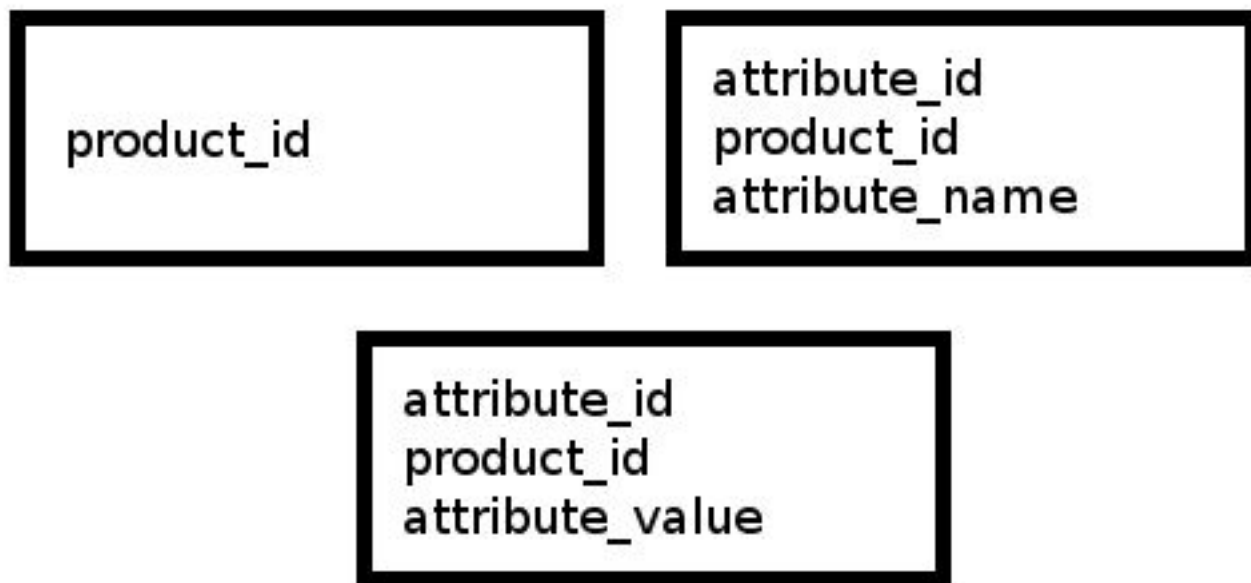


EAV مخفف ( Entity Attribute Value ) می‌باشد، مدلی از طراحی دیتابیس که کاربر را به آیتم‌های ثابت محدود نمی‌کند، فرض کنید در یک فروشگاه می‌خواهید چندین کالا بفروشید هر کالا هم برای خودش ویژگی‌های منحصر به فرد دارد، آیا با ویژگی‌های ثابت برای کالاهای متفاوت می‌توان پاسخگوی نیاز مشتری بود؟ یقیناً پاسخ منفی خواهد بود. موجودیت ( Entity ): در یک سیستم می‌تواند کالا، مشتری، فروشنده و... باشد. ویژگی ( Attribute ): برای کالا: رنگ، وزن و... برای مشتری: نام، تلفن، آدرس و... می‌باشد مقدار ( value ) : هر ویژگی برابر مقداری می‌باشد مثلاً برای رنگ‌ها آبی، قرمز و... می‌باشد جداول پایه طراحی شده:



ورود داده ها:

شیوه ورود داده‌ها را برای موجودیت کالا بیان می‌کنیم  
ابتدا کالا در جدول موجودیت ثبت می‌گردد  
سپس عنوان ویژگی‌های آن مانند رنگ، وزن و... در جدول ویژگی‌ها ثبت می‌گردد.  
مقدار هر ویژگی هم در جدول مقدارها ثبت می‌شود.  
در زیر شیوه ذخیره به صورت شکل مشاهده می‌کنید.



شیوه خواندن داده ها:

این قسمت هم به راحتی با 2 inner join می توان به کالا، ویژگی ها و مقادیر آن دست پیدا کرد.

نکات:

نکته 1: این 3 جدول را باید برای هر موجودیت قابل توسعه ایجاد کرد، مثلا برای کالا، مشتری و...

نکته 2: می توان برای گروه بندی کالاها و همچنین ویژگی ها جداول جداگانه ایی تعریف کرد.

نکته 3: از مهمترین ویژگی های این تفکر قابل گسترش بودن سیستم می باشد.

نکته 4: می توان برای آیتم هایی مثل نمایش داده شود یا خیر، چیدمان نمایش و... آیتم هایی به جدول ویژگی ها اضافه کرد.

نکته 5: این مدل در نرم افزار magento استفاده شده است.

همچنین جهت مطالعه بیشتر ساختار دیتابیس مجنتو در لینک زیر می باشد.

[MAGENTO\\_v1.0.19700---Database-Diagram.zip](#)

منابع: [Entity-attribute-value model](#)

## نظرات خوانندگان

نویسنده: افشار محبی  
تاریخ: ۱۳۹۲/۰۹/۰۱ ۲۳:۵۰

روش مفید و موثری است. ولی نمی‌دانم با داده‌های حجم بالا هم می‌توان خوب کار کند یا نه. علاوه بر این روش‌های متعارف query روی دیتابیس را به چالش می‌کشد. به عنوان مثال دیگر نمی‌توان با یک Query معمولی فهرست کالاها و مشخصات و گروه بندی بر اساس فلان Attribute را استخراج کرد.

نویسنده: حسین صفدری  
تاریخ: ۱۳۹۲/۰۹/۰۲ ۸:۳۴

اگر جدول را خوب ایندکس گذاری کنیم مشکل سرعت حل می‌شود از طرفی نحوه نمایش هم باید بهینه باشد، مثلا 10 تا 10 نمایش داده دهیم و...  
اتفاقا قسمت قشنگ ماجرا اینجا می‌شود که شما تمام Attribute‌های یک گروه کالا را به کاربر نمایش می‌دهید و بنا به درخواست کاربر داده‌های آن ویژگی را می‌توان نمایش داد.

نویسنده: سجاد ف  
تاریخ: ۱۳۹۲/۰۹/۰۲ ۸:۵۱

میشه توضیح بدین اگه چند Attribute با نوع مختلف داشتیم چطور ذخیره کنیم ؟ نوع Attribute Value باید string باشه ؟

نویسنده: vahid  
تاریخ: ۱۳۹۲/۰۹/۰۲ ۸:۵۸

باسلام  
سایت ebay بگونه ای طراحی شده که وقتی ما کالای را انتخاب میکنیم میتوانیم آن را با ویژگی‌ها خاص خود آن کالا آن را فیلتر نماییم و همان زمان تعداد موجودی آن کالا و قسمت نسبت به ویژگی‌ها و ... را می‌دهد چگونه به این سرعت عمل میکند؟ با آن حجم اطلاعات؟ ممنون از شما

نویسنده: محسن خان  
تاریخ: ۱۳۹۲/۰۹/۰۲ ۹:۳۳

[چه زمانی بهتر است از بانک‌های اطلاعاتی NoSQL استفاده کرد و چه زمانی خیر؟](#)

نویسنده: حسین صفدری  
تاریخ: ۱۳۹۲/۰۹/۰۲ ۱۴:۴۵

طوری که مجتو طراحی کرده است مطابق فایل ضمیمه برای هر type یک مجموعه جدول طراحی کرده است. فکر می‌کنم این شیوه بهینه می‌باشد.

نویسنده: حسین صفدری  
تاریخ: ۱۳۹۲/۰۹/۰۲ ۱۴:۴۷

آن طور که بنده اطلاع دارم، در دیتا بیس‌های NoSQL برای سیستم هایی با حجم داده‌های بزرگ قابل استفاده است (چند صد میلیون)، و شیوه طراحی آن به گونه ایی است که هرچه داده‌ها بیشتر شوند سرعت جستجو هم مطابق تابع نمایی بالا می‌رود.

نویسنده: حسین صفدری  
تاریخ: ۱۳۹۲/۰۹/۰۲ ۲۳:۴۱

سوال قشنگی بود، طوری که امروز کلی تو سایت‌ها راجع بهش مطلب خوندم.. اگر شما ویژگی‌های کالا را طبق مدل بالا انجام دهید و آن‌ها را گروه بندی کنید می‌توانید به کاربر نمایش دهید و از طرفی به ازای افزودن هر فیلتر در عبارت شرطی خود AND و ویژگی و مقدار آن را می‌آوریم.

اما ebay! طراحی دیتابیس سایت ebay منحصر به فرد است، همچنین طبق [اینجا](#) قسمت search suggestion and the internal Cloud Manager از MongoDB که بر پایه NO SQL می‌باشد کار می‌کند. همچنین جهت آشنایی بیشتر با ebay این [لینک](#) را ببینید.

نویسنده: افشار محبی  
تاریخ: ۱۰:۲۶ ۱۳۹۲/۰۹/۰۳

چرا توی هر موضوع جدیدی ذهن بیشتر افراد فوراً منحرف می‌شود به سمت Amazon و eBay و فیسبوک و بقیه غول‌ها؟ مگه ما تو ایران چند تا از این غول‌ها داریم یا خواهیم داشت؟ بیشترین نیاز نرم‌افزارها و مشاغل مختلف در ایران نیازهای کوچک و متوسط هستند.

کاش اینقدر غیر عملی و غیر کاربردی فکر نمی‌کردیم. در داخل کشور بیشتر از این که نیاز به ارسال ماهواره به فضا داشته باشیم نیاز به داشتن یک خودروی معمولی داریم که سالی چند هزار کشته و زخمی جاده و خیابان‌ها نداشته باشیم.

بیخشید که از موضوع خارج شدم.

نویسنده: محسن خان  
تاریخ: ۱۱:۱۶ ۱۳۹۲/۰۹/۰۳

این روحیه شما جستجوگری را از بین می‌برد. تفکر در مورد راه‌های مختلف را منع می‌کند. اقلان به روش‌های عهد عتیق طراحی را که الزاما بهینه نیستند، ترویج می‌کند. جستجوی در مورد راه‌های NoSQL الزاما به معنای استفاده از آن‌ها نیست ولی حداقل دید شخص را نسبت به الگوریتم‌ها و طرز تفکرهای مختلف موجود جهت حل مسایل باز می‌کند. خیلی‌ها مثلا جبهه می‌گیرند در مورد ORM. به این افراد باید گفت، اشکالی نداره. استفاده نکنید. حداقل طراحی اون‌ها رو مطالعه کنید که توسط بزرگان دنیا انجام شده و ارزش درس یاد بگیرید تا کدهای SQL Helper مشکل داری رو طراحی نکنید. برید SQL بنویسید بجای LINQ. اما حداقل یادگیری اونی که اومده لایه DAL جنریک درست کرده، طراحی‌اش چطوری بوده. دو تا نکته ارزش یاد بگیرید. نمی‌خواهین با MVC کار کنید، مهم نیست. حداقل طراحی‌اش رو بررسی کنید که چطور تونسته ViewState رو حذف کنه اما باز هم بعد از post back به سرور می‌تونه مقادیر وارد شده در فرم‌ها رو در صورت نیاز حفظ کنه.

مورد دوم اینکه اون تعداد کشته‌ها ربط مستقیم داره به میزان بی‌سوادی در کشور. [مطابق نظر معاون وزیر آموزش و پرورش](#) در سال قبل «در کشور نزدیک به 9 میلیون و 700 هزار نفر خواندن و نوشتن بلد نیستند و بیش از 10 میلیون نفر نیز تحصیلات حداکثر پنجم ابتدایی دارند.» خوب این‌ها مسلما مشکل‌زا هستند. همه چیز تقصیر ماشین و جاده نیست. این‌ها هم کسانی هستند که قانع هستند به آنچه که دارند و نیازی برای پیشرفت حس نمی‌کنند.

نویسنده: افشار محبی  
تاریخ: ۱۱:۴۱ ۱۳۹۲/۰۹/۰۳

منظور من این نیست که از روش‌ها و فناوری‌ها به روز و جدید استفاده نکنیم. درست برعکس. هر چه به روزتر باشیم بهتر است. نظر من این است که به کاربردهایی که در کشور داریم فکر کنیم نه چیزی که شاید فقط در آمریکا کاربرد داشته باشد. به عبارتی دیگر از روش‌ها و فناوری‌های روز استفاده کنیم منتها برای نرم‌افزارهایی که در داخل کشور کاربرد دارند نه فیسبوک و امثالهم.

اون قدر تمرکز نکنیم روی مدل دیتابیس فیسبوک که از دیتابیس‌های ساده مورد نیاز نرم‌افزارهای خودمان غافل شده و قدرت کار

بهینه روی آن را از دست بدهیم. مثال می‌زنم. در جمع‌های برنامه‌نویسی چه در دانشگاه‌ها، چه در شرکت‌ها چه در فضای اینترنت وقتی صحبت از دیتابیس می‌شود همه طوری حرف می‌زنند که انگار طراحی دیتابیس فیسبوک و امثال آن را فوت آب هستند اما وقتی پای عمل می‌رسد در عوض کردن دیتابیس یک نرم‌افزار کوچک از MS SQL به Express یا CE یا SQLite از زمین تا آسمان مشکل دارند و نرم‌افزاری که تولید کرده‌اند فقط روی یکی دو مدل ویندوز با یک سری config‌های خیلی محدود قابل اجرا هستند. بخشی از این مشکل به خاطر عدم تمرکز روی کاربردهای رایج و همه روزه از دیتابیس، کامپیوتر، نرم‌افزار، فناوری و غیره است.

البته و صد البته مطالعه و بررسی مواردی که حتی در ایران قابلیت اجرا ندارند خیلی هم خوب است و دیدگاه‌های جدیدی به آدم می‌دهد. منتها اول باید به عنوان یک مهندس نرم‌افزار خوب به وظایفمان که تولید و راهبری سیستم‌های نرم‌افزاری در حیطه کشور است عمل کنیم سپس به سراغ چنین مباحثی برویم.

ذهن من پر مثال است. در حالی که استاندارد ساختمان سازی در ایران بسیار پایین است و عمر ساختمان در حد ۳۰ سال است (یک چندم استاندارد جهانی)، مقاومت زیادی در برابر زلزله ندارند و هزار ایراد دیگر، آن وقت میایم دست به ساختن برج میلاد می‌زنیم. اگر واقعاً توان مهندسی داریم اول آن را جاهایی که ضروری‌تر است مصرف کنیم سپس جاهایی که اسم گنده دارند و برایمان «رزومه» می‌شود.

نویسنده: Programmer  
تاریخ: ۱۵:۱۳ ۱۳۹۲/۰۹/۰۳

اگر امکانش هست یک مثال از کد SQL مربوطه بزارید ممنون میشم

نویسنده: محسن خان  
تاریخ: ۱۷:۱۴ ۱۳۹۲/۰۹/۰۳

فایل MAGENTO v1.0.19700 - Database Schema [SpacedFormat].sql داخل فایل zip پیوست شده هست.

نویسنده: Programmer  
تاریخ: ۱۰:۲۱ ۱۳۹۲/۰۹/۰۴

البته منظورم دستور SELECT همراه با INNER JOIN هستش. میخوام ببینم آیا میشه کل تولیدات رو با تمام خواص انتخاب کنم؟ بعد برای ستون بفرض رنگ که بعضی تولیدات چنین ویژگی ای ندارند چه چیزی میاد؟

نویسنده: محمد  
تاریخ: ۱۰:۲۶ ۱۳۹۲/۰۹/۰۶

سلام

این مدل طراحی من رو به یاد کتاب The Data Model Resource می‌اندازه به نظرم کتاب خوب و فوق العاده ای برای طراحی انواع پایگاه‌های داده برای پروژه‌های خاص البته vol 1,2,3 داره من فقط کتاب یک و دو رو دیدم .  
ممنون از مطلب خوبتون

نویسنده: سعید  
تاریخ: ۱۳:۵۳ ۱۳۹۲/۱۰/۱۳

سلام

سوال اولم اینه که چه نیازی به کد محصول در جدول مقادیرها میباشد ؟ مگه کد محصول تو جدول ویژگی‌ها نیست؟! و از اون طرفی هر ویژگی یه کد منحصر به فرد داره (attribute\_id) که این کد نیز در جدول ویژگی‌ها وجود دارد . پس با Join کردن موجودیت و ویژگی توسط کد موجودیت و Join کردن نتیجه با جدول مقدار توسط کد ویژگی میتوان به تمام مقادیر یک موجودیت دست یافت از اونجایی که فکر کردم سوالم مربوط به این مورد هست میپرسم . در نرم افزارهای حسابداری ما تفصیلی‌ها رو گروه بندی میکنیم مثل گروه اشخاص ، کالا ، سهامداران ، بانک و ...

و هر کدام از این گروه‌ها دارای مقادیر خاصی هستند که در گروه دیگر وجود ندارد مثل کد فنی کالا و ...  
حالا سوالم اینه که به نظر شما برای این مورد از این روش مشه استفاده کرد ؟  
یعنی:

یک جدول برای تفصیلی ها  
یک جدول برای ویژگی‌های تفصیلی  
یک جدول برای مقادیر ویژگی ها  
یک جدول برای گروه‌های تفصیلی  
یک جدول هم برای ارتباط گروه تفصیلی با خود تفصیل (جهت مشخص کردن تفصیلی‌های موجود در یک گروه)  
با تشکر

نویسنده: حسین صفدری  
تاریخ: ۲۲:۴۵ ۱۳۹۲/۱۰/۱۳

سلام

در مورد سیستم حسابداری متاسفانه به صورت حرفه ایی بنده کار نکردم..  
این مدل معایبی هم دارد که selectهای پیچیده ایی باید نوشت، که روی عملکرد سیستم بسیار تاثیر می‌گذارد.  
بنده در این مقاله چون دیتابیس مجنتو را مطالعه کردم و با مدلی جدید آشنا شدم آن را با شما دوستان عزیز به اشتراک گذاشتم...