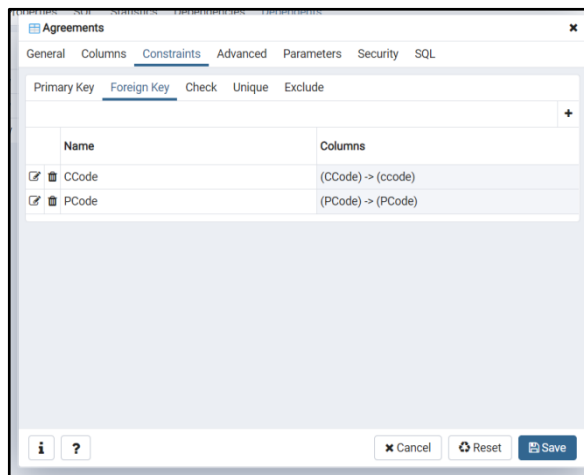
Name	Data type	Length	Precision	Not NULL?	Primary key?
ccode	integer			Yes	Yes
cname	character	40		No	No
cfamily	character	40		No	No
ccm	character	15		No	No
cadress	character	100		No	No
cphone	character	15		No	No

شکل 1_ ایجاد جدول مشتری

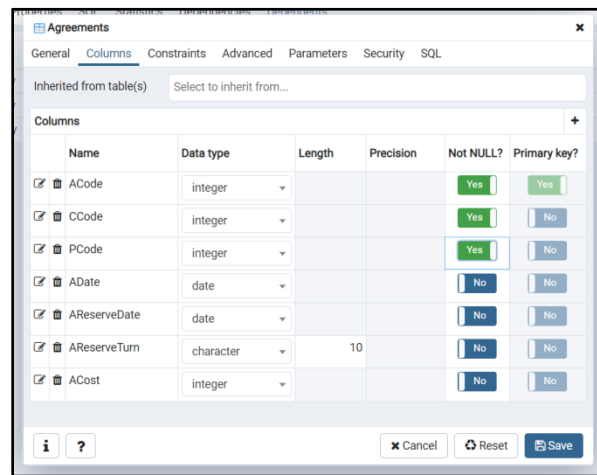
جدول Agreement:

در این جدول که جدول قرارداد است شامل آیدی مخصوص هر قرارداد (ACode) است که برای هر قرار داد منحصر به فرد است و شامل آیدی مخصوص هر مشتری ای (Ccode) که قرارداد بسته است، شامل آیدی پرسنلی (Pcode) که با این قرارداد هستند و شامل تاریخ قرارداد (Adate)، تاریخ رزرو قرارداد (Areservedate) و تاریخ فسخ قرارداد (AReserve turn) و مبلغ قابل پرداخت قرارداد (Acost) میباشد. در اینجا آیدی

هر قرار داد (ACode) کلید اصلی، آیدی هر مشتری (Ccode) و هر پرسنل (Pcode) کلید خارجی میباشد. جدول قرار داد با جدول مشتری (customer) برای ایجاد و ثبت قرار داد و همچنین با فاکتور فروش غذا (Order_Agreement) در ارتباط است.



شکل 2.2_ کلید های خارجی جدول قرارداد



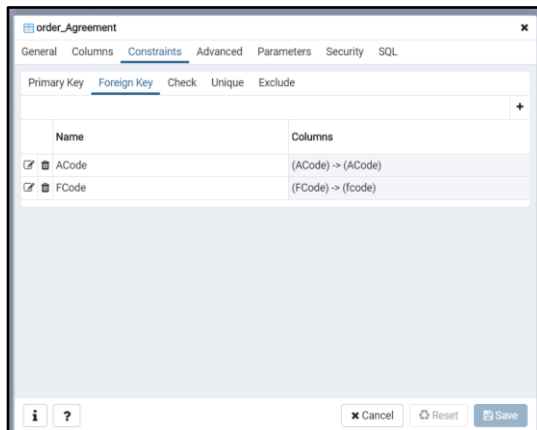
شکل 2.1_ ایجاد جدول قرارداد

جدول Order_Agreement:

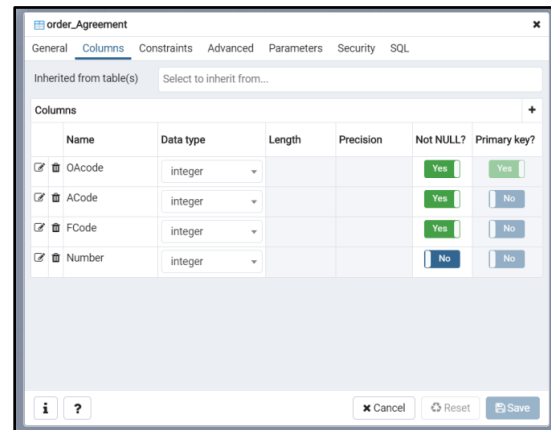
جدول فاکتور فروش غذا است که شامل آیدی مخصوص هر فاکتور فروش (OAcode) است که مختص هر فاکتور است و همچنین شامل آیدی قراردادی (Acode) است که این فاکتور شامل آن قرارداد میباشد و همچنین آیدی غذا هایی (Fcode) که برای قرارداد انتخاب میشود در این فاکتور نگه داری میشود و علاوه بر آن تعداد غذا هایی (Number) که برای سفارش انتخاب شدند در این فاکتور ثبت میشوند.

در این جدول آیدی هر فاکتور کلید اصلی است و آیدی قرارداد و آیدی غذا ها کلید های خارجی این جدول هستند .

این جدول با جدول Agreement برای ثبت قرار داد و با جدول Foods برای ثبت غذا و ثبت تعدا آن در ارتباط است.



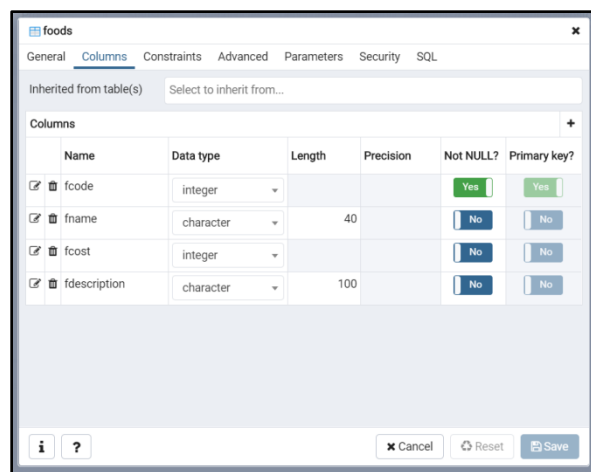
شکل 3.2_ کلید های خارجی جدول فاکتور فروش غذا



شکل 3.1_ ایجاد جدول فاکتور فروش غذا

جدول Foods:

جدول غذا از اصلی ترین موجودیت این سیستم است که عامل ورود مشتری به این سیستم است که با موجودیت های دیگر هم در ارتباط است. جدول غذا ها که شامل آیدی مخصوص هر غذا (Fcode) است که برای هر غذا منحصر به فرد است و شامل اسم غذا (Fname) و قیمت غذا (Fcost) و توضیحات غذا (Fdescription) است که نشان می دهد هر غذا از چه مواد غذایی تشکیل شده است. در این جدول اطلاعات هر غذا ثبت میشود و با جدول های فاکتور فروش غذا (Order_Agreement) که در آن از غذا های انتخابی ثبت میشود و با جدول صورت حساب غذا (Order_bills) در ارتباط است. در این جدول آیدی هر غذا (Fcode) کلید اصلی این جدول است.



شکل 4_ ایجاد جدول غذا

جدول Personels :

این جدول شامل کلید اصلی Pcode است که نشان دهنده ی آیدی هر پرسونل است و برای هر پرسونل متفاوت است در این جدول نام (PName) و نام خانوادگی پرسونل (PFamily) و محل سکونت (PAele) آن نیز ذخیره می شود .

این جدول با جداول قبض (Bills) و قرار داد (Agreements) در ارتباط است که برای اینکه قرار دادی صورت بگیرد باید اطلاعات پرسونل وجود داشته باشد .

Name	Data type	Length	Precision	Not NULL?	Primary key?
PCode	integer			Yes	Yes
PName	character	40		No	No
PFamily	character	40		No	No
PAle	character	40		No	No

شکل 5_ ایجاد جدول پرسونل

جدول Bills :

شامل کلید اصلی bcode و کلید های خارجی pcode و CAcode است.

در این جدول bcode به عنوان آیدی هر قبض و pcode به عنوان آیدی پرسونل هایی که قبض را دریافت کرده اند و Bdate به عنوان تاریخ صدور قبض و CAcode به عنوان آیدی صندوق دار و bcost به عنوان قیمت صورت حساب ذخیره می شود.

این جدول با 3 جدول پرسونل (Personels) و صورت حساب (orther_bills) و صندوق دار (cashiers) در ارتباط است. صورت حساب مربوط به غذاهای سفارش داده شده است که صندوق دار بعد از ایجاد صورت حساب قبض را تهیه میکند.

Properties SQL Statistics Dependencies Dependencies

Bills

General Columns Constraints Advanced Parameters Security SQL

Inherited from table(s) Select to inherit from...

Columns

	Name	Data type	Length	Precision	Not NULL?	Primary key?
<input checked="" type="checkbox"/>	bcode	integer			Yes	Yes
<input checked="" type="checkbox"/>	pcode	integer			No	No
<input checked="" type="checkbox"/>	bdate	date			No	No
<input checked="" type="checkbox"/>	CAcode	integer			No	No
<input checked="" type="checkbox"/>	bcost	integer			No	No

Cancel Reset Save

شکل 6.1_ ایجاد جدول Bills

Properties SQL Statistics Dependencies Dependencies

Bills

General Columns Constraints Advanced Parameters Security SQL

Primary Key Foreign Key Check Unique Exclude

Foreign Key

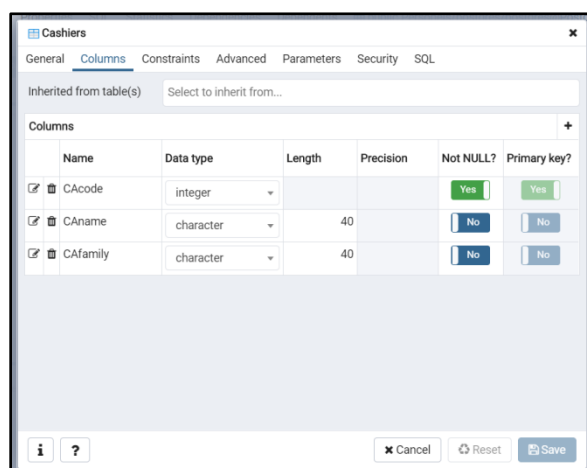
Name	Columns
CAcode	(CAcode) -> (CAcode)
pcode	(pcode) -> (PCode)

Cancel Reset Save

شکل 6.2_ کلید های فرعی جدول

جدول Cashiers :

شامل کلید اصلی CACode است.
این جدول دارای 3 پارامتر است که نام و نام خانوادگی صندوق دار و همچنین آیدی آن را نگه می دارد در این جدول CACode آیدی هر صندوق دار است.
این جدول با جدول قبض (Bills) در ارتباط است .



شکل 7_ ایجاد جدول صندوق دار

جدول های Other_bills:

شامل کلید اصلی OBcode و کلید های خارجی BCode و FCodde است.
در این جدول OBcode به عنوان آیدی صورت حساب است که برای هر صورت حساب متمایز از دیگری است و همچنین آیدی قبض ها (BCode) و آیدی غذاهایی (FCodde) را که انتخاب شده اند را نیز نگه داری میکند و آن تعداد غذا هایی (Number) که برای سفارش انتخاب شدند در این فاکتور ثبت میشوند.
این جدول دو جدول غذا (Foods) و قبض (Bills) را به هم متصل می کند. و با این دو جدول در ارتباط است.

order_bills

General Columns Constraints Advanced Parameters Security SQL

Inherited from table(s) Select to inherit from...

Columns

	Name	Data type	Length	Precision	Not NULL?	Primary key?
<input checked="" type="checkbox"/>	OBCode	integer			Yes	Yes
<input checked="" type="checkbox"/>	BCode	integer			Yes	No
<input checked="" type="checkbox"/>	FCode	integer			Yes	No
<input checked="" type="checkbox"/>	Number	integer			No	No

Cancel Reset Save

شکل 8.1_ ایجاد جدول صورت حساب

order_bills

General Columns Constraints Advanced Parameters Security SQL

Primary Key Foreign Key Check Unique Exclude

Name	Columns
<input checked="" type="checkbox"/> BCode	(BCode) -> (bcode)
<input checked="" type="checkbox"/> FCode	(FCode) -> (fcode)

Cancel Reset Save

شکل 8.2_ کلید فرعی جدول صورت حساب

نمودار ER :

برای رسم ERD ابتدا موجودیت هارا در نظر میگیریم. هر موجودیت قوی که حاوی کلید اصلی است به شکل چهارگوش نمایش داده میشود. که اسم موجودیت در بالای آن مینویسیم و حاوی صفات میباشد که زیر کلید اصلی آن خط کشیده شده است.

که ما دارای 8 مجموعه موجودیت قوی میباشیم. که بین موجودیت ها رابطه نیز برقرار میباشد. رابطه ای که از نوع یک باشد با فلش و اگر از نوع چند باشد با خط صاف نمایش داده ایم برای مثال:

رابطه بین personel و bills یک رابطه چند به یک است که زیرا هر personel چند فاکتور را میتواند بگیرد ولی هر فاکتور نزد یک پرسنل است. پس فلش به سمت پرسنل و خط صاف سمت فاکتور است.

رابط بین bills و cashiers یک به چند میباشد زیرا هر صندوق دار میتواند چند فاکتور و هر فاکتور میتواند یک صندوق دار داشته باشد.

رابطه بین فاکتو و سفارشات فاکتور یک به یک است زیرا متقابلن یک به یک نظیر میشوند.

رابطه بین سفارشات فاکتور و غذا چند به چند است. زیرا هر غذا ممکن است در چند سفارش وجود داشته باشد و برعکس هر سفارش میتواند چند غذا داشته باشد.

رابطه بین پرسنل و قرار داد یک به یک است چون هر قرار داد بین یک پرسنل بسته میشود.

رابطه بین قرارداد تالار و سفارشات قرارداد نیز چند به یک میباشد.

رابطه بین سفارشات قرار داد و غذا رابطه یک به چند است.

بین پرسنل و فاکتور رابطه ای وجود دارد به معنی صدور فاکتور که به صورت مخفف PB نمایش دادیم.

رابطه بین فاکتور و سفارشات فاکتور به معنی ثبت سفارش است که به صورت مخفف BO است.

رابطه بین صندوق دار و فاکتور به معنی پرداخت فاکتور است که به صورت مخفف BC است.

رابطه بین سفارشات فاکتور و غذا به معنی تهیه غذا است که به صورت مخفف OFa است.

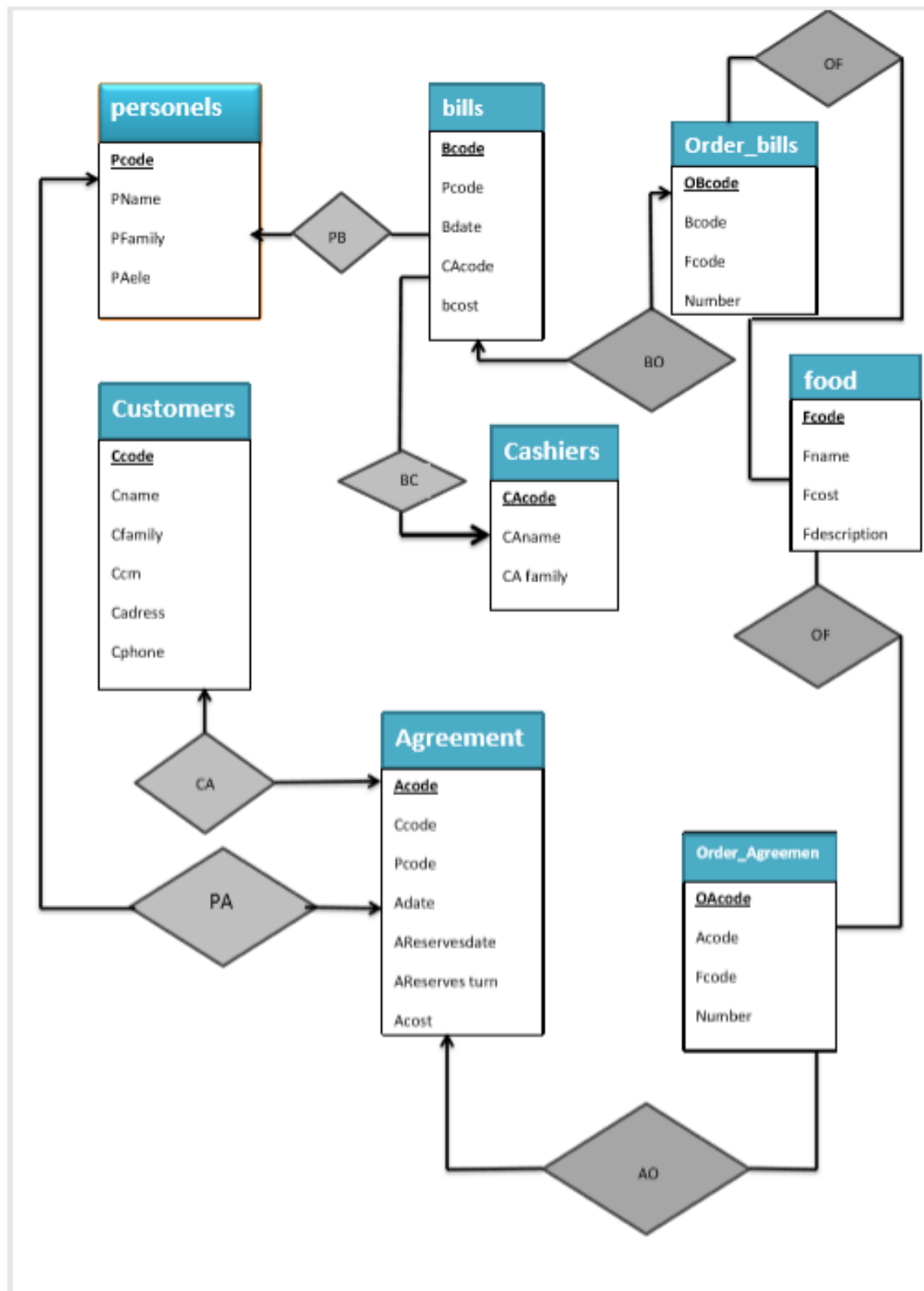
رابطه بین مشتری و قرار داد تالار به معنی عقد قرار داد است که به صورت مخفف CA است.

رابطه بین پرسنل و قرارداد تالار به معنی عقد قرارداد کاری است که به صورت مخفف PA است.

رابطه بین قرارداد تالار و سفارشات قرارداد به معنی ثبت قرارداد است که به صورت مخفف AO است.

رابطه بین سفارشات قرارداد و غذا به معنی ثبت قرارداد است و به صورت مخفف با Ofb است.

در شکل زیر نمودار ER را نمایش می دهیم.



کدهای مربوط به پروژه:

Selectes:

1. به ترتیب بیشترین، کم ترین، مجموع و میانگین قیمت قبض را میدهد.
2. مجموع قرار داد هایی که مشتری بسته است بر حسب فامیلی بر می گرداند.
3. اجتماع Bcode هایی که بین 20-1-2 و 20-3-12 و بین 20-3-10 و 20-3-11 هستند را برمیگرداند.
4. اسم و فامیلی مشتری ها را بر حسب شماره ملی ردیف می کند.
5. اسم پرسونل هایی که با مشتری هایی که اول اسم آنها t است و یا دومین حرف فامیلی آنها a است قرار داد بستن را برمیگرداند.
6. اسم غذا هایی که در آنها گوشت وجود دارد و هم در آنها نان وجود دارد.
7. فامیلی صندوق دار هایی را برمیگرداند که قبض آنها بیشتر از 300 است بجز آنهایی که تاریخ قبض آنها کوچک تر از 20-3-12 است.
8. اسم و کوچه پرسونل هایی که قیمت قرار داد هایی که میگیرن بیشتر از 200 باشد و همچنین مشتریان برای آنهايي که شماره ملی آنها با 5 شروع می شود.
9. شماره تلفن را زمانی که تاریخ کم تر از 20-4-9 باشد را بر می گرداند.
10. اسم غذاهایی که حداکثر یبار سفارش داده شده اند.
11. غذاهایی که در قبض نیامده اند.
12. اسم های متفاوت از غذا ها بر می گرداند که با جدول order_bills جوین شده اند.
13. نام و قیمت غذایی که قیمت آن از همه بیشتر است.
14. نام و کد صندوق دار هایی را برمی گرداند که در قبض شرکت نداشتند.
15. سطر هایی از جدول مشتری را بر میگرداند که شماره کد ملی بزرگ تر باشد از هر کسی که شماره کد ملیش از انهایی که مقدارش از 09366351302 بیشتر شود.

Trigger:

1. **تریگر upgradecustomers** که برای قبل از **insert** است به همراه تابع **upgradeCustomers()** فراخوانی میشوند که در تابع اسم و فامیلی مشتری را **space** اضافه را از آن حذف میکند و هم چنین اسم و فامیلی را به حروف بزرگ لاتین تبدیل میکند.

2. **تریگر upgardefoods** که برای قبل از عمل درج در جدول **foods** است و با تابع **upgardefoods()** فراخوانی میشوند که در آن شرط قیمت غذاهایی که از 500 کمتر باشد را 100 به قیمت جدید آن اضافه میکند و آن را در جدول قرار میدهد که با اجرای دستور

Insert into foods....

غذا با آیدی 11 و نام برگر و قیمت 500 و توضیحات گوشت و نان و سالاد به جدول **foods** اضافه میشود.

3. **تریگر customers_in** که برای قبل از عمل درج و یا به روز رسانی جدول **customers** است. که با تابع **in_outCustomers()** فراخوانی میشوند که در تابع اگر شماره تلفن مشتری یا شماره ملی با چیزی غیر از عدد وارد شود یا شده باشد را با **exception** ایی مبنی بر **'state must numeric'** مشخص میکند اگر طول شماره موبایل وارد شده یا شماره موبایلی که وارد میشود کمتر از 11 باشد یا شماره ملی طولش برابر 10 باشد **exception** ایی مبنی بر **'length is rong'** نشان می دهد.

با دستور اول **insert**

4. **تریگر foods** برای قبل از عمل درج یا بروز رسانی جدول **foods** به کار می رود که با تابع **in_up_food()** فراخوانی میشوند اگر اسم غذا با عدد باشد **exception** ایی مبنی بر **'state must character'** میدهد. با اجرای دستور **insert** که اسم غذا با عدد شروع شد پیام **exception** به عنوان خروجی برگردانده میشود.

5. تریگر Agreement که برای بعد از عمل insert با شرط قیمت قرار داد کمتر از 300 بر روی جدول Agreement است که با تابع Agreement() فراخوانی میشوند که اگر شرط موجود در تریگر درست باشد در تابع قیمت 10 تا اضافه میشود. با اجرای select جدول Agreement سطرهای با آیدی 44 و قیمت 40 به جدول اضافه شد.

Function & Procedures:

1. تابع `ft()` بررسی میکند که عبارتی که تمام مشخصات غذاهایی که قیمتشان کمتر از 400 و در اسم غذا حرف "آ" باشد را نشان میدهد آیا مقدار بازگشتی آن برابر تهی است یا خیر. اگر برابر تهی نباشد مشخصات را بر میگرداند. اما اگر برابر تهی باشد مشخصات غذاهایی که قیمتشان بیشتر از 500 باشد بر میگرداند. با اجرای این تابع حالت اول یعنی `select` اول را به عنوان خروجی میدهد.
2. تابع `chek_cost_food()` با دو آرگومان ورودی که آیدی غذا میباشد و قیمت دو غذای دریافتی را مقایسه میکند و بزرگتری، کوچکتری، مساوات قیمت هارا را نشان میدهد. در اینجا بین غذا با آیدی 3 و 4 مقایسه صورت گرفت که قیمت غذا با آیدی 3 از قیمت غذا با آیدی 4 کوچکتر است.
3. تابع `insert_food()` با آرگومان های ورودی که صفات جدول `foods` هستند عمل درج را انجام میدهد. با تست این تابع غذا با آیدی 10 و اسم قیمه و قیمت 700 با توضیحات لپه یک سطر جدید به جدول `insert foods` اضافه میشود.
4. رویه `asd()` با اجرای این رویه فامیلی و مجموع قیمت قرارداد های بسته شده پرسنل را نشام می دهد.
5. رویه `show()` که یک آرگومان ورودی آن ، آیدی جدول `Agreement` مشخص میکند و قیمت قرار داد با آن را 20 درصد کاهش میدهد و آن سطر از جدول را به روز میکند.

Transaction:

1. تراکنشی که جدول foods ها را به صورتی که با کسر مقدار 100 و از یک غذا با شماره آیدی مشخص و اضافه کردن 100 مقدار به ارزش غذایی با آیدی دیگر، به روز نگه میدارد.و برای برگشت به حالت قبل با تعریف یک نقطه امن میتوان به حالت قبل از بروز شده رسید.
2. تراکنشی که جدول Agreements ها را به صورتی که با کسر مقدار 10 و از یک قرار داد با شماره آیدی مشخص و اضافه کردن 10 مقدار به ارزش قرار داد با آیدی دیگر، به روز نگه میدارد.و برای برگشت به حالت قبل با تعریف یک نقطه امن میتوان به حالت قبل از بروز شده رسید.

Update & Delete:

1. کد پرسونلی که اسم آن سینا است 11 می شود.
2. شماره قبض را به 09365248485 تغییر میدهد برای کد اول.
3. اسم غذا را به مخصوص تغییر میدهد اگر قیمت آن بزرگ تر از 500 باشد و اسم آن با h شروع شود.
4. از جدول قرار داد کد داری 5 حذف می شود .
5. از جدول مشتری ، مشتری دارای کد ملی 2150527322 حذف می شود.
6. از جدول order_Agreements آن سطری که دارای شماره 09365689752 هست حذف می شود.