

## مدل رابطه ای

مدل رابطه ای متداول ترین مدل داده است که داده ها و ارتباطات بین آنها را به صورت مجموعه ای از جداول نمایش می دهد .

### اصطلاحات

#### کلید

#### خواص مدل

#### نرمالسازی

#### تبدیل نمودار ER به مدل رابطه ای

مدل رابطه ای (relational model) در سال ۱۹۷۰ توسط ریاضیدانی به نام Edgar.F.Codd طراحی شد. مدل داده پیشنهادی یک مدل منطقی بر مبنای ریاضیات است که از منطق گزاره ها و تئوری مجموعه ها به عنوان زیربنا استفاده شده است . یک پایگاه داده رابطه ای (relational database) پایگاه داده ای است که با مدل رابطه ای مطابقت داشته باشد و به صورت مجموعه ای از جدول هائی که از دید کاربر قابل درک هستند دیده می شود . یک سیستم مدیریت پایگاه داده رابطه ای (RDBMS) سیستمی است که داده را طبق مدل رابطه ای مدیریت می کند . RDBMS ها معمول ترین نوع سیستم های مدیریتی پایگاه داده امروزی هستند (نظیر Microsoft SQL Server ، Microsoft Access ، Oracle ، MySQL ، Sybase ، DB2 و Informix) . اکثر RDBMS ها SQL را به عنوان زبان پرس و جوی خود بکار می برند .

## اصطلاحات

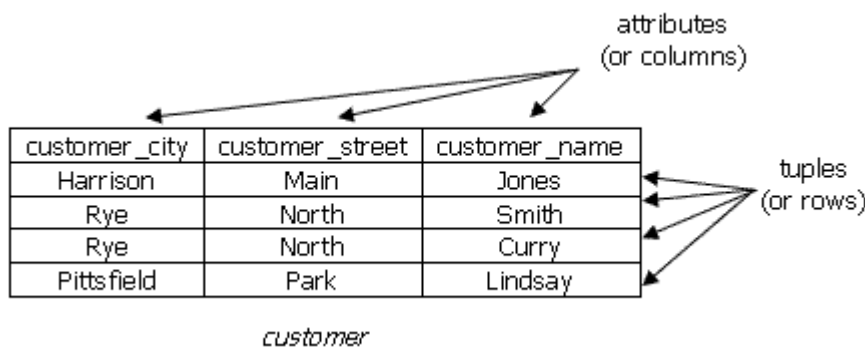
### جدول (رابطه)

پایگاه داده رابطه ای مجموعه ای از آرایه های دو بعدی است که جداول (table) یا رابطه (relation) نامیده می شوند. جدول مهمترین ساختار داده ای در سیستم پایگاه داده رابطه ای است . هر جدول (یا رابطه) داده ها را به صورت سطر ها و ستون ها شکل می دهد. هر سطر شامل یک نمونه منحصر بفرد داده و مربوط به یک نمونه موجودیت است. ستون ها صفات خاصه آن موجودیت را بیان می کنند . ترتیب سطر ها و ستون ها در جداول اهمیت ندارد. تعداد ستون های هر جدول را درجه (Degree) و تعداد سطر های آن را کاردینالیتی (Cardinality) می نامند .

### تاپل (رکورد)

یک سطر از یک رابطه را یک تاپل (tuple) می نامند. هر تاپل در جدول نمایانگر یک نمونه از یک موجودیت است که رکورد هم گفته می شود . تاپل ها ممکن است روی یکی از صفات خاصه به طور مرتب ذخیره شوند. ولی به طور کلی لازم نیست مرتب باشند .

مثال. رابطه Customer را در نظر بگیرید.



### فیلد (صفت خاصه)

هر ستون در جدول نشان دهنده يك صفت خاصه از يك نوع موجودیت است که فیلد (field) نامیده می شود. در هر فیلد نوع خاصی از داده ها ذخیره می شود . مقادیر هر سطر باید با فیلدهای جدول نظیر به نظیر باشد به عبارت دیگر ترتیب مقادیر فیلدها در همه رکوردها باید یکسان است. ولی ترتیب ستون ها اهمیت ندارد .

### دامنه

مجموعه ای از مقادیر مجاز یک ستون دامنه (domain) نام دارد. معمولا دامنه یک ستون از یک نوع داده است . محصولات RDBMS مستقیما دامنه را حمایت نمی کنند بلکه مجموعه ای از انواع داده عددی، متن، تاریخ و غیره را دارند که نحوه ذخیره سازی داده ها را مشخص می کنند. تاثیر دامنه را بیشتر می توان روی قیدها و مکانیسم های ورود داده مشاهده کرد . مقادیر صفات خاصه معمولا باید اتمیک باشند یعنی غیر قابل تفکیک باشند . مقدار خاص null عضوی از هر دامنه است .

### NULL

null یا <null> یک علامت خاص است که تهی بودن فیلدی را نشان می دهد، برای زمانی که مقداری برای فیلد وجود ندارد یا آنرا نمی دانیم استفاده می شود(برای مثال مشتری تلفن ندارد یا شماره آنرا نداریم). اکثر اوقات باید مشخص کنیم که آیا یک فیلد می تواند تهی باشد یا خیر . سعی در درج null در فیلدی که مجاز نیست ایجاد مشکل می کند . دانستن چگونگی برخورد RDBMS با null اهمیت دارد زیرا مقادیر تهی نمی توانند در عملیات داده ای شرکت کنند . بعضی از RDBMS ها null را اصلا ندارند. راه حل آنها تعریف یک ستون اضافی برای ستونی است که می تواند تهی باشد. این ستون معین می کند آیا ستون مرتبط به آن دارای مقدار معتبر است یا خیر . یک راه دیگر پر کردن فیلد با یک مقدار پیش فرض است. اگر مقدار پیش فرض قابل مشاهده ای وجود ندارد یا مشکل را حل نمی کند از null می توان استفاده کرد ولی باید مطمئن شد که در عملیات مسئله ساز نمی شود .

### پایگاه داده

یک پایگاه داده شامل چند جدول است. هر جدول بخشی از داده های سازمان را نمایش می دهد. ذخیره کلیه اطلاعات در یک جدول باعث تکرار داده یا نیاز به مقدار null می شود .

مثال. رابطه های محصول، مشتری و فروش را در یک سیستم فروش به صورت زیر تعریف می شوند:

PRODUCT				
ProductNo	Description	ReorderLevel	Price	QtyInStock
AQX88916	Flush widget	1000	23.95	4937
AAD62726	Left-handed doodad	0	4.95	10673
FGE91822	Grunge nut	5000	0.50	155016
AHL46785	Flange bracket	25000	1.35	51745
DHU69863	Self-righting socket	5	2.37	52

CUSTOMER				
CustomerNo	First	Last	Address	CreditLimit
4649-4673	Richard	Johnston	14 West Avenue	1000
1166-3461	Amelia	Waverley	995 Forth Street	<null>
7671-3496	Antonio	Gonzales	55B Granary Lane	<null>
6794-1674	Diane	Adams	364 East Road	1500
1113-7741	Wayne	Jones	42 York Street	<null>

SALE						
SaleNo	SaleDate	CustomerNo	ProductNo	Qty	Amount	Salesrep
12345	Aug 12 2002	4649-4673	AQX88916	1	23.95	Dave Williams
12346	Aug 12 2002	1113-7741	AQX88916	7	167.65	Sara Thompson
12347	Aug 13 2002	1166-3461	AHL46785	3705	5001.75	Li Qing
12348	Aug 13 2002	<null>	DHU69863	50	118.50	Sara Thompson

به جای رسم جدول به صورت فوق می توان رابطه ها را به شکل زیر تعریف کرد:

PRODUCT(ProductNo, Description, ReorderLevel, Price, QtyInStock)  
 CUSTOMER(CustomerNo, First, Last, Address, CreditLimit)  
 SALE(SaleNo, SaleDate, CustomerNo, ProductNo, Qty, Amount, Salesrep)

## کلید

در مدل رابطه ای هیچ دو سطری در جدول نباید مشابه باشند. این در واقع یک ویژگی اساسی جدول است. اگر دو سطر دو نمونه موجودیت متفاوت را در دنیای واقعی نشان دهند به نحوی باید از هم متمایز شوند تا به هر کدام در جدول بتوان جداگانه رجوع کرد. بنابراین حداقل یک مقدار منحصر به فرد باید وجود داشته باشد که باعث متمایز شدن سطرها از یکدیگر شود. ستونی که حاوی این مقدار است کلید نامیده می شود.

کلید داری دو خاصیت را باید دارا باشد؛ منحصر به فرد بودن و غیر تهی بودن (قانون اول جامعیت).  
 در یک رابطه انواع مختلفی از کلید ممکن است وجود داشته باشد:

- کلید کاندید
- کلید اصلی
- کلید ترکیبی
- کلید خارجی

### کلید کاندید

از مجموعه صفات خاصه يك رابطه آنهایی که دارای دو ویژگی زیر هستند به عنوان کلید کاندید (candid key) در رابطه مذکور مطرح می شوند:

- منحصر به فرد و غیر تهی بودن
- غیر قابل کاهش بودن، یعنی هیچ زیر مجموعه مناسبی از صفات خاصه تشکیل دهنده کلید، دارای خاصیت منحصر به فرد بودن نباشد.

مثال. شماره دانشجویی و کد ملی کلیدهای کاندید در جدول مشخصات دانشجو در دانشگاه می توانند باشند.

### کلید ترکیبی

کلید ترکیبی (compound key) کلیدی است که از ترکیب چند صفت خاصه ساخته می شود.

مثال. در رابطه دانشجو مجموعه نام و شماره شناسنامه می توانند به عنوان کلید ترکیبی در نظر گرفته شوند.

### کلید اصلی

کلید اصلی (primary key)، کلید کاندیدی است که توسط طراح پایگاه داده انتخاب و معرفی می شود. به عبارتی طراح بانک، یکی از کلیدهای کاندید را به عنوان کلید اصلی بر میگزیند.

در تعیین کلید اصلی از بین کلیدهای کاندید باید دو ضابطه زیر را در نظر گرفت:

- اهمیت کلید اصلی نسبت به سایر کلیدهای کاندید در پرس و جوها
- کوتاهتر بودن طول کلید کاندید از نظر تعداد بایت

نکته. هر جدول تنها یک کلید اولیه دارد اما به این معنی نیست که تنها یک شناسه منحصر به فرد دارد.

نکته. کلید می تواند صفات طبیعی موجودیت انتخاب شود، ولی اگر هیچ کدام از صفات خاصه مناسب نبودند یک کلید جانشین نسبت داده شود (مانند شماره کارمندی برای جدول کارمند).

نکته. در جدول، زیر کلید اولیه یک خط کشیده می شود.

نکته. اگرچه در مدل رابطه ای کلیه جداول باید دارای کلید اولیه باشند، ولی تعدادی از RDBMS ها اجباری در تعیین کلید برای هر رابطه نمی کنند، در اینصورت ترکیب کلیه صفات خاصه به عنوان کلید در نظر گرفته می شود.

مثال. شماره دانشجویی در جدول مشخصات دانشجو را می توان به عنوان کلید اصلی انتخاب کرد.

## کلید خارجی

کلید خارجی (foreign key) صفت خاصه ای در یک جدول است که در جدول دیگر نقش کلید اصلی یا کاندید را بازی کند. کلید خارجی ارتباط بین دو جدول را برقرار می کند. کلید خارجی بر خلاف کلید اصلی می تواند تکراری یا null باشد و ممکن است در یک جدول بیشتر از یک کلید خارجی وجود داشته باشد. جدول شامل کلید خارجی را گاهی جدول فرزند و جدولی که به آن ارجاع دارد را جدول والد می نامند.

مثال. شماره مشتری در جدول SALE کلید خارجی است زیرا در جدول CUSTOMER کلید اصلی است. شماره مشتری که در جدول فروش بدست می آید در جدول مشتری جستجو می شود تا تعیین شود محصول به کدام مشتری فروخته شده است.

## خواص مدل

مدل رابطه ای دارای ویژگی های زیر است:

- متداول ترین مدل است
- بر اساس تئوری ریاضی است
- داده ها و ارتباطات بین آنها در پایگاه داده به صورت مجموعه ای از جداول دیده می شود
- هیچ جدولی دارای سطرهای تکراری نیست.
- ترتیب سطرها و ستون ها در هر جدول مهم نیست
- ستون ها اتمیک هستند یعنی مقادیر ستون ها غیر قابل تجزیه اند.
- هر مقدار که در دو رکورد مختلف واقع می شود رابطه ای را بین دو آن رکورد می فهماند
- ارتباط رابطه ها با یکدیگر از طریق صفات خاصه مشترک انجام می گیرد.
- ایجاد، دسترسی و توسعه آن آسان است. بعد از ایجاد پایگاه داده اولیه، جداول جدید می توانند اضافه شوند بدون اینکه نیاز به تغییر کاربردهای موجود باشد.
- مدل دید کاربر است نه روشی که داده بطور داخلی سازماندهی می شود.

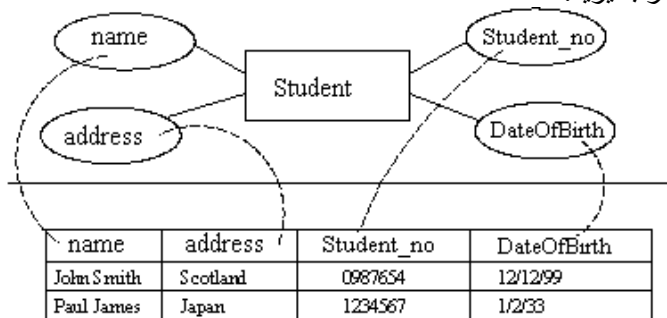
## نرمالسازی

نرمالسازی (normalization) با نحوه تقسیم جداول در پایگاه داده رابطه ای سروکار دارد. نرمالسازی فرآیند تغییر ساختار پایگاه داده به منظور اجرای بهتر و راحتی کارکردن با داده است. فرم های مختلف نرمالسازی که روی پایگاه داده اعمال می شوند را فرم های نرمال می نامند.

## تبدیل نمودار ER به مدل رابطه ای

هر موجودیت در مدل ER به یک جدول در مدل رابطه ای تبدیل می شود. صفات خاصه موجودیت ستون های جدول و هر نمونه موجودیت سطرهای آنرا می سازند. قبل از اینکه فرآیند تبدیل انجام شود باید مطمئن شویم که مدل ER تا حد ممکن ساده شده است.

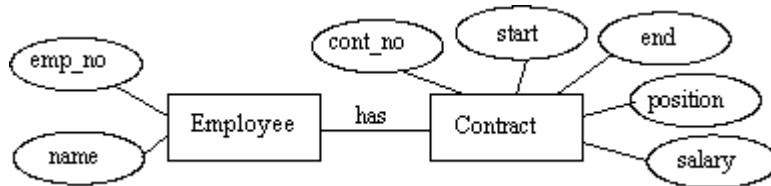
مثال. موجودیت دانشجو را در نظر بگیرید.



student(student\_no, name, address, date\_of\_birth)

برای تبدیل ارتباط یک به یک به رابطه، با توجه به اختیاری بودن یا اجباری بودن ارتباط، موجودیت ها یا ترکیب می شوند یا کلید اصلی یک موجودیت به عنوان کلید خارجی در دیگری قرار می گیرد.

مثال. ارتباط یک به یک بین موجودیت های کارمند و قرارداد را در نظر بگیرید. هر کارمند یک قرارداد دارد و هر قرارداد مربوط به یک کارمند است.



به رابطه زیر تبدیل می شود:

Employee(emp\_no, name, cont\_no, start, end, position, salary)

یا می تواند به صورت دو رابطه زیر تبدیل شود:

Employee (emp\_no, name, contract\_no)

Contract(cont\_no, start, end, position, salary)

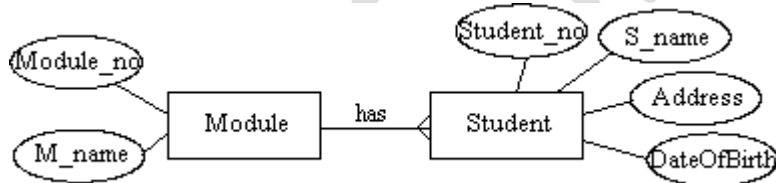
یا

Employee (emp\_no, name)

Contract(cont\_no, start, end, position, salary, emp\_no)

در تبدیل ارتباط ها یک به چند کلید اصلی موجودیت سمت "یک" به عنوان کلید خارجی در سمت دیگر قرار می گیرد.

مثال. ارتباط یک به چند دانشجو و رشته تحصیلی را در نظر بگیرید:



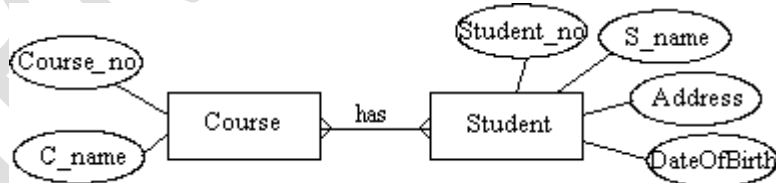
که به رابطه های زیر تبدیل می شود:

Student(student\_no, s\_name, address, DateOfBirth, module\_no)

Module(module\_no, m\_name)

در ارتباط چند به چند یک رابطه جدید با کلید های اولیه هر دو موجودیت ساخته می شود.

مثال. ارتباط دانشجو و درس را در نظر بگیرید.



که به رابطه های زیر تبدیل می شود:

Student(student\_no, s\_name, Address, DateOfBirth)

Course(Course\_no, C\_name)

Study(student\_no, Course\_no)