# فیلدهای از نوع انتخابگر حرفی

در تعریف و تدوین فیلدهای فرایندها، در بعضی از موارد به فیلدهایی برمی خوریم که اطلاعات این فیلدها قبلا در یک پایگاه داده ذخیره شده و وجود دارند. برای مثال در یک فرایند فروش نیاز داریم اطلاعات مشتری (نام ، کد، آدرس، شماره تلفن ، ...) را داشته باشیم که این اطلاعات قبلا و در سیستم مالی ذخیره شده و وجود دارد. یا مثلا در یک فرایند تدارکات و تامین، یکی از اصلی ترین اطلاعات، نام و کد کالاهاست که این داده ها هم قبلا در سیستم انبار مالی وارد و ذخیره شده است. حال برای تعیین و ثبت این داده ها، دو راه حل وجود دارد:

## راه حل 1: تعریف یک فیلد از نوع حرفی و وارد کردن اطلاعات به شکل دستی

واضح است که این روش علاوه بر دوباره کاری و ورود اطلاعات تکراری و چندگانه، به دلیل احتمال خطای انسانی دقیق و قابل اعتماد نیست. به عبارت دیگر کاربر در هنگام وارد کردن نام کالا ممکن است اشتباه تایپی داشته باشد و یا اینکه نام یک کالا را به صورتهای گوناگون ثبت کند. مثلا یک کالا را یکبار "مانیتور" و بار دیگر "مونیتور" ثبت کند یا مثلا یکبار نام یک مشتری را "شرکت نرم افزار و سخت افزار ایران" و بار دیگر شکل مخفف آن "نوسا" را وارد کند و لذا به دلایلی که گفته شد، این اطلاعات یکدست و معتبر نخواهد بود.

## راه حل 2: امکان ملاحظه فهرست داده های موجود در پایگاه داده مبدا و جستجو از این فهرست

برای تحقق این راه حل، در سیستم فرایندها، فیلدهای از نوع انتخابگر حرفی پیاده سازی شده است. در این نوع از فیلدها با استفاده از امکان View از SQL Server ، و از طریق یک Query مشخص از پایگاه داده مبدا (در اینجا سیستم مالی)، یک جدول مجازی تولید می شود که شامل اطلاعات مورد نظر (در مثالهای بالا ، نام کالا، نام مشتری) می باشد. این جدول در یک محاوره به کاربر نمایش داده می شود و کاربر می تواند در این فهرست، کالای مورد نظر خود را جستجو و بازیابی کند.

## مزایای استفاده از فیلدهای انتخابگر حرفی

**عدم نگهداری اطلاعات تکراری :** طبق یک اصل کلی، اطلاعات نباید در پایگاه داده های متعدد تکرار شود. اطلاعات تکراری علاوه بر تحمیل هزینه برای ورود مکرر اطلاعات، احتمال مغایرت در داده ها را بالا برده و به سرعت اعتبار داده ها را کاهش می دهد.

**یکدستی اطلاعات:** در این روش، چون اطلاعات در یکجا نگهداری می شود لذا اصالت و یکدستی داده ها حفظ می شود.

**بروز بودن اطلاعات:** از آنجایی که اطلاعات از مبدا نمایش داده می شود، اطلاعات همیشه بروز است و در هر لحظه، آخرین نسخه اطلاعات نمایش داده می شود. برای مثال اگر نام یک کالا در سیستم انبار تغییر کند، در سیستم مدیریت فرایندها هم همین نام تغییریافته ثبت خواهد شد.

حال برای آشنایی بیشتر با این مفهوم، بهتر است ابتدا نگاهی به محاوره انتخابگر حرفی بیندازیم:



مشاهده می کنید که در این محاوره، فهرست محصولات به همراه کد محصولات نمایش داده شده و کاربر می تواند با انتخاب دلخواه، جستجوی مورد نظر خود را در کد محصولات یا نام محصولات انجام داده و محصول مورد نظر خود را انتخاب نماید.

## ساختار محاوره انتخابگر حرفی

همانطور که در محاوره مثال بالا مشاهده می کنید، این محاوره اطلاعات فراخوانی شده را در دو ستون نمایش می دهد:

### ستون کد

در این ستون هر نوع اطلاعات عددی یکتا (Unique) که نشانگر (Identifier) موجود مورد نظر (در اینجا کالا) است استخراج و نمایش داده می شود. برای مثال در این نمونه، کد کالاها در این ستون نمایش داده می شود.

### ستون نام

در این ستون هر نوع اطلاعات حرفی مرتبط با موجود مورد نظر (در اینجا کالا) نمایش داده می شود. چیزی که بیشتر مرسوم است این است که در این ستون، نام یا عنوان نمایش داده شود.

## محدودیت دوستونی برای نمایش اطلاعات

محدودیت دو ستونی، ویژگی اصلی این محاوره است اما برای محتویات هر ستون، محدودیتی برای اطلاعات وجود ندارد. یعنی این که مثلا در ستون "نام" می توانیم هر ترکیبی از اطلاعات را با جداکننده های معین، نمایش دهیم. این ویژگی، خاصیت جستجوپذیری فیلدهای انتخابگر حرفی را توسعه می بخشد. برای مثال می توانیم الگوی ("نام کالا" + ":" + "نام لاتین کالا" ) را برای این ستون نمایش دهیم و به کاربر این امکان را بدهیم که بتواند علاوه بر نام فارسی کالا، نام لاتین کالاها را نیز جستجو و از این طریق کالای مورد نظر خود را بازیابی کند.

## ایجاد و تدوین View برای یک فیلد از نوع انتخابگر حرفی

برای تعریف یک View باید در SQL Server Management Studio ، روی پایگاه فرایند مورد نظر کلیک راست کنید و سپس از منوی بازشده، روی گزینه New Query کلیک کنید.(مانند شکل زیر)



حال کافی است در صفحه باز شده برای Query، و با دستور Create View یک View جدید ایجاد کنید. فقط دقت کنید که نام این View باید با عبارت \_NosaWorkView شروع شود. برای مثال به چند View که در این پایگاه تعریف شده دقت کنید (برای دیدن View ها کافی است مطابق شکل زیر، روی آیکون + سمت چپ پایگاه فرایند کلیک کنید تا محتویات این پایگاه نمایش داده شود و از آنجا پوشه Views را باز کنید).



همانطور که مشاهده می کنید، چهار view در این فرایند ایجاد شده که می توانند محتوای فیلدهای انتخابگر حرفی را تعیین کنند.

 حال برای مثال، یک View جدید با نام \_NosaWorkView\_Products برای نمایش محصولات ایجاد می کنیم. Query این View به شکل زیر است:

CREATE VIEW NosaWorkView\_Products

AS

SELECT mat\_Code AS nwv\_Code, mat\_Name AS nwv\_Name

FROM \_AccXP\_hamayesh992.\_mats

ملاحظه می کنید که View شامل دو ستون **کد** (nwv\_Code) و **نام** (nwv\_Name) است. دقت کنید که این نامها قرارداد است و حتما باید از همین نامها در View استفاده شود. یادآوری می شود که در ستون کد، معمولا اطلاعات عددی از نوع Unique Identifier، و در ستون نام، عموما اطلاعات حرفی از نوع نام، نمایش داده می شود. در مثال فوق که ساده ترین نمونه یک View است. در ستون کد، اطلاعات کد محصول (mat\_Code) در درخت کالاها، و در ستون نام ، اطلاعات نام محصول (mat\_Name) از جدول محصولات (\_mats) از پایگاه سیستم مالی (\_AccXP\_hamayesh992) انتخاب شده است.

## توسعه امکانات انتخابگر حرفی

در مواردی که تعداد نتایج View زیاد می شود برای دستیابی راحت تر و سریع تر به رکورد مورد نظر، بهتر است این فهرست به نوعی فیلتر و طبقه بندی شود تا نتایج دقیق تر و خاص تر گردد . به این منظور دو امکان دسته بندی و امکان فیلتر کردن نتایج براساس اطلاعات کار جاری، به امکانات فیلدهای انتخابگر حرفی افزوده شده که در ادامه توضیح داده شده است:

## امکان دسته بندی

وقتی حجم اطلاعات زیاد می شود نیاز به دسته بندی بیشتر حس می شود. برای نمونه در مثال بالا که می خواهیم نام کالا را اتنخاب کنیم، بهتر است کالاها را براساس انبار نگهداری کالاها دسته بندی کنیم. به این ترتیب از همان ابتدا می توانیم کالاها را براساس انبار فیلتر کنیم و فقط کالاهای یک انبار خاص را نمایش دهیم. به این منظور باید View را ویرایش کنیم و یک ستون دیگر با نام nwv\_Catogory به View اضافه کنیم. در ادامه روش ویرایش View و افزودن دسته بندی با یک مثال شرح داده می شود.

## نحوه ویرایش View

برای ویرایش یک View موجود، روی نام View مورد نظر کلیک راست کنید و از منوی بازشده، گزینه Script View as را کلیک کنید، سپس از منوی کشویی بعدی، گزینه ALTER To و سپس از منوی آخر، گزینه New Query Editor Window را کلیک کنید (مطابق شکل زیر):



به این ترتیب، Viewی مورد نظر در یک صفحه جدید ویرایشگر Query باز می شود که می توانیم به شکل دلخواه، دسته بندی مورد نظر خود را اضافه کنیم. در مثال زیر، ستون دسته بندی براساس نام انبار (st\_Name) به Query اضافه شده است:

**توضیح:** نام ستون دسته بندی طبق قرارداد nwv\_category می باشد.

ALTER VIEW \_NosaWorkView\_Products AS

SELECT mat\_Code AS nwv\_code ,mat\_Name AS nwv\_name, st\_Name AS nwv\_category

FROM [\_AccXP\_hamayesh992].[dbo].\_Mats

 inner join [\_AccXP\_hamayesh992].[dbo].\_MatStock on mat\_Key =mst\_MatKey

 inner join [\_AccXP\_hamayesh992].[dbo].\_Stocks on mst\_StockKey =st\_Key

بنابراین با اضافه شدن دسته بندی به انتخابگر حرفی، یک ستون حاوی اطلاعات دسته بندی به جدول اطلاعات محاوره انتخابگر حرفی افزوده می شود. به ستون دسته بندی "نام انبار" در شکل زیر توجه کنید:



دقت کنید که علاوه بر ستون دسته بندی، یک منوی کشویی دیگر هم به محاوره افزوده شده که شامل فهرست گزینه های دسته بندی (در این مثال نام انبارها) است:



فهرست دسته بندی در واقع مثل این است که به View یک Group by روی ستون دسته بندی اضافه کنیم. حال با انتخاب یکی از این انبارها، فهرست کالاها محدود به این انبار خواهد شد. به عبارت دیگر منوی دسته بندی به شکل یک فیلتر عمل می کند. برای مثال با انتخاب "انبار شهرستانها" ، فهرست کالاهای این انبار به شکل زیر به نمایش درمی آید:



**توضیح**: با افزودن امکان دسته بندی به فیلد انتخابگر حرفی، محدودیت دوستونی به سه ستون افزایش می یابد.

## 2- امکان فیلتر کردن نتایج View به "کار"

گاهی اوقات لازم می شود که نتایج View به محتوای یکی از فیلدهای کار، محدود شود. برای درک بهتر این قضیه، بهتر است مثالی بزنیم. فرض کنید یک شرکت دارای چند شعبه یا نمایندگی در استانهای مختلف است. این شرکت یک فرایند فروش دارد که امور فروش تمام شعب را در بر می گیرد. در این فرایند یک فیلد انتخابگر حرفی وجود دارد که فهرست مشتریان شرکت را از سیستم مالی فراخوانی می کند و نمایش می دهد. حال منطقی است که بخواهیم هر کدام از شعب، فهرست مشتریان استان خود را مشاهده کند. به این ترتیب با توجه به اینکه اطلاعات شعبه، در هر کار ثبت می شود لذا در این حالت، نتایج View محدود می شود به اطلاعات فیلد شعبه در کار جاری.

## یک مثال عملی در فهرست محصولات

همانطور که قبلا مشاهده کردید در این مطلب، مثالی در مورد نمایش فهرست محصولات زده شد که امکان دسته بندی نیز به آن اضافه شد. حال فرض کنید بخواهیم فهرست کالاها را براساس سطح اول درخت کالاها فیلتر کنیم. در ابتدا بهتر است نگاهی به سطح اول کالاها بیندازیم:



حال اگر در هر کار ابتدا ، در فیلد سطح اول، گزینه دلخواه خود را تعیین کنید، آنگاه فهرست کالاها در فیلد محصولات، براساس انتخاب شما فیلتر خواهد شد. برای مثال به شکل زیر دقت کنید. در این فرم سه فیلد تعریف شده است:

**سطح یک کالاها:** در این فیلد سطح اول کالا مشخص می شود

**تمام محصولات:** در این فیلد، تمام محصولات بدون هیچ فیلتری نمایش داده می شود.

**محصولات خاص:** در این فیلد، فهرست محصولات براساس اطلاعات فیلد سطح یک کالاها فیلتر می شود.



برای مثال اگر مطابق شکل بالا، سطح اول را "واکسن" انتخاب کنید، آنگاه فقط فهرست واکسن ها را مشاهده خواهید کرد (شکل زیر):



حال به فهرست کالاهای "مواد اولیه" توجه کنید:



## نحوه افزودن امکان فیلترینگ در View

**نکته مهم:** قبل از این که به جزئیات نوشتن Query مربوط به این امکان بپردازیم باید دقت کنید که برای درک این بخش نیاز داریم با جداول اطلاعات کارها (\_WorkflowFieldInfs22) و رخدادها (\_WorkflowLogs) در پایگاه داده های فرایند، آشنایی مقدماتی داشته باشید.

امکان فیلترینگ براساس کار، از طریق افزودن یک ستون دیگر با نام nwv\_owner و به شکل زیر انجام می شود:

ALTER VIEW [dbo].[\_NosaWorkView\_SpecificProduct]

AS

SELECT mat\_Code AS nwv\_Code, mat\_Name AS nwv\_Name, st\_Name AS nwv\_category,

wfi\_WorkflowLog AS nwv\_Owner

FROM [\_AccXP\_hamayesh992].dbo.\_Mats

INNER JOIN \_WorkflowFieldInfs22 on left(mat\_Code, 2)=wfi\_WF1353

INNER JOIN \_WorkflowLogs ON (wfi\_WorkflowLog=wfl\_Key) and (wfl\_Key=wfl\_StartWfKey)

INNER JOIN [\_AccXP\_hamayesh992].[dbo].\_MatStock on mat\_Key =mst\_MatKey

INNER JOIN [\_AccXP\_hamayesh992].[dbo].\_Stocks on mst\_StockKey =st\_Key

در این Query خطوط سبزرنگ مربوط به امکان فیلترینگ است که به شرح زیر است:

wfi\_WorkflowLog AS nwv\_Owner

wfi\_WorkflowLog در واقع کلید کار است و به این ترتیب در ستون nwv\_owner باید کلید کار درج شود. توضیح این که این مورد، یکی از ستونهای جدول \_WorkflowFieldInfs22 است که شامل اطلاعات کارها به همراه فیلدهای قابل تعریف می باشد. عدد 22 در این جدول نشاندهنده کلید گردش کار مورد نظر است.

INNER JOIN \_WorkflowFieldInfs22 on left(mat\_Code, 2)=wfi\_WF1353

در اینجا جدول اطلاعات کارها با جدول کالاها join شده و شرطی که اعمال شده این است که دو کاراکتر ابتدای کد کالا (mat\_Code) برابر با محتوای فیلد سطح اول کالا (wfi\_WF1353) باشد. یعنی به عبارت دیگر در اینجا کالاها براساس کد سطح اول، فیلتر می شود.

INNER JOIN \_WorkflowLogs ON (wfi\_WorkflowLog=wfl\_Key) and (wfl\_Key=wfl\_StartWfKey)

در اینجا در واقع با Join شدن جدول رخدادها (\_WorkflowLogs) کلید کار مشخص می شود. توضیح این که آنچه در پایگاه داده ها اصالت دارد کلید رخدادها (wfl\_Key) است و کلید کار (wfi\_WorkflowLog) یک مفهوم مستقل نیست و درواقع کلید رخداد آغازین (wfl\_StartWfKey) است.

**نکته** **مهم**: هر تغییری در View در لحظه اعمال نمی شود. بنابراین برای عملیاتی شدن تغییرات، باید سیستم اطلاعاتی جاری مجددا انتخاب شود. برای این کار می توانید کلاینت را ببندید و مجددا باز کنید یا اینکه از طریق گزینه تغییر سیستم اطلاعاتی ، پایگاه اطلاعاتی جاری را مجددا انتخاب کنید. به این ترتیب سیستم اطلاعاتی بروز می شود و آخرین تغییرات در آن اعمال می‌شود.

حال که با این دو امکان آشنا شدید ممکن است این سوال در ذهن شما ایجاد شود که آیا ترکیب این دو امکان هم ممکن است و در این صورت چه اتفاقی می افتد؟ در ادامه، این سوال پاسخ داده شده است.

## ترکیب دو امکان دسته بندی و فیلتر کردن براساس کار

همانطور که در مثال بالا ملاحظه کردید، این View شامل هر دو امکان دسته بندی و فیلترینگ براساس کار بود. در چنین مواردی اتفاقی که می افتد این است که دسته بندی هم براساس کار فیلتر می شود. یعنی این که برای هر کار، فهرست منوی کشویی دسته بندی تغییر می کند و فیلتر می شود. برای مثال به شکل زیر توجه کنید:

 

در این مثال ملاحظه می کنید که در فیلد سطح یک کالاها، "اقلام فروشی" انتخاب شده و براین اساس، لیست دسته بندی (نام انبار) شامل 6 گزینه (انبار) است. حال سطح یک کالاها را تغییر می دهیم:



مشاهده می کنید که برای "کالاهای خرد"، فقط سه انبار موجود است. نتیجه این که با ترکیب دو امکان دسته بندی و فیلتر براساس کار، علاوه بر این که امکان دسته بندی ایجاد می شود، همزمان هم دسته بندی و هم فهرست کالاها براساس کار فیلتر می شود.

حال با شناخت مفهوم و ساختار فیلد انتخابگر حرفی، بهتر است با مراحل تعریف این نوع از فیلدها آشنا شویم.

## مراحل تعریف یک فیلد انتخابگر حرفی

* در محاوره فهرست فیلدهای گردش کار، و از نوار ابزار، علامت + را برای تعریف یک فیلد جدید کلیک کنید تا محاوره تعریف فیلد به شکل زیر باز شود (اگر این آیکون فعال نیست ابتدا آیکون  را کلیک کنید تا محاوره در حالت ویرایش قرار بگیرد):



* مطابق مثال بالا، در این محاوره، پس از وارد کردن "نام" (در این مثال، محصول) و "نام در برنامه" (در اینجا product) از منوی کشویی "نوع داده"، گزینه "انتخابگر حرفی" را انتخاب کنید.
* در قسمت مشخصات خاص نوع داده، روی زبانه انتخابگر حرفی کلیک کنید تا محاوره این تنظیمات به شکل زیر باز شود:



در این بخش، چهار آیتم برای تنظیم وجود دارد که به شرح زیر تدوین می شود:

**نام SQL View فیلد انتخابگر حرفی :** در این قسمت فهرست Viewها را باز کنید و از آنجا Viewی مورد نظر خود (در اینجا \_NosaWorkView\_Products) را انتخاب کنید.

**گزینه "انتخابگر حرفی براساس کار فیلتر شود"** : اگر می خواهید نتایج براساس اطلاعات کار جاری فیلتر شود، این گزینه را تیک بزنید. در اینجا دقت کنید که باید قبلا این مورد را در View اضافه کرده باشید در غیر این صورت، در هنگام باز شدن محاوره با پیغام خطا مبنی بر عدم وجود nwv\_owner روبرو خواهید شد.

**گزینه "انتخابگر حرفی دارای دسته بندی است":** اگر مایل به استفاده از دسته بندی هستید این گزینه را تیک بزنید. در اینجا دقت کنید که باید قبلا دسته بندی مورد نظر خود را در View اضافه کرده باشید در غیر این صورت، در هنگام باز شدن محاوره با پیغام خطا مبنی بر عدم وجود nwv\_catogory روبرو خواهید شد.

**نام دسته بندی انتخابگر حرفی:** اگر گزینه قبلی را تیک زده اید در اینجا نام دسته بندی مورد نظر خود را وارد کنید. هر عبارتی را که در اینجا وارد کنید در محاوره انتخابگر حرفی در دو ناحیه ستون دسته بندی و برچسب منوی کشویی دسته بندی نمایش داده خواهد شد (این دو ناحیه در شکل زیر با فلش مشخص شده است):

