# عملیات خاص گردش کار-- (Action) 1

# عملیات از نوع انجام کار

برای درک بهتر عملیات خاص گردش کار ابتدا باید با انواع عملیات (فعالیتهای) موجود در فرایندها آشنا شویم.

## انواع فعالیتهای موجود در فرایندها

انواع فعالیتها در گردش کار به دو دسته تقسیم می شود: فعالیتهایی که کاری را انجام می دهند و فعالیتهایی که از نوع گزارشگیری و اطلاع از وضعیت (حالت) کارها هستند. در ادامه، به فعالیتهای نوع اول (انجام کار) می­پردازیم.

## فعالیتهای انجام کار

انواع فعالیتهای از نوع انجام کار را می توان به شکل زیر طبقه بندی کرد:

* تغییر مقادیر فیلدهای گردش کار
* ارسال پیام (message)
* استفاده از ابزارهای سیستم نظیر ارسال اعلامیه ها (پیامک و ایمیل)
* تغییر وضعیت کار

برداشت و حس عمومی از فرایندها این است که هر فعالیتی از این نوع در گردش کار، در قالب یک وظیفه (Task) انجام می شود. از این طریق هر کدام از این فعالیتها می تواند از طریق فرمهای عمومی یا اختصاصی و یا از طریق برنامه و اجرای یک تابع اجرایی (WorkTask) وابسته به وظیفه انجام شود. اما نکته ای که باید در نظر داشته باشید این است که ، انجام هر وظیفه، کار را از یک وضعیت (مبدا) به یک وضعیت دیگر (مقصد) منتقل می کند.

**نکته مهم:** فعالیتهایی که در قالب وظیفه انجام می شوند، وضعیت (Status) کار را تغییر می دهند.

حال فرض کنید، می خواهیم فعالیتهایی را انجام دهیم بدون اینکه وضعیت کار تغییر پیدا کند. در ادامه بحث، به این نوع از فعالیتها می پردازیم:

## فعالیتهای انجام کار بدون تغییر وضعیت کار

بنا به هردلیل، ممکن است در طول روند یک گردش کار، نیاز به فعالیتی داشته باشیم که کار مورد نظر ما را انجام دهد اما وضعیت گردش کار تغییر نکند. برای روشن شدن این قضیه، به مثالهای زیر توجه کنید:

### مثال 1: افزایش قیمت محصول

مثال فرض کنید در یک فرایند فروش محصولات، به دلیل تورم، بخواهیم در هر مرحله از کارهای فعال، قیمت محصول را افزایش دهیم. در این صورت باید بتوانیم فارغ از وضعیت فعلی کارها، محتوای فیلد قیمت محصول را تغییر دهیم بدون این که در جریان عادی فرایند، تغییری ایجاد شود.

### مثال 2: سپردن کارها به یک کارشناس دیگر

فرض کنید یک گردش کار خاص برای پیگیری شکایات مشتریان داریم که یک کارشناس خاص، مسئول رسیدگی به این شکایات است. حال به هر دلیل این کارشناس با سازمان قطع همکاری می کند. حال باید بتوانیم کل کارهای فعال و در دست انجام این کارشناس را به کارشناس دیگری ارجاع دهیم یا به عبارت بهتر، تمام کارهای کارتابل این کارشناس را به کارتابل کارشناس دیگری منتقل کنیم. واضح است که این کار هم مستقل از وضعیت کارهاست و تنها قرار است کاربر مسئول انجام کارها را تغییر دهد.

### مثال 3: ارسال پیامک به مشتریان در هر مرحله از کار

فرض کنید در یک فرایند ثبت نام برای سمینار، بخواهیم یک مورد خاصی را به مشتریان اطلاع رسانی کنیم. برای مثال تغییر تاریخ یا مکان سمینار، لغو سمینار و ... ممکن است که این اطلاع رسانی به وضعیت کار بستگی نداشته باشد و می‌خواهیم بتوانیم در هر مرحله، این اطلاع رسانی را انجام دهیم.

این دسته از فعالیتها به شکل عملیات خاص طراحی و پیاده سازی می شود. با توجه به مطالب گفته شده، در اینجا به یک ویژگی کلی در رابطه به عملیات خاص می رسیم:

### عملیات خاص به منزله ابزار

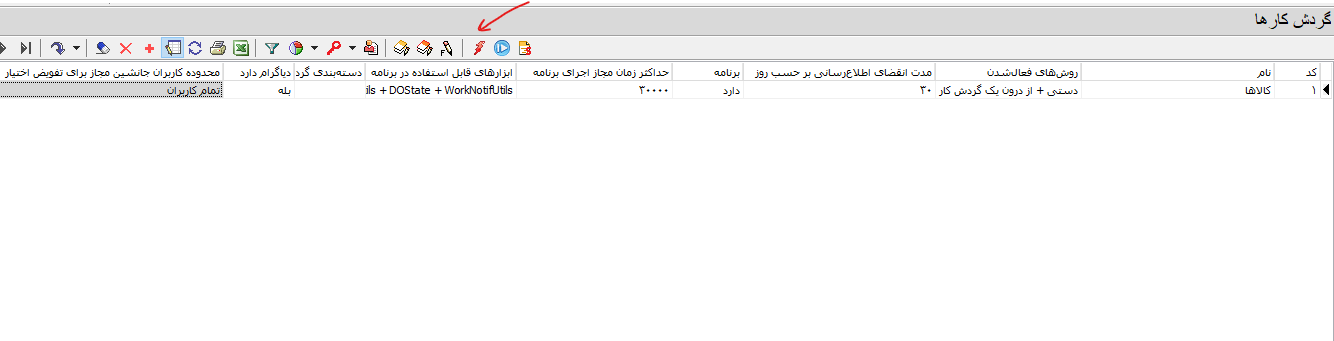
عملیات خاص در واقع یک سری ابزار است که در سطح گردش کار تعریف می شود و در اختیار کاربر قرار می گیرد. این ابزارها مستقل از وضعیت و در تمام مراحل کارهای گردش کار می تواند استفاده شود.

در ادامه، با مراحل تعریف یک عملیات خاص از نوع انجام کار آشنا می شویم:

## مراحل تعریف عملیات خاص برای انجام کار

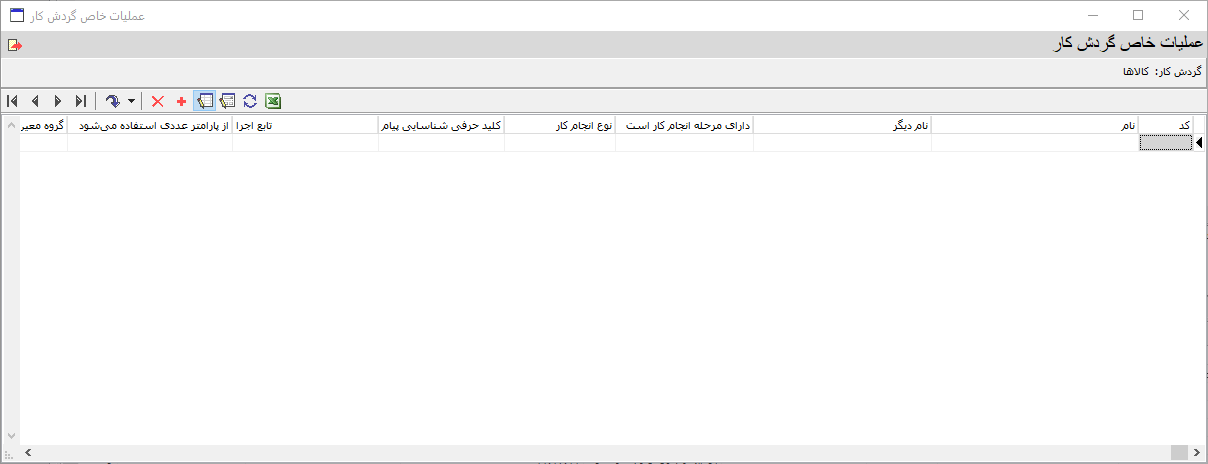
برای تعریف یک عملیات خاص:

1. از منوی **گردش کار**، گزینه **تعریف و اصلاح گردش کارها** را کلیک کنید و گردش کار مورد نظر خود را انتخاب کنید (شکل زیر)

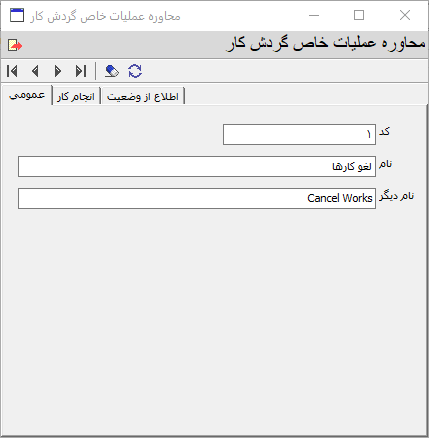


همانطور که مشاهده می کنید، در این مثال، گردش کار **کالاها** انتخاب شده و قرار است برای این گردش کار، عملیات خاص تعریف کنیم. حال:

1. روی آیکون عملیات خاص  (در شکل با فلش قرمزرنگ مشخص شده است) کلیک کنید تا محاوره **تعریف عملیات خاص** مانند شکل زیر روی صفحه باز شود:



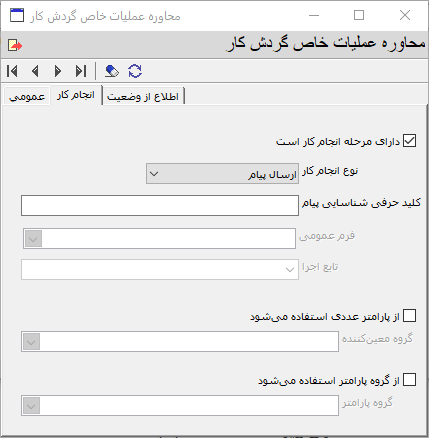
1. برای ایجاد یک عملیات خاص جدید، روی آیکون  کلیک کنید (یا اینکه در ناحیه پایین دبل کلیک کنید). به این ترتیب، محاوره تعریف عملیات خاص به شکل زیر باز می شود:



1. در زبانه **عمومی**، در قسمت **نام** یک نام فارسی و در صورت تمایل، در قسمت **نام دیگر**، یک نام انگلیسی وارد کنید (نام دیگر اختیاری است)

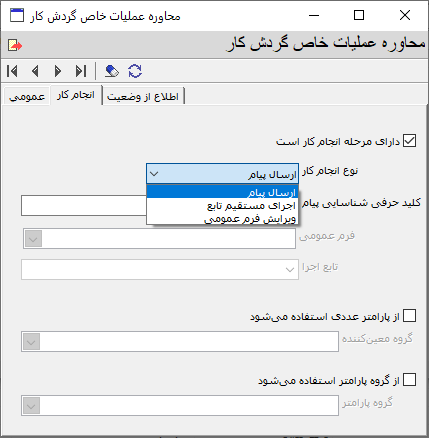
**توضیح**: برای پرهیز از تکرار و طولانی شدن مطلب، مرحله تعیین نام برای عملیات خاص فقط برای این مورد ذکر می شود و برای مثالهای بعدی تکرار نخواهد شد و فرض می شود که در تعریف هر عملیات خاص، ابتدا نام عملیات خاص تعیین می شود.

1. حال باید زبانه **انجام کار** را کلیک کنید تا محاوره تنظیمات انجام کار به شکل زیر باز شود:



در این محاوره به شکل پیش فرض، گزینه **دارای مرحله انجام کار است** باید فعال باشد (بطور پیش فرض هم همینطور است)

1. حال باید نوع انجام کار را مشخص کنید. برای این کار، فهرست انواع انجام کار را باز کنید که دارای سه گزینه به شکل زیر است:



همانطور که می بینید، سه نوع انجام کار وجود دارد که بسته به انتخاب هر گزینه، تنظیمات خاص آن باید به شرح زیر انجام شود:

## انواع انجام کار در عملیات خاص

### روش ارسال پیام

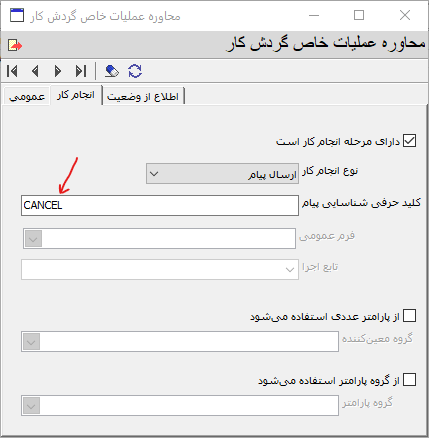
در این روش، انجام کار از طریق ارسال یک پیام (Message) با پارامترهای مشخص عددی و حرفی انجام می شود. دقت کنید که در این روش، نتیجه کار توسط گیرنده پیام تعیین می شود یعنی این که ما پیام را به کارها ارسال می کنیم و پردازش پیام در کارها انجام می شود و به عبارتی عملکرد پیام به عهده گیرنده گذاشته می شود. مثلا این که کار به وضعیت آغاز پیام می رود یا این که تابع WorkProcessMessage در مقصد اجرا می شود یا هر دو و یا ممکن است در شرایط خاصی هیچ کاری انجام نشود. حال ممکن است این سوال پیش بیاید که چه عملیاتی از طریق ارسال پیام انجام می شود. در جواب باید بگوییم که هر گونه عملیاتی با این روش ممکن است فقط لازم است گیرنده پیام پیش‌بینی شده باشد.

**تنظیمات عملیات خاص از نوع ارسال پیام**

با یک مثال در این زمینه، به شرح تنظیمات این نوع از عملیات خاص می پردازیم:

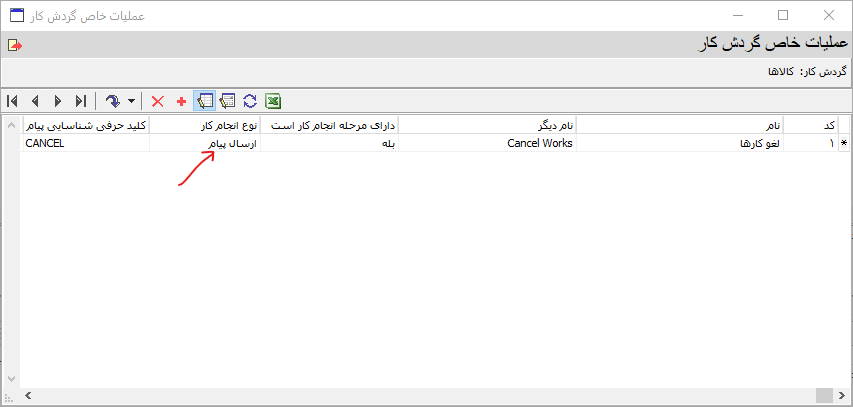
**مثال : لغو یکباره تعدادی از کارهای یک گردش کار**

فرض کنید بخاطر مرور زمان یا بنا به هر دلیل دیگری می خواهیم تعدادی از کارهای یک گردش کار را به یکباره لغو کنیم. برای این کار کافی است گزینه ارسال پیام را انتخاب کنید و در ناحیه کلید حرفی شناسایی پیام، عبارت کلیدی پیام لغو را وارد کنید (همانند شکل زیر):



**توضیح:** اگرچه برای ارسال پیام امکان تعیین پارامترهای عددی و گروه پارامتر هم وجود دارد اما در این مطلب برای راحتی و پرهیز از پیچیدگی مثالی زده شده که نیازی به تعیین پارامتر ندارد.

به این ترتیب، یک عملیات خاص از نوع ارسال پیام ایجاد می شود که با بستن این محاوره، عملیات خاص ایجاد شده را در محاوره زیر مشاهده خواهید کرد:



**نکته مهم:** در این مثال، فرض بر این است که قبلا یک وضعیت آغاز پیام با عبارت کلیدی CANCEL در دیاگرام تعریف شده است که کار را به وضعیت لغو منتقل می کند.

### روش اجرای مستقیم تابع

در این روش، فعالیت مورد نظر، در یک تابع با الگوی عنوان WorkActionMySample درج می شود. و با اجرای مستقیم این تابع، عملیات خاص مورد نظر انجام می شود. تمام فعالیتهای از نوع تغییر مقادیر فیلدها و یا فعالیتهای ارسال اعلامیه (پیامک یا ایمیل) یا استفاده از سایر قابلیت‌های برنامه گردش کار می‌تواند از این طریق انجام شود.

### تفاوتهای روش اجرای مستقیم تابع با روش ارسال پیام

* روش اجرای مستقیم تابع، به شکل همزمان (سنکرون)انجام می شود اما روش ارسال پیام غیرهمزمان (آسنکرون) است. توضیح اینکه در روش اجرای مستقیم تابع، برخلاف روش ارسال پیام، عملیات بطور مستقیم انجام می شود و به عهده گیرنده گذاشته نمی شود. به عبارت دیگر در روش ارسال پیام، عملیات بطور غیرمستقیم و با واسطه پیام انجام می شود و به همین دلیل درگیر روند دریافت و پردازش پیام و تاخیرهای احتمالی این روند در گیرنده است ولی در عوض، این مزیت را دارد که تغییر در شیوه دریافت پیام توسط گیرنده، منجر به تغییر تنظیمات عملیات خاص نمی‌شود.
* تفاوت دوم این دو روش در ثبت وقایع (history log) است. در روش ارسال پیام تمام وقایع ثبت (log) می شود اما در روش اجرای مستقیم تابع، اینطور نیست و وقایع ثبت نمی شود.

**نکته مهم:** هر دو روش اجرای مستقیم تابع یا روش ارسال پیام کارکردهای خاص خود را دارند و بسته به نوع طراحی فرایند در هر مرکز و ترجیح روالهای همزمان (سنکرون) یا غیرهمزمان (آسنکرون) که بستگی زیادی به شرایط و معماری فرایندها و تعدد کاربران دارد، می توان از هر دو روش برای عملیات خاص استفاده کرد.

**تنظیمات عملیات خاص از نوع اجرای مستقیم تابع**

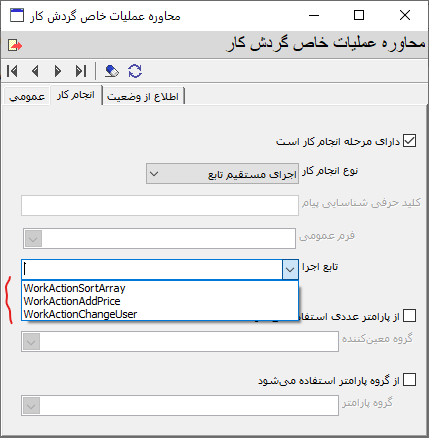
حال برای آشنایی بیشتر با نحوه تعریف این نوع از عملیات خاص به مثال زیر توجه کنید:

**مثال 1: افزایش قیمت کالا به میزان متغیر و دلخواه**

برای این کار، باید از فهرست انواع انجام کار، گزینه دوم یعنی اجرای مستقیم تابع را انتخاب کنید. به این ترتیب محاوره تنظیمات عملیات خاص به شکل زیر در می آید:

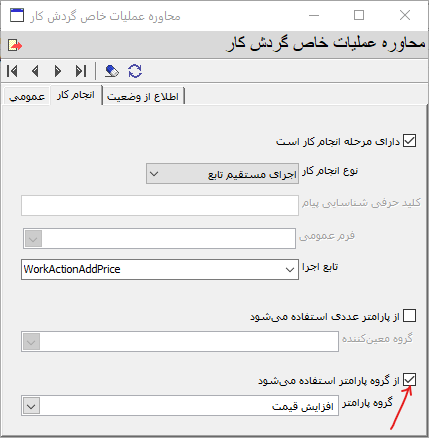


همانطور که می بینید، ناحیه تابع اجرا فعال می شود یعنی این که در این مرحله باید تابع اجرای این عملیات خاص را تعیین کنید. برای این کار، فهرست کشویی تابع اجرا را باز کنید تا فهرست توابع اجرای عملیات خاص را به شکل زیر مشاهده کنید:



**توضیح:** این توابع باید از قبل نوشته شده و موجود باشند. در غیر این صورت بطور همزمان هم می توانید این توابع را ایجاد و کدنویسی کنید. برای آشنایی با نحوه نوشتن این توابع، به ضمیمه این مطلب با عنوان **مثالهای توابع اجرای مستقیم عملیات خاص** مراجعه کنید.

حال باید از فهرست توابع اجرا ، تابع مورد نظر خود را انتخاب کنید. برای مثال در اینجا تابع WorkActionAddPrice\* را انتخاب می کنیم که برای افزایش قیمت کالا نوشته شده است:

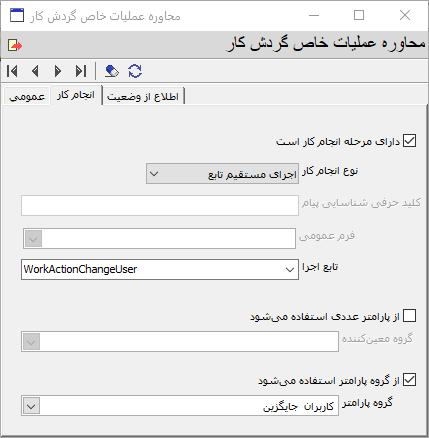


برای اینکه مقدار افزایش قیمت مقدار ثابتی نیست و بسته به شرایط ممکن است تغییر کند لذا این مقدار باید پارامتریک باشد و در لحظه انجام عملیات خاص توسط کاربر تعیین شود. برای این کار همانطور که در شکل بالا هم مشخص شده است، گزینه **از گروه پارامتر استفاده می شود** را تیک بزنید و از فهرست گروه پارامترهای موجود، گروه پارامتر مرتبط با این عملیات خاص (در اینجا گروه پارامتر افزایش قیمت) را انتخاب کنید.

\*: جزئیات این تابع در ضمیمه این مطلب با عنوان **مثالهای اجرای مستقیم توابع عملیات خاص** شرح داده شده است.

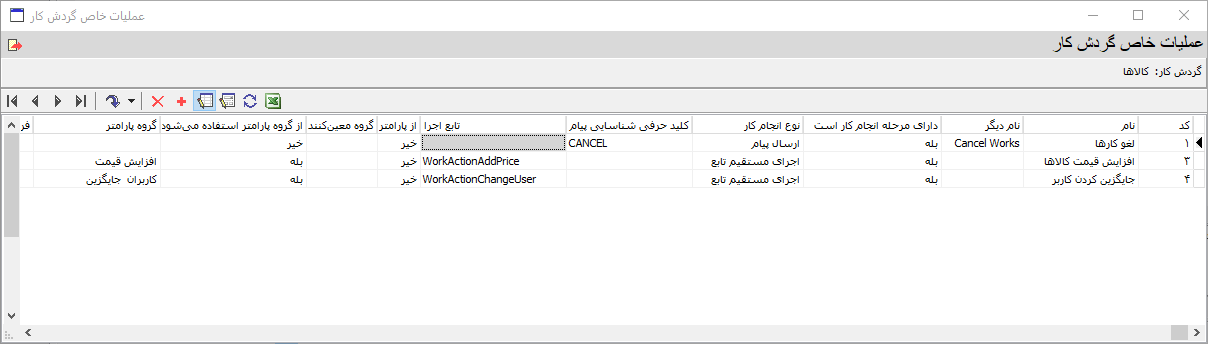
**مثال 2: جایگزین کردن کاربر**

برای تعریف این عملیات خاص، در قسمت تابع اجرا، تابع WorkActionChangeUser را که برای تغییر کاربر مجازی نوشته شده انتخاب کنید.



در این مثال نیز چون کاربر جایگزین در لحظه تعیین می شود به شکل پارامتریک تعریف شده و گروه پارامتر **کاربران جایگزین** برای این عملیات خاص انتخاب شده است.

با تعریف این دو عملیات خاص، فهرست عملیات خاص به شکل زیر بروز می شود:

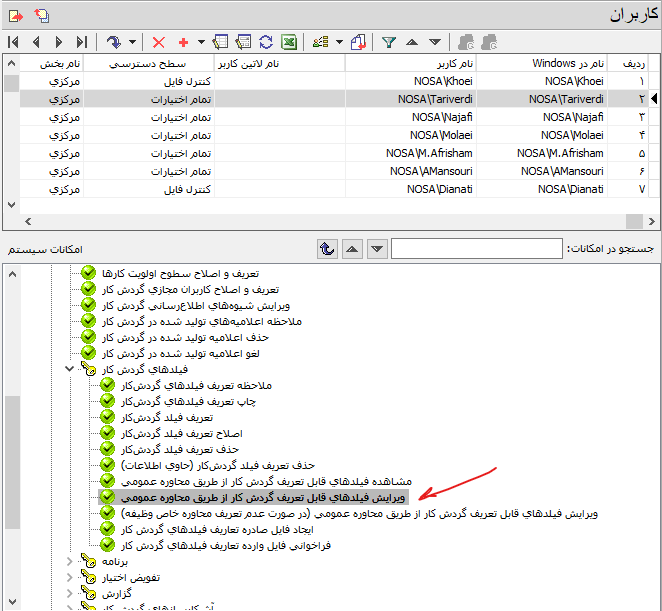


### روش ویرایش فرم عمومی

در این روش، فرم عمومی باز می شود و کاربر می تواند مقادیر فیلدهای موجود در فرم را ویرایش کند. مزیت استفاده از فرم عمومی نسبت به روش قبل (اجرای مستقیم تابع) این است که محدود به یک فیلد نیست و دست کاربر باز است تا بتواند تمام فیلدهای موجود در فرم عمومی را ویرایش کند. به عبارت دیگر اگر این روش نبود، برای تک تک فیلدهای موجود در فرم عمومی باید یک عملیات خاص از نوع اجرای مستقیم تابع تعریف می کردیم.

قبل از اینکه وارد جزئیات این روش شویم، بهتر است به یک سوال یا ابهامی که ممکن است در ذهن ایجاد شود بپردازیم:

**سوال:** چرا بجای استفاده از عملیات خاص که پیچیدگیهای خاص خود را دارد، اختیار (Facility) ویرایش فرم عمومی به کاربر داده نمی شود؟ (این اختیار در شکل زیر نمایش داده شده است)



**جواب:** به دلایل زیر بهتر است از عملیات خاص استفاده شود:

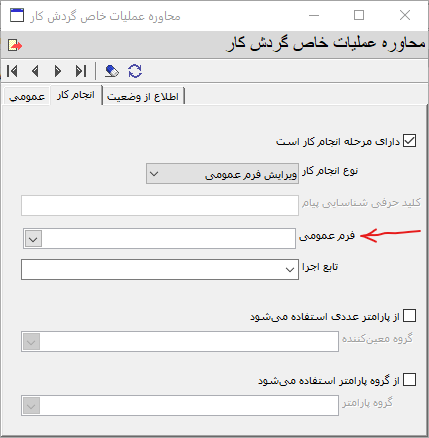
1. برای ویرایش فرم عمومی باید اختیار مدیریت کارها را هم به کاربر بدهیم. اختیار مدیریت کارها علاوه بر ویرایش فرم عمومی، شامل اختیارات دیگری هم هست از جمله ، ارسال پیام، تعیین کاربران مجازی کارها به شکل دستی، تعیین اولویت کارها، تعلیق کارها و ... که شاید مدیر سیستم مایل نباشد این اختیارات به کاربر داده شود .
2. اختیار ویرایش فرم عمومی در سطح پایگاه است و اگر این اختیار به کاربری داده شود، آنگاه در تمام گردش کارها اختیار ویرایش فرم عمومی را خواهد داشت.
3. عملیات خاص از درگاه وب نیز قابل انجام است. با این ابزار می‌توان امکان ویرایش فیلدهای کار توسط هر فرم عمومی (نه فقط فرم عمومی پیش‌فرض) را به کاربران وب داد.
4. دسترسی کاربران به عملیات خاص مورد به مورد قابل تعیین و تنظیم است.

پس نتیجه می گیریم که با استفاده از عملیات خاص، می توانیم ویرایش فرم عمومی را به شکل کنترل شده و مدیریت شده به در اختیار کاربران قرار دهیم.

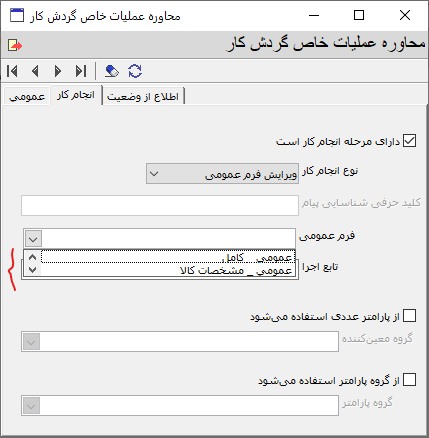
**نکته**: امکان ویرایش مقادیر فیلدها منحصر به روش ویرایش فرم عمومی نیست و از روشهای ارسال پیام یا اجرای مستقیم تابع هم می توان این کار را انجام داد اما این روشها پیچیدگیهای خاص خود را دارند و نیازمند پاس کردن پارامترهای خاص برای این کار هستند. بنابراین ویرایش از طریق فرم عمومی بسیار ساده تر است و در اکثر موارد نیازی به پاس کردن پارامتر ندارد.

**تنظیمات عملیات خاص از نوع ویرایش فرم عمومی**

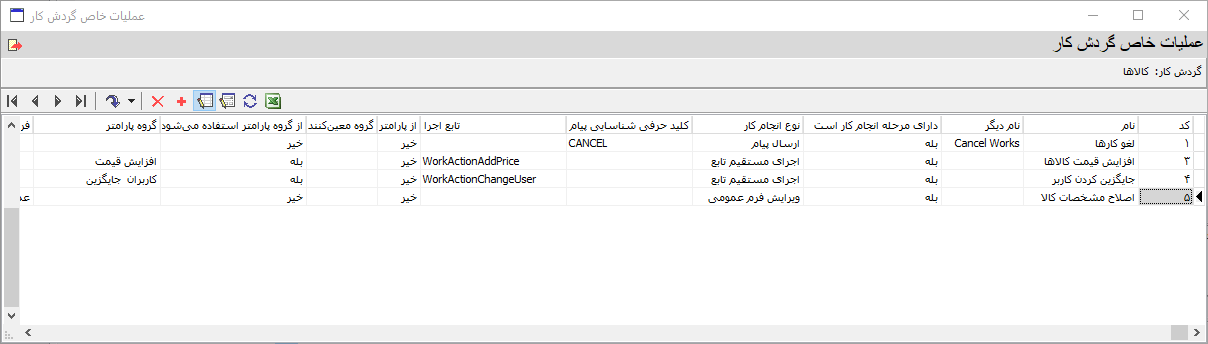
برای تعریف این نوع از عملیات خاص، از فهرست انواع کارها، گزینه سوم یعنی ویرایش فرم عمومی را انتخاب کنید. به این ترتیب، محاوره تنظیمات انجام کار به شکل زیر در می­آید:



همانطور که می بینید، در این حالت، ناحیه انتخاب **فرم عمومی** فعال می شود.حال باید فرم عمومی مورد نظر خود را انتخاب کنید. برای این کار فهرست کشویی فرم عمومی را باز کنید (شکل زیر) و از این فهرست، فرم عمومی مورد نظر خود را انتخاب کنید:



توضیح این که در سیستم مدیریت فرایندها امکان تعریف فرمهای عمومی متعدد وجود دارد. به این ترتیب می توانید برای هر سناریوی ویرایش، یک فرم عمومی مختص به آن سناریو تعریف کنید و در این فرم، فیلدهایی را قرار دهید که می خواهید ویرایش شود. برای مثال برای این عملیات خاص، فرم عمومی\_مشخصات کالا را انتخاب می کنیم. یعنی می خواهیم از طریق این فرم تنها مشخصات کالا (نام کالا، نام تولید کننده، سایز، قیمت، تعداد و ...) ویرایش شود. به این ترتیب یک سطر دیگر به فهرست عملیات خاص اضافه می شود:



# ضمیمه: مثالهای توابع اجرای مستقیم عملیات خاص

در این مطلب، در روش عملیات خاص از نوع اجرای مستقیم تابع، دو مثال زده شد. در این قسمت برای آشنایی بیشتر با نحوه کدنویسی این توابع به شرح جزئیات تابع در این دو مثال می پردازیم:

## مثال 1: افزایش قیمت کالا

تابع اجرای مستقیم این عملیات خاص به شکل زیر است:

function WorkActionAddPrice(n, p }{

if (WorkState1.kalaPrice) {

var amountUp=WorkState1.kalaPrice \* p.addPriceCoef;

WorkState1.kalaPrice+=amounUpt;

{

{

حال این تابع را خط به خط بررسی می کنیم:

function WorkActionAddPrice(n, p }{

همانطور که قبلا گفته شد، عنوان توابع اجرای مستقیم با پیش عبارت WorkAction شروع می شود. نکته دیگر پارامترهای این تابع هستند که n برای پارامتر عددی و p نشانگر گروه پارامتر است. در اینجا یادآوری می شود که برای این مثال گروه پارامتر **افزایش قیمت** انتخاب شده است.

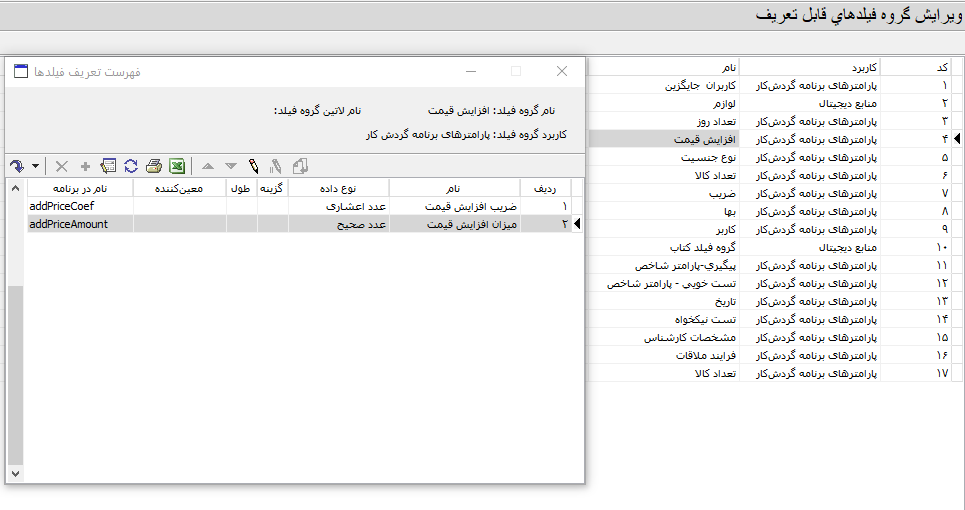
**نکته**: چون در این مثال، پارامتر عددی برای عملیات خاص تعیین نشده است، لذا این پارامتر صرفا برای رعایت syntax درج شده و استفاده دیگری نخواهد داشت.

if (WorkState1.kalaPrice)}

در اینجا شرط گذاشته شده که در صورت وجود قیمت کالا، این تابع اجرا شود. بطور کلی روی هر فیلد که می خواهید پردازشی انجام دهید ابتدا باید وجود آن فیلد را بررسی کنید و در صورت وجود، پردازشها را انجام دهید در غیر این صورت با پیغام خطا روبرو خواهید شد.

var amountUp=WorkState1.kalaPrice \* p.addPriceCoef;

در این سطر یکی از پارامترهای زیرمجموعه گروه پارامتر **افزایش قیمت** برای درک بهتر این سطر، بهتر است ابتدا نگاهی به این گروه پارامتر بیندازیم:



همانطور که ملاحظه می کنید، این گروه پارامتر دارای دو پارامتر است:

* ضریب افزایش قیمت (addPriceCoef): این پارامتر یک عدد اعشاری است که درصد افزایش قیمت را تعیین می کند. بنابراین اگر بخواهیم افزایش قیمت را به شکل درصدی درنظر بگیریم باید از این پارامتر استفاده کنیم.
* میزان افزایش قیمت (addPriceAmount) : اگر بخواهیم یک مقدار ثابت (مثلا 15000) را برای افزایش قیمت در نظر بگیریم آنگاه می توانیم از این پارامتر استفاده کنیم.

حال که با جزئیات این گروه پارامتر آشنا شدیم می توانیم سطر برنامه را بهتر تحلیل کنیم:

var amount p=WorkState1.kalaPrice \* p.addPriceCoef;

واضح است که در این مثال از پارامتر ضریب افزایش قیمت (addPriceCoef) استفاده شده و با ضرب کردن این ضریب در قیمت کالا، میزان افزایش قیمت در متغیر amountUp قرار داده شده است.

**یادآوری:** گروه پارامتر (در اینجا p) یک Object است که پارامترهای زیرمجموعه آن طبق الگوی Object استخراج می شود. مثلا در اینجا p.addPriceCoef

**نکته:** چون در هنگام تعریف عملیات خاص، و در محاوره تنظیمات عملیات خاص، گروه پارامتر را به شکل دستی مشخص کرده ایم لذا نیازی به ذکر نام در برنامه گروه پارامتر نیست و سیستم بطور اتوماتیک (و از طریق p) گروه پارامتر را تشخیص می دهد.

WorkState1.kalaPrice+=amountUp;

در این سطر، میزان افزایش قیمت به قیمت کالا افزوده می شود.

## مثال 2: جایگزینی کاربر انجام وظایف

تابع اجرای مستقیم این عملیات خاص به شکل زیر است:

function WorkActionChangeUser(n, p) {

var selectedUser=p.newUser;

WorkUserUtils.ClearVirtualUserMemberList("Nosa\\WorkPerformer", WorkState1.WorkKey);

WorkUserUtils.AddVirtualUserMember("Nosa\\WorkPerformer", WorkState1.WorkKey, selectedUser);

}

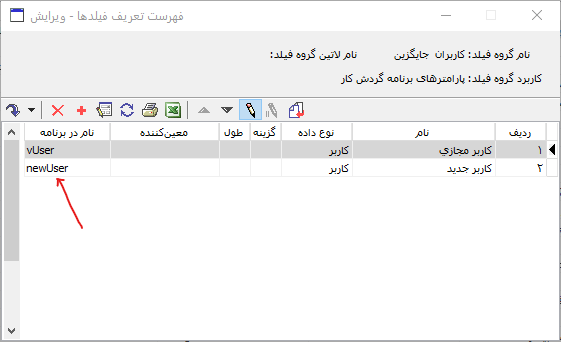
این تابع را نیز خط به خط تحلیل می کنیم:

function WorkActionChangeUser (n, p) {

این تابع هم با پیش عبارت WorkAction تعریف شده و نشانگر این است که یک تابع اجرای مستقیم عملیات خاص است. ضمنا چون این عملیات هم پارامتریک است، لذا باید پارامترهای عددی (در صورت لزوم) و گروه پارامتر نیز به این تابع پاس شود.

var selectedUser=p.newUser;

در اینجا محتوای پارامتر **کاربر جدید** (newUser) از گروه پارامتر **کاربران جایگزین** در یک متغیر بنام selectedUser قرار داده شده است. در این مورد هم بهتر است نگاهی به گروه پارامتر **کاربران جایگزین** داشته باشیم:



ملاحظه می کنید که این گروه پارامتر هم شامل دو پارامتر **کاربر مجازی (vUser)** و **کاربر جدید** (**newUser**) است که در این مثال از پارامتر **کاربر جدید** استفاده شده است.

WorkUserUtils.ClearVirtualUserMemberList("Nosa\\WorkPerformer", WorkState1.WorkKey);

چون قرار است کاربر تغییر کند لذا در این مرحله، لیست کاربران کاربر مجازی WorkPerformer خالی می شود تا در مرحله بعد کاربر مورد نظر به این لیست اضافه شود.

WorkUserUtils.AddVirtualUserMember("Nosa\\WorkPerformer", WorkState1.WorkKey, selectedUser);

در این مرحله، کاربر جایگزین selectedUser که از طریق پارامتر تعیین شده است، به لیست کاربر مجازی WorkPerformer افروده می شود و به این ترتیب کارهای کاربر قبلی به کارتابل کاربر جایگزین منتقل می گردد.