

تعریف محدودیت ها (constraints)

شما یک کلید اولیه و یک را برای جدول خود تعریف کرده اید. به هر حال کار ما تمام نشده و باید یکپارچگی محدودیت ها را ترکیب کنیم.

توضیحات

به نظر من اشتباه اصلی در طراحی ای است که تمام چک کردن داده را در داخل برنامه و ردیف کاربر انجام دهیم، در حالیکه دیتابیس و مکانیزم های آن برای برقراری تمامیت داده ها را نادیده می گیریم. از آنجایی که محدودیت ها برای موتور دیتابیس داخلی هستند، گزینه ی موثرتری برای چک کردن درستی می باشند. شما باید محدودیت های **CHECK, UNIQUE, DEFAULT, Primary** و **Key, Foreign Key** را برای محافظت از داده در مقابل ورودیهای نامعتبر تعریف کنید. با استفاده از محدودیت هایی که در ابزار مدلسازی داده نیز نمایش داده می شوند، می توان ورودیها را در داخل مدل داده ی خود بیشتر ثبت کرد.

داستانی که همیشه استفاده می کنم برای تکمیل بحث خود در مورد ترکیب محدودیت های داخلی (**native constraints**) حول یک تعامل مشاوره (**consulting engagement**) می چرخد که سال ها پیش به بررسی آن پرداختم. برنامه ی مشتری روی **Visual BASIC** نوشته شده بود و چک کردن همه ی داده ها در داخل کد این برنامه انجام شد. متأسفانه خود دیتابیس کم و بیش به عنوان محل ذخیره سازی تعریف شده بود. وارد کردن یک داده از یک سیستم گروه سوم لازم بود و یک وارد کننده ی جدید با استفاده از برنامه ی **VB**، به عنوان پایگاه (زیرا آنجا تنها مکانی است که همه ی قوانین داده ها قرار می گرفتند) نوشته شد. اطمینان کافی داریم که برخی قوانین از بین رفتند و نتیجه ی ورودی دیتابیس را با داده ی نامعتبر غربال کرد.

هنگام نفوذ به یکپارچگی محدودیت ها (**leveraging constraints integrity**) همیشه آنها را نامگذاری کنید. اگر **SQL Server** مجاز است که به طور خودکار نام محدودیت تولید کند، ممکن است از دیتابیزی به دیتابیس دیگر متفاوت باشد. اگر تعریف از یک محدودیت باید تنظیم شود، منطق خاص در مقابل ویوهای سیستم باید نوشته شود تا آنچه را که نام تولید شده به آن اختصاص داده شد، شکل دهد. اگر محدودیت های شما در جلو نامگذاری شده اند، کد گذاری خاص در اینجا جلوگیری می شود علاوه بر اینکه از یک نصب به نصب دیگر به دیتابیس ها فضای استحکام می بخشد.

در این مثال دودیتابیس ایجاد کرده و به هر دو یک جدول یکسان افزوده ایم. محدودیت ها را با عنوان **UNIQUE** و **DEFAULT** نامگذاری نمی کنیم.

```
create database db1
go
use db1
go
create table t1 (value int not null default 1)
create table t2 (value int not null unique)
go
```

آدرس آموزشگاه : تهران - خیابان شریعتی - بالا تر از خیابان ملک - جنب بانک صادرات - پلاک 651 طبقه دوم - واحد 7

88146323 - 88446780 - 88146330

```
create database db2
go
use db2
go
create table t2 (value int not null unique)
create table t1 (value int not null default 1)
go
```

اجازه بدهید محدودیت ها را در داخل ویوهای سیستم بررسی کنیم.

```
use db1
go
select db_name() as dbname, name from sys.default_constraints
select db_name() as dbname, name from sys.key_constraints
go
use db2
go
select db_name() as dbname, name from sys.default_constraints
select db_name() as dbname, name from sys.key_constraints
go
```

آموزشگاه تحلیلیکرو داده ها

آدرس آموزشگاه : تهران - خیابان شریعتی - بالا تر از خیابان ملک - جنب بانک صادرات - پلاک 651 طبقه دوم - واحد 7

88146323 - 88446780 - 88146330

<http://www.tahlildadeh.com/>

Results		Messages	
dbname	name	dbname	name
1	db1	DF_t1_value_7E6CC920	
1	db1	UQ_t2_40BBEA3A014935CB	
1	db2	DF_t1_value_023D5A04	
1	db2	UQ_t2_40BBEA3A7F60ED59	

نام محدودیت ها با نام دیتابیس ها یکی نیست که این امر باعث می شود هر عملکردی آنها را خسته کننده کند. محدودیت های خود را نام گذاری کنید.

آموزشگاه تحلیگر داده ها

آدرس آموزشگاه : تهران - خیابان شریعتی - بالا تر از خیابان ملک - جنب بانک صادرات - پلاک 651 طبقه دوم - واحد 7

88146323 - 88446780 - 88146330

<http://www.tahlildadeh.com/>