

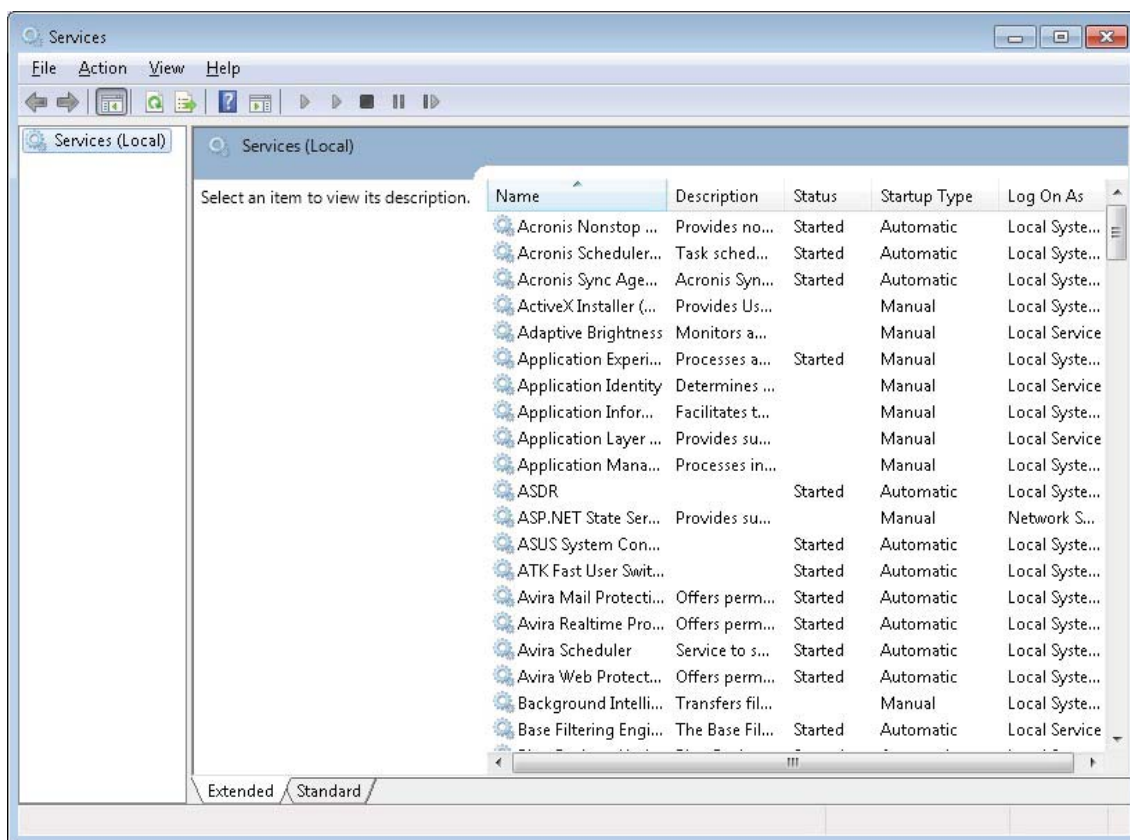
SAYEYE ZOHOR SOFTWARE GROUP

Windows Service in Delphi Interface

With Delphi 2010

Windows Service in Delphi

1391/03/30



E-Mail: M_Nikoei2005@Hotmail.com

Copyright(c) 1391-1392 SayeyeZohorSoftwareGroup

مراحل ساخت ، نصب و اجرای یک سرویس ویندوزی مانند (MSSQLSERVER، PCASVC)

توضیحی راجع به سرویس های ویندوز :

حتماً تا بحال برنامه آنتی ویروس نصب کرده اید. اکثر اینگونه برنامه ها یک ویژگی به نام **Auto Protection** دارند. وقتی این ویژگی را فعال کنید به محض ورود به یک پوشه فایل یا فایل هایی که آلوده به ویروس باشد را شناسایی کرده و شما را از این موضوع با خبر می کند . یا شما در یک زمان از طریق چندین برنامه فایل هایی را برای پرینت شدن به **Printer** متصل به کامپیوتر یا متصل به شبکه ارسال می کنید و فایل های شما بدون هیچ مشکلی پرینت می شوند و مثال هایی از این قبیل . سوالی که مطرح می شود این است که کنترل پوشه ها برای محافظت از ویروسها و یا مدیریت فایل های ارسالی به پرینت توسط چه برنامه هایی و به چه صورت انجام می شود؟

چیزی که واضح است این است که ما با اکثر این برنامه ها در تعامل (**Interact**) نیستیم ، اکثر این برنامه ها را ما به صورت دستی اجرا نمی کنیم ، این برنامه ها همیشه در حال اجرا هستند و سرویس هایی را برای ما و یا سایر برنامه ها ارائه می دهند. این برنامه ها **سرویس** نامیده می شوند . در واقع سیستم عامل اکثر کارها و سرویس هایی که برای ما انجام می دهد توسط این سرویس ها ارائه می دهد . پس سیستم عامل کنترل زیادی بر روی این سرویس ها خواهد داشت .

از سرویس هایی که سیستم عامل ارائه می دهد می توان به **IIS (Internet Information Service)** ، **Telnet** و ... اشاره کرد . علاوه بر اینها **Windows XP** سرویسی به نام **Themes** دارد که وظیفه آن ایجاد ظاهر گرافیکی برای ویندوز است . اگر شما این سرویس را **Stop** کنید ظاهر ویندوز اکس پی، شبیه ویندوز ۲۰۰۰ و یا ویندوز ۲۰۰۳ سرور خواهد شد . در ادامه به بررسی ویندوز سرویسها و نحوه ایجاد آن در **Delphi 2010** خواهیم پرداخت .

آشنایی با سرویسها و مشاهده آنها

برای مشاهده و کنترل تمام سرویس های نصب شده بر روی ویندوز می توانید از **Service Management Console** استفاده کنید. برای دسترسی به این برنامه از ۴ روش زیر می توان استفاده نمود . در این قسمت شما کلیه سرویس هایی که بر روی سیستم نصب شده اند را مشاهده می کنید . همچنین می توانید وضعیت آنها را در ستون **Status** و نحوه اجرا شدن آنها را در ستون **Start Type** و نیز صاحب سرویس را مشاهده کنید . با انتخاب هر سرویس ، در سمت چپ اطلاعاتی در مورد سرویس ظاهر خواهد شد. در شکل زیر می توانید این مطالب را مشاهده کنید.

تعریف Windows Service

ویندوز سرویس یک سری کد است که در **Background** اجرا می شود و معمولاً با کاربر در ارتباط نمی باشد. از خصوصیات این کدها کنترل کردن (**Handle**) رویدادهای خاص سیستم است. مثل تغییرات برق سیستم و ... در ضمن این برنامه ها می توانند به صورت اتوماتیک هنگام روشن شدن سیستم شروع به اجرا شوند. سرویس ها در تمام مدت در حال اجرا هستند و سرویس مور نظر را ارائه می دهند.

سرویس های ویندوز به ما امکان اجرای رویه هایی در پس زمینه یا **Background** سیستم را می دهد. با کمک سرویس های ویندوز می توانیم تکالیفی مانند کنترل استفاده از پایگاه داده را انجام دهیم. قابل توجه این که، سرویس های ویندوز در فضای مخصوص به خود اجرا می شوند و به فعالیت ادامه می دهند تا کاربر یا کامپیوتر آن سرویس را غیر فعال یا به اصطلاح خاموش کند.

سرویس های ویندوز نوعی ویندوز اپلیکیشن هستند که در فضای جداگانه ای اجرا می شوند ، این سرویس ها به صورت پشت پرده و در هنگام بارگزاری ویندوز اجرا می شوند . معمولاً برای انجام عملیات دوره ای در زمان های پی در پی از این سرویس ها به کمک یک **Timer** استفاده می شود.

در حقیقت شما میتوانید برای عملیاتی که نیاز است به صورت دوره ای به بانک اطلاعات وصل شده و یک سری عملیات را انجام دهید از این سرویس ها استفاده نمایید ، به عنوان مثال شما قصد دارید به هر کاربری که اعتبارش از ۵۰۰ ریال کمتر شد یک ایمیل بفرستید ، این کار به سادگی از طریق سرویس های ویندوز قابل پیاده سازی است.

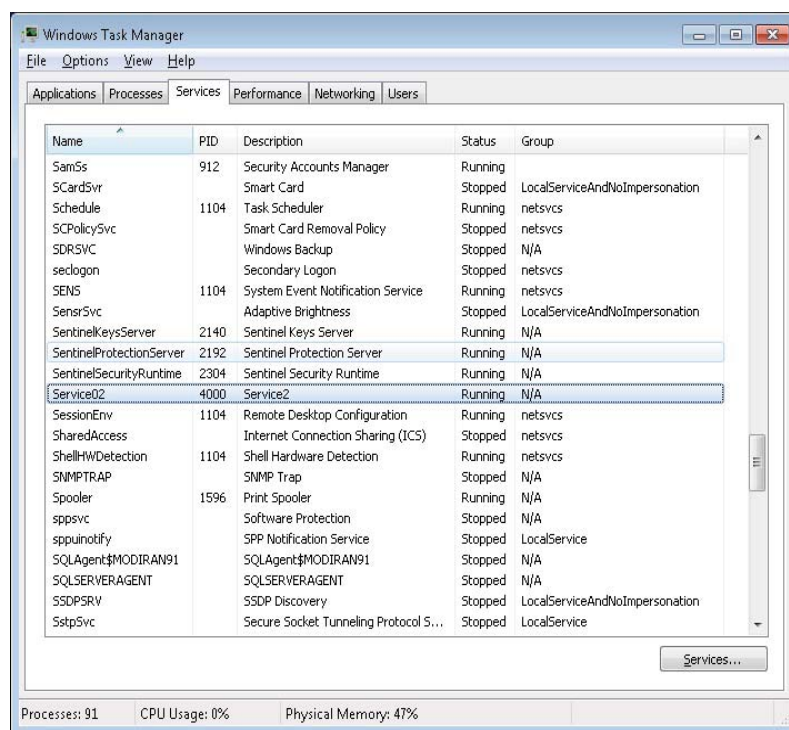
روش کار به این صورت است که یک ویندوز سرویس ایجاد می کنید ، در آن یک Timer قرار داده که مثلاً هر ۱ ساعت یکبار دیتابیس کاربران را چک نماید و لیست کاربرانی که اعتبار آن ها از ۵۰۰ ریال کمتر است واکشی کند ، پس از واکشی ، با یک حلقه به همه آن ها ایمیل ارسال می شود

قابلیت های ویندوز سرویس ها :

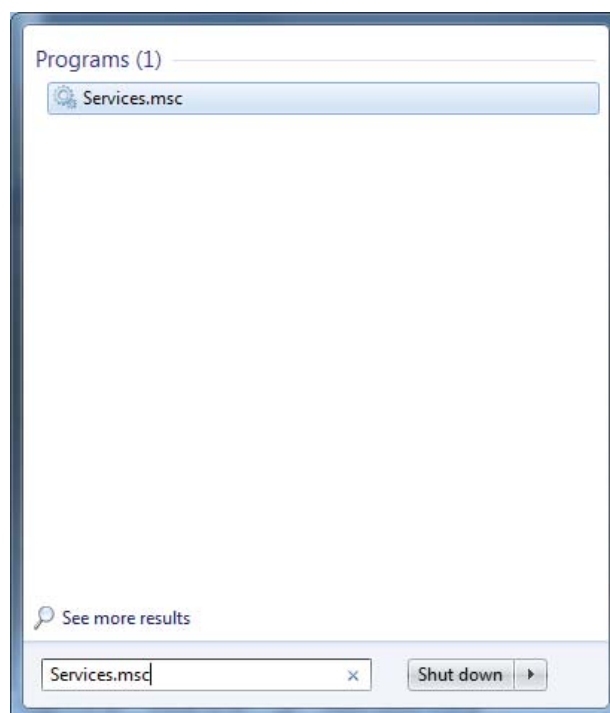
نام متد	کارایی
Close	قطع ارتباط نمونه کلاس ServiceController از سرویس و بازگرداندن منابع
Continue	ادامه کار سرویس پس از توقف یا pause
ExecuterCommand	اجرای دستورات خاص در یک سرویس
Pause	توقف موقت سرویس
Refresh	بازآوری اطلاعات سرویس
Start	شروع به کار سرویس
Stop	متوقف کردن سرویس

روش های مشاهده پنجره سرویس های ویندوز (Services):

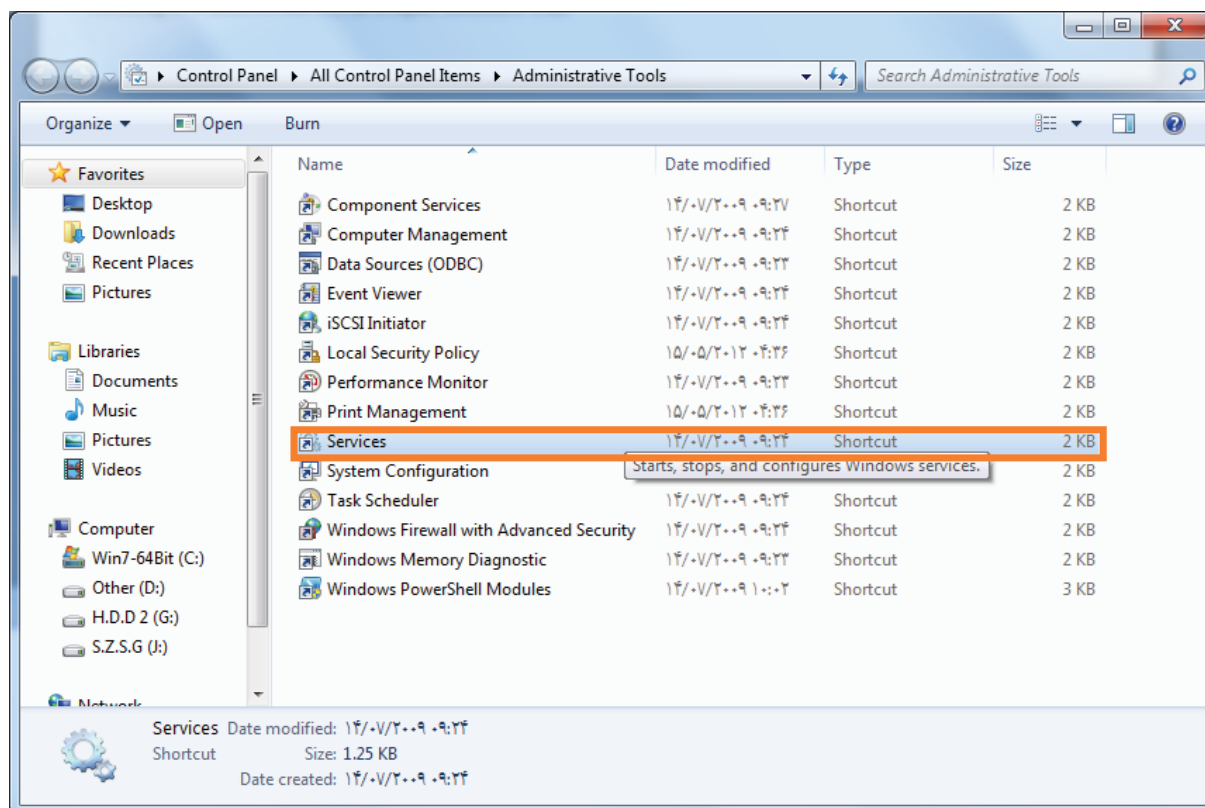
۱- با زدن کید های ترکیبی Ctrl – Alt – Del



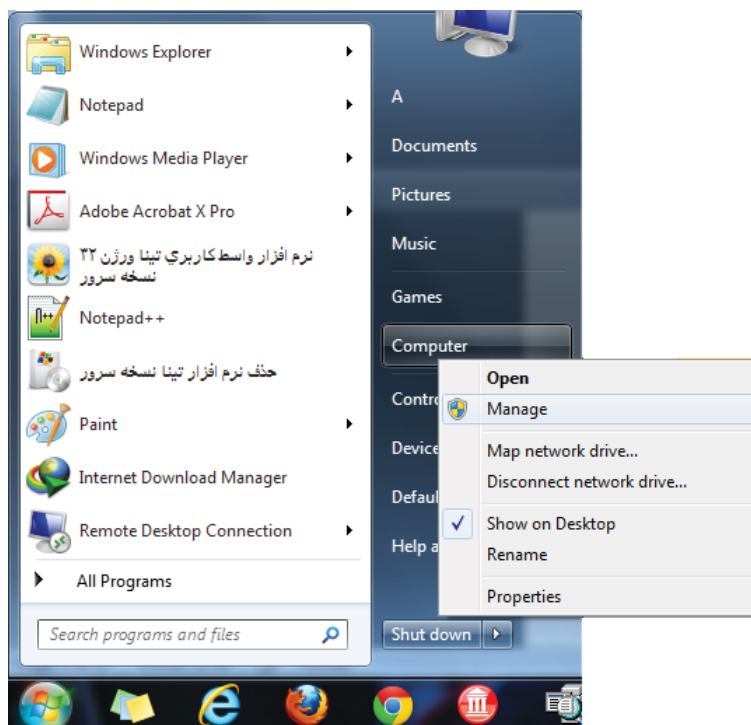
۲- با تایپ services.msc در قسمت RUN و زدن کلید Enter

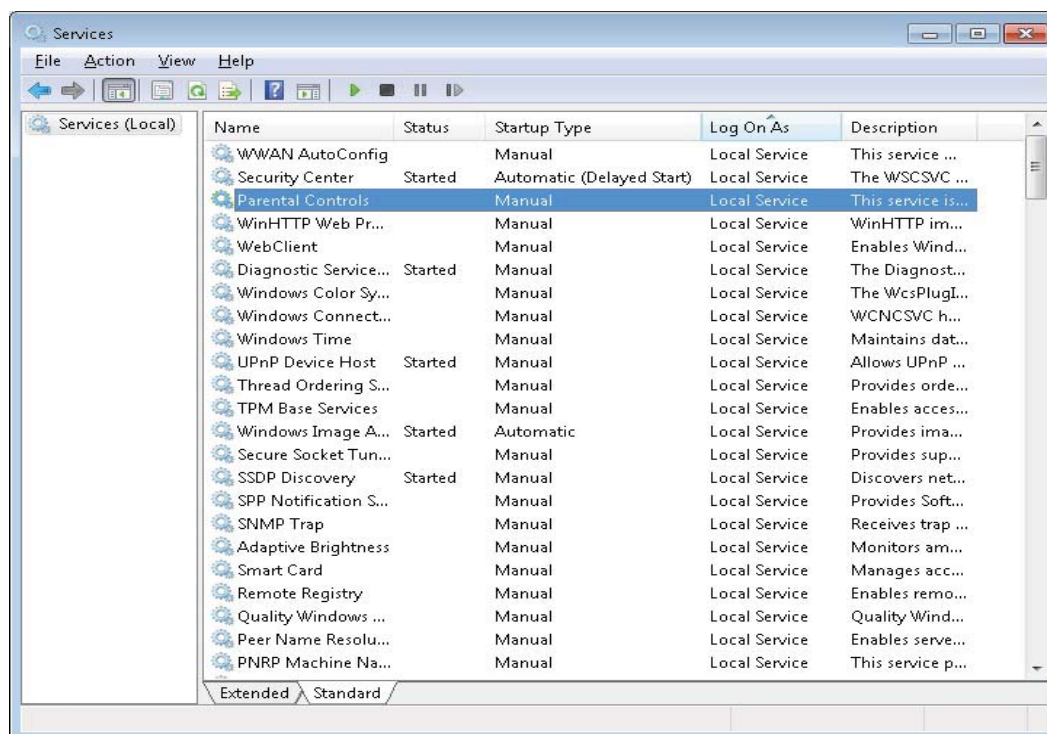
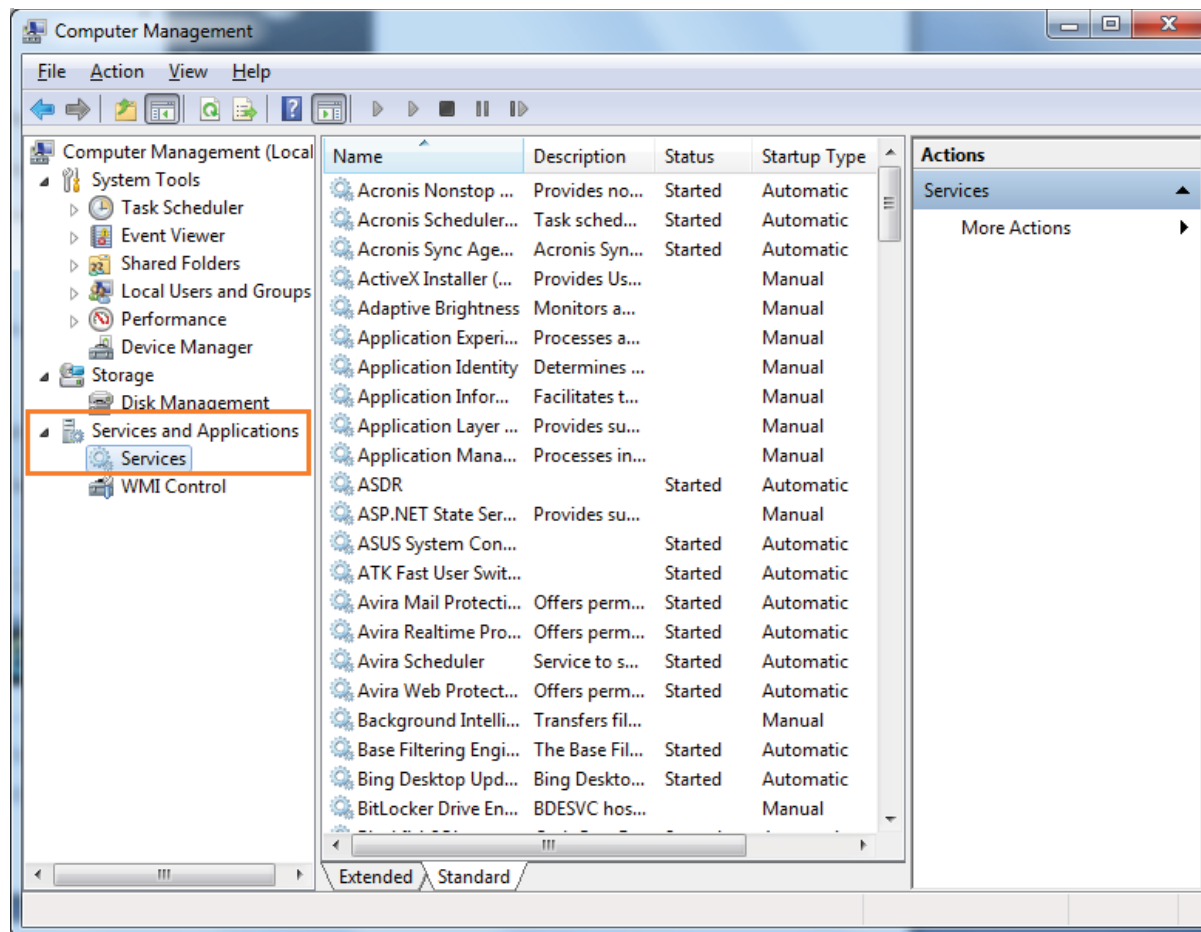


۳- با رفتن به قسمت Control Panel\Administrative Tools\Services



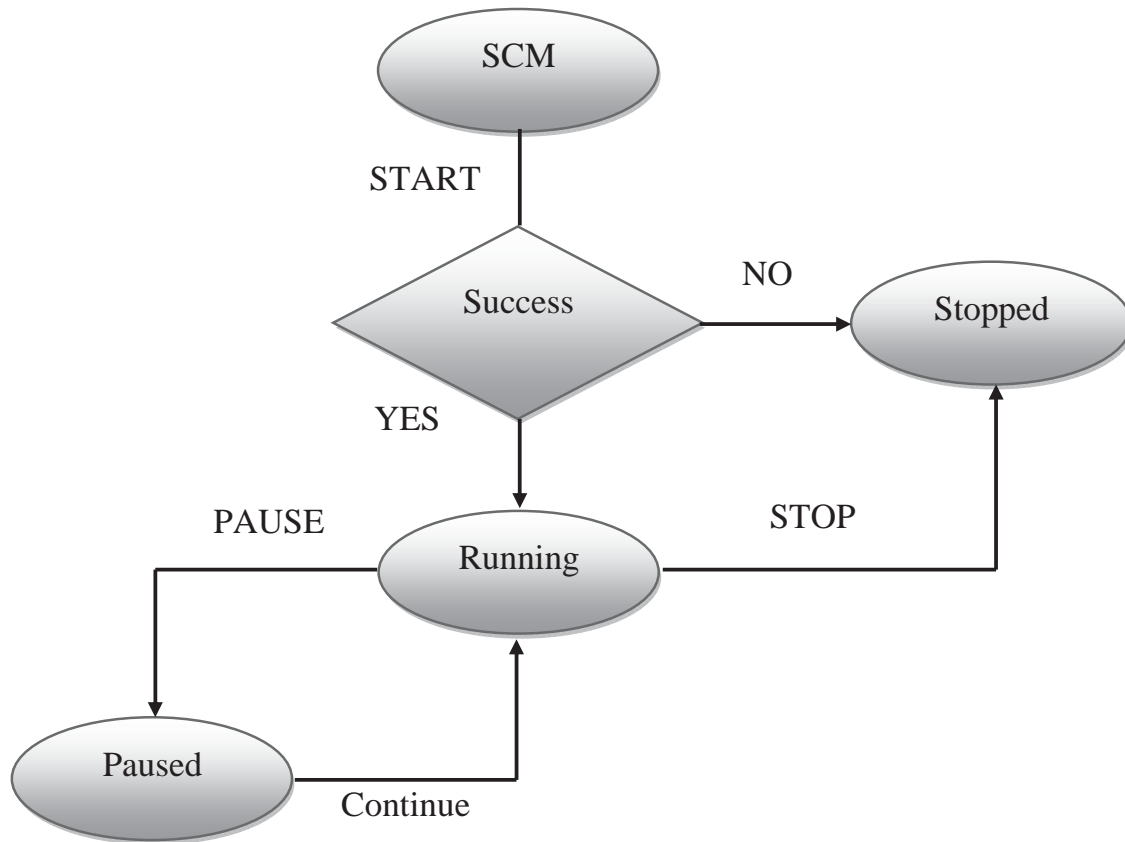
۴- Right Click In My Computer\Manage(Computer Management)\Services And Applications\Services





: Service Control Manager

چون سرویسها در Background اجرا می شوند نیازی نیست برای اجرای اکثر آنها کاربر با یک حساب کاربری خاصی به سیستم Login کند و بر اساس نوع کاربر سرویس خاصی اجرا شود. سرویسها توسط SMC (Service Control Manager) اجرا و کنترل می شوند. شکل زیر چرخه اجرای سرویسها و نحوه رد و بدل شدن Command های SMC را در بین سرویسها نشان می دهد.



The life cycle of a Windows service

ساخت ، نصب و اجرای سرویس :

ساختن ویندوز سرویس را در چند مرحله پی می گیریم :

۱- روش ساخت یک سرویس از طریق Command :

"نام مجازی سرویس" Display Name = "آدرس فایلی که قرار است به سرویس تبدیل شود" binPath = "نام سرویس" SC Create

SC Create "Service1" binPath = "c:\p\c.exe" Display Name = " Service1"

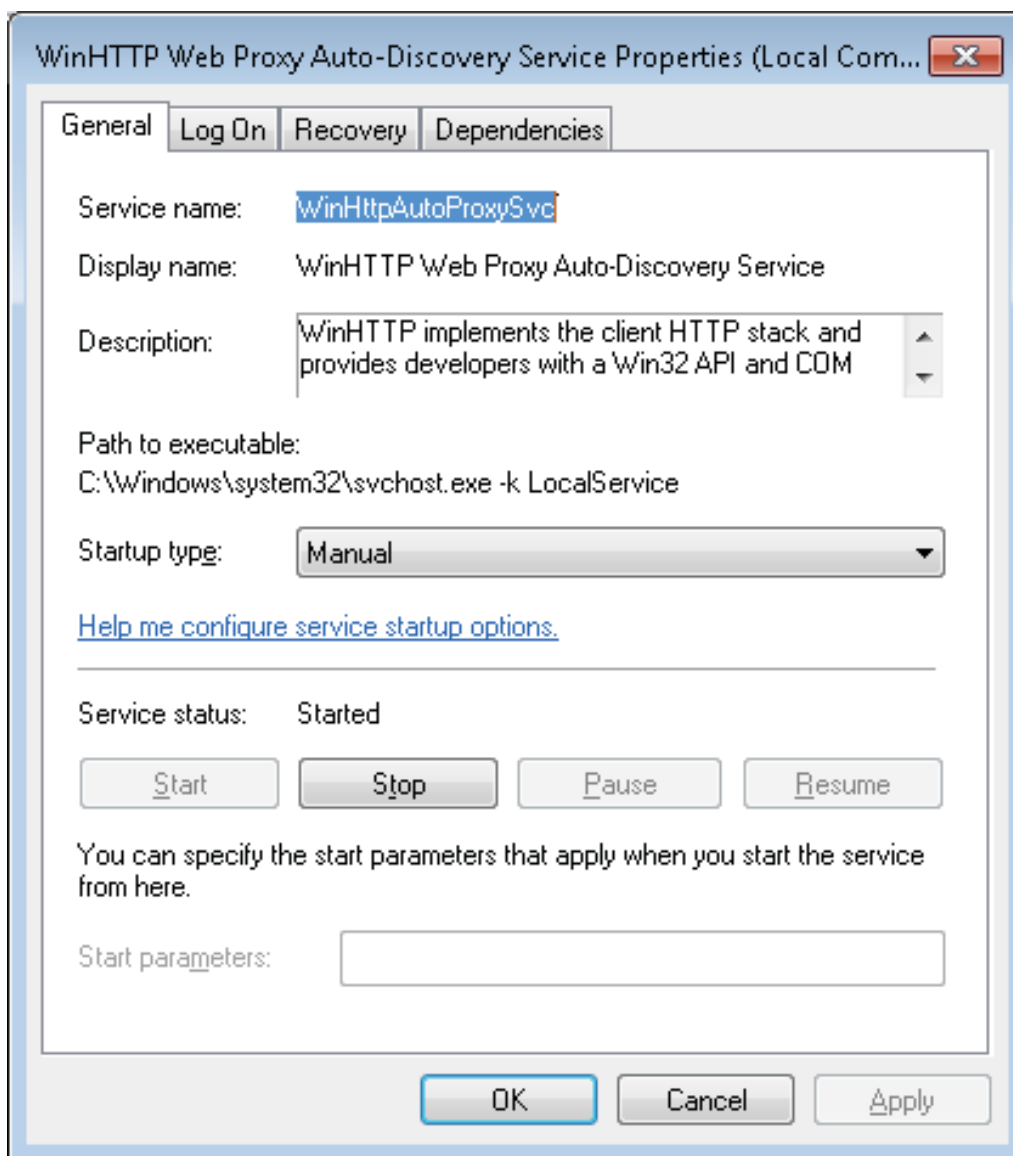
۲- روش حذف سرویس از طریق Command :

SC Delete "نام سرویس"

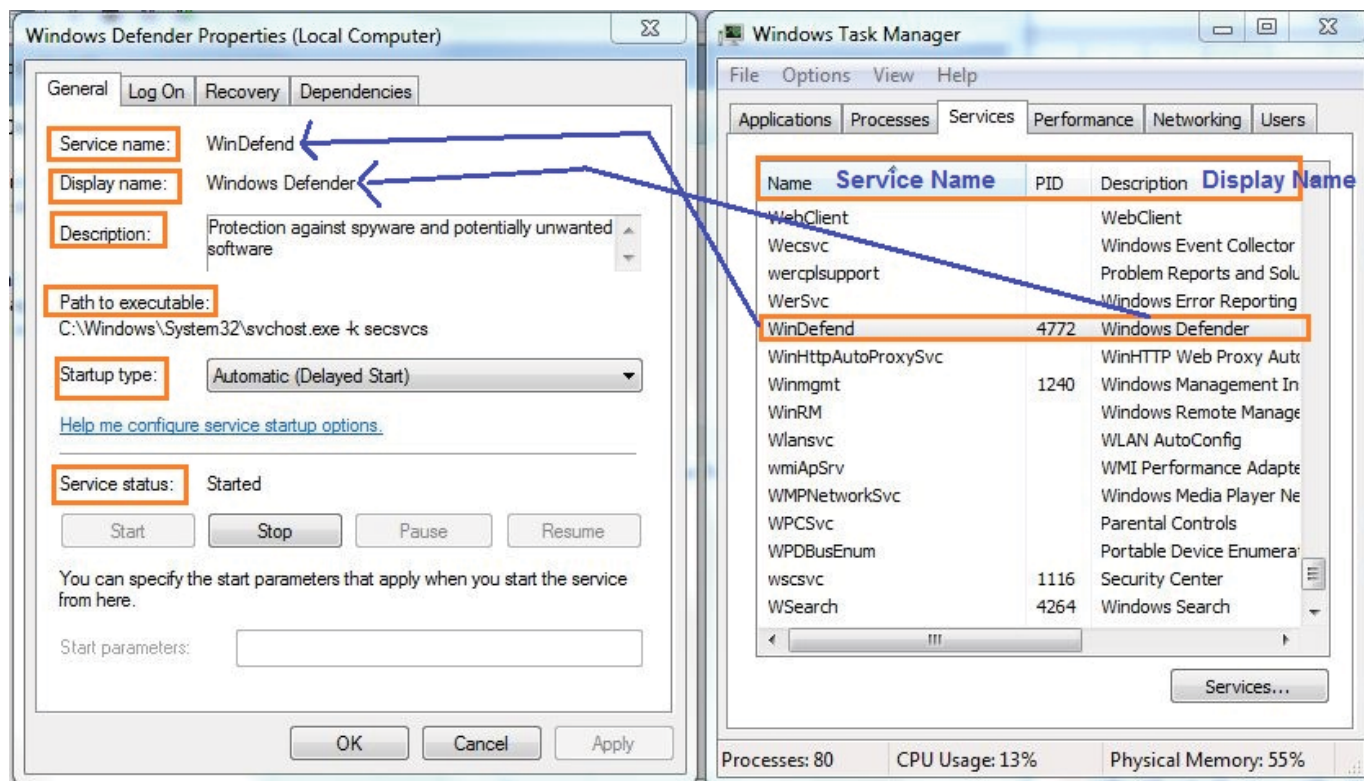
SC Delete "Service1"

۳- بررسی خصوصیات مختلف یک سرویس :

روی یکی از سرویس ها راست کلیک کرده گزینه Properties را انتخاب می کنیم یا روی آن سرویس دابل کلیک کنید .



توضیح تب General :



Service Name -۱

در اصل نام اصلی سرویس می باشد ، همان نامی ست که در Task Manager\Services در Tab ای به نام Name مشخص شده

Display Name -۲

همان نامی ست که در Task Manager\Services در Tab ای به نام Description مشخص شده ، نام مجازی سرویس است که در Service Control Manager یا Task Manager نمایش داده می شود ، هر اسمی می توانید در این قسمت قرار دهید .

Description -۳

توضیحاتی درباره سرویس در این قسمت می توان نوشت

Path to executable -۴

آدرس فایل اجرایی (exe) که سرویس از آن استفاده می کند

Startup Type -۵

می توان یکی از سه حالت زیر رو انتخاب کرد

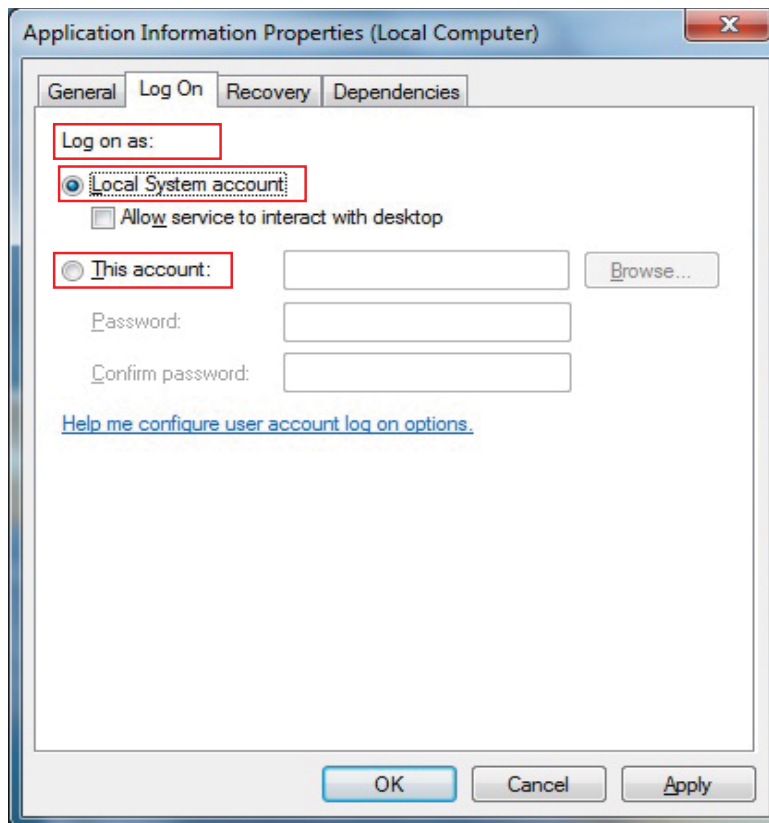
Manual, Automatic, Disable

Service Status -۶

وضعیت ، حالت سرویس رو مشخص می کنه که ۴ حالت داره

Stop, Start, Resume, Pause

توضیح تب Log On :



برای تعیین اینکه سرویس مورد نظر ما برای چه کاربر یا کاربرانی در هنگام ورود به سیستم (LogON) اجرا شود از این تب استفاده می شود. در این قسمت تنظیمات مربوط به Account های راه انداز سرویس انجام می شود و حسابهایی (Accounts) که سرویسها تحت آن اجرا می شوند را نشان می دهد.

Account های راه انداز سرویس در حالت کلی به دو دسته زیر تقسیم می شوند :

۱) System Account ها (اکانت های سیستمی)

۲) User Account ها (اکانت های کاربری)

محل تنظیمات این Account ها به ازای هر کدام از سرویس ها برنامه Services می باشد . به پنجره بالا با دقت نگاه کنید مشاهده خواهید کرد که تنظیم Account ها برای سرویس ها به دو صورت Local System Account و This Account می باشد . با انتخاب Local System Account می توانید با استفاده از Account های سیستمی سرویس مورد نظر خود را راه اندازی کنید.

- System Account ها (اکانت های سیستمی)

در قسمت زیر لیستی از Account های سیستمی را به همراه توضیحاتشان می توانید مشاهده کنید.

۱- Local System Account :

این Account دسترسی کامل به منابع سیستم محلی را دارد و به عنوان یک اکانت داخلی برای ویندوز شما هست که در حد Admin ، به تمام کامپیوترهای شبکه شما دسترسی دارد .
یک اکانت با مزایای گسترده بر روی کامپیوتر Local. توسط این اکانت می توان با میز کار (Desktop) در تعامل بود ولی با این اکانت نمی توان به شبکه دسترسی داشت ، این Account با نام NT Authority\System نیز شناخته می شود . هر سرویسی با این Account راه اندازی شود قدرتش در حد Local Admin خواهد بود بنابراین وقتی شما سرویسی را با استفاده از این Account راه اندازی کنید سرویس شما قدرت انجام هر کاری در حد Local Admin خواهد داشت می تواند فایل های سیستم را حذف و ... کند یا اگر یک Hacker بتواند سرویس را در دسترس بگیرد به راحتی می تواند ... (کار بد)

۲- Local Service Account :

این Account دسترسی محدود به منابع سیستم محلی را داشته و امکان ارتباط با کامپیوترهای راه دور را نخواهد داشت .
این Account با نام NT Authority\LocalService نیز شناخته می شود.

۳- Network Service Account :

این Account دسترسی کامل به منابع سیستم محلی را نداشته و امکان ارتباط با کامپیوترهای راه دور را در حد گروه (User) خواهد داشت. مزایای کمتری بر روی کامپیوتر Local دارد. ولی می تواند به منابع شبکه دسترسی داشته باشد تا به هر کس یا گروهی خاص اجازه دسترسی به منابع شبکه را بدهد . این Account با نام NT Authority\NetworkService نیز شناخته می شود.

- User Account ها (اکانت های کاربری)

اما اگر بخواهید سرویس های خود را با استفاده از یک User خاص راه اندازی کنید می توانید با استفاده از گزینه This Account این کار را انجام دهید. کافی است پس از انجام این کار نام کاربری و رمز عبور Account مورد نظر خود را در کادرهای User Name و Password تایپ نمایید Account های مشخص شده در این قسمت با توجه به سطح دسترسی تعریف شده امکان دسترسی به منابع محلی و سایر کامپیوترهای راه دور را دارند.

بر اساس مجوز دسترسی که تعریف شده می تواند سرویس به صورت Local یا Network domain اجرا شود

در صورتی که هر کدام از سرویس های شما با استفاده از یکی از Account های خودتان تعریف کرده اید راه اندازی شود باید به نکات زیر توجه کنید :

- ✓ Account تعریف شده باید اجازه راه اندازی سرویس را داشته باشد
- ✓ در صورتیکه Password مربوط به Account عوض شود پس از اولین Restart ، (چه سرویس و چه ویندوز) سرویس شما دیگر اجرا نخواهد شد. شما باید Password جدید را در پنجره Service Properties وارد کرده و سرویس را مجدداً راه اندازی نمایید

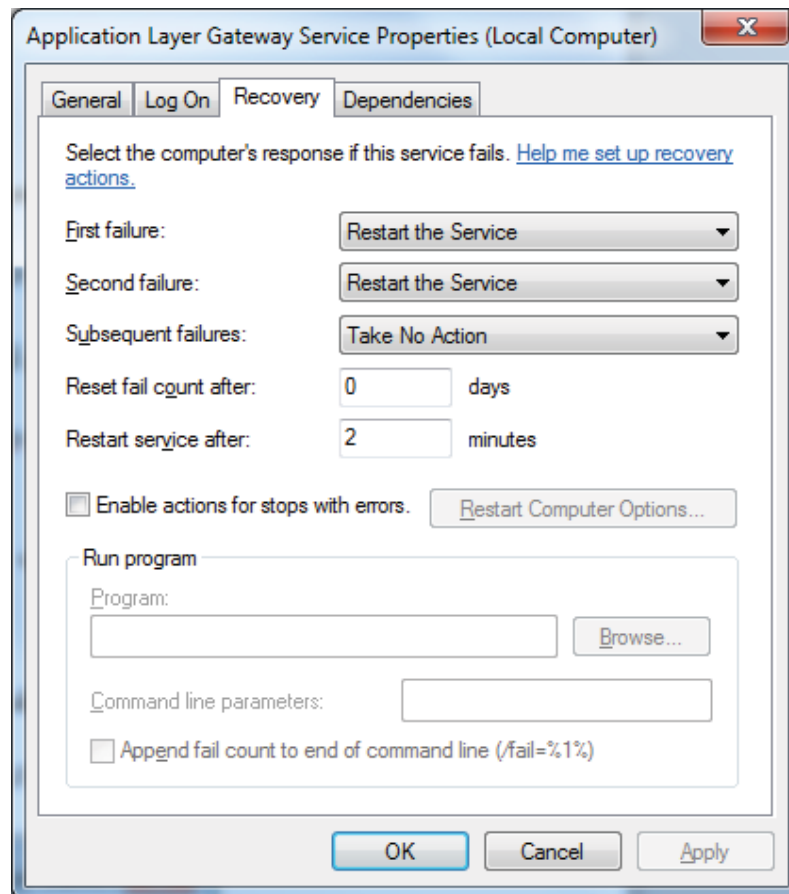
توضیح راجع به Allow Service to interact With Desktop :

If you allow the service to interact with the desktop, any information that the service displays on the desktop will also be displayed on an interactive user's desktop. A malicious user could then take control of the service or attack it from the interactive desktop.

طبق تعریف سرویس ، سرویسها برنامه هایی هستند که در Background اجرا می شوند و اکثراً با کاربر در تعامل نمی باشند ولی می توانند شامل المانهایی از رابط کاربر (با محدودیت طراحی) باشند. اگر سرویسی قرار است با Desktop در ارتباط باشد لازم است که تحت اکانت Local System اجرا شود تا به آن اجازه ارتباط با Desktop داده شود و همچنین باید گزینه Allow service to interact with desktop در تگ Logon پنجره Service Properties به حالت انتخاب در بیاوریم.

اجازه معرفی سرویس به عنوان یک سرویس تعاملی با دسکتاپ را به شما می دهد

توضیح تب Recovery :



در صورتی که سرویس دچار مشکل شد با تنظیمات این قسمت می توان واکنش هایی که قرار است در این زمان اتفاق بیافتد را تنظیم کرد .

۱- First Failure :

زمانی که سرویس انتخابی ما برای اولین بار دچار خطا شود ، یکی از * ۴ حالت انتخاب شده زیر فعال می شود.

۲- Second Failure :

زمانی که سرویس انتخابی ما برای دومین بار دچار خطا شود ، یکی از * ۴ حالت انتخاب شده زیر فعال می شود.

۳- Subsequent Failures :

زمانی که سرویس انتخابی ما برای دفعه بعدی دچار خطا شود ، یکی از * ۴ حالت انتخاب شده زیر فعال می شود.

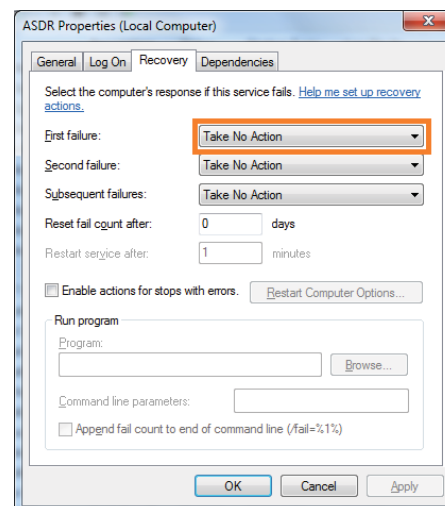
۴- Reset Fail Count After...Days :

تنظیم مجدد سرویس پس از ... روز

* ۴ حالتی که زمان خطای سرویس فعال می شود :

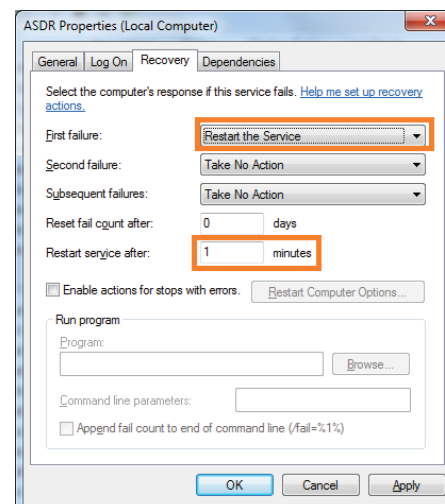
۱. Take No Action

در این حالت ، در صورت بروز خطا برای سرویس ، کاری انجام نمی شود.



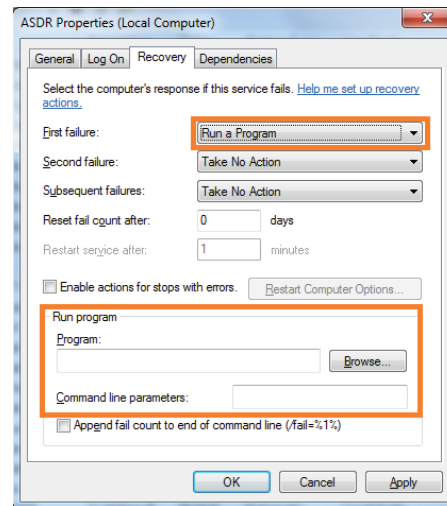
۲. Restart The Service

در این حالت ، در صورت بروز خطا برای سرویس ، سرویس بعد از ... دقیقه ریستارت می شود.



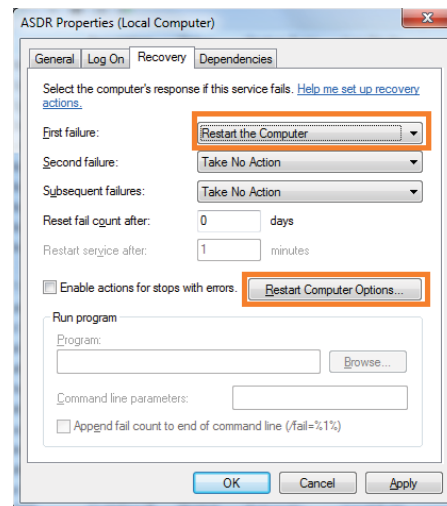
۳. Run A Program

در این حالت ، در صورت بروز خطا برای سرویس ، برنامه (فایل اجرایی exe) انتخابی ما اجرا می شود ، این برنامه میتواند با یک پارامتر اجرا شود.



۴. Restart The Computer

در این حالت ، در صورت بروز خطا برای سرویس ، سیستم ریستارت می شود

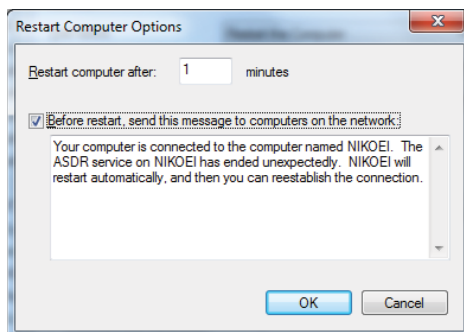


با دکمه ... Restart Computer Option ، می توان تعیین کرد که چند دقیقه بعد از رویدادن خطا برای سرویس ، سیستم

ریستارت شود ؟ و با تیک زدن گزینه

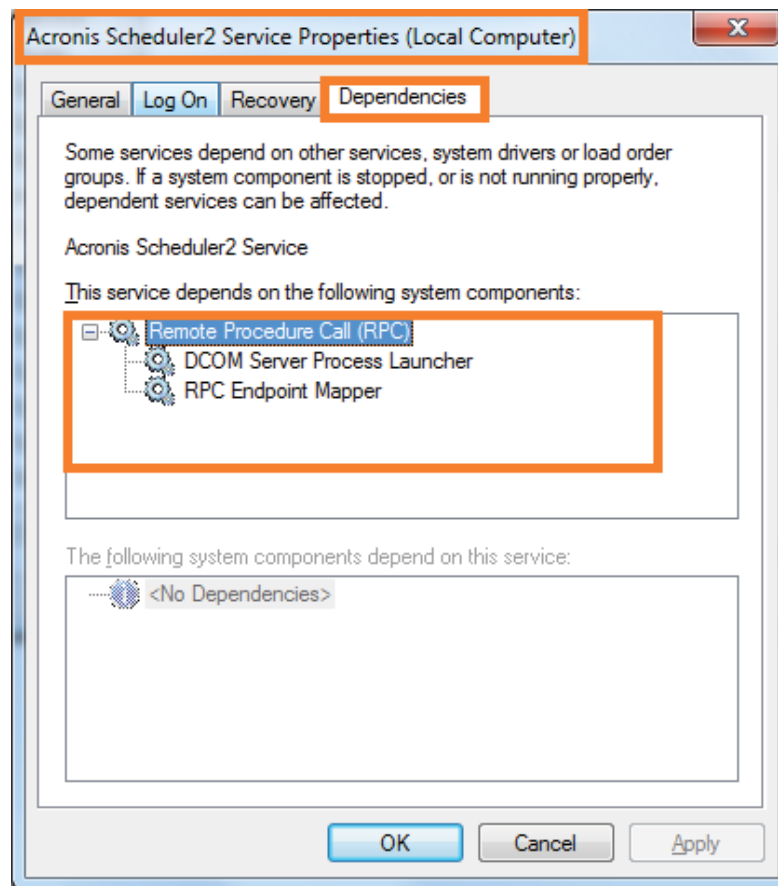
Before restart, Send this message to computers on the network ، قبل از ریستارت شدن سیستم ، پیغام وارد شده در تگست باکس برای

همه کامپیوترهای موجود در شبکه فرستاده شود .



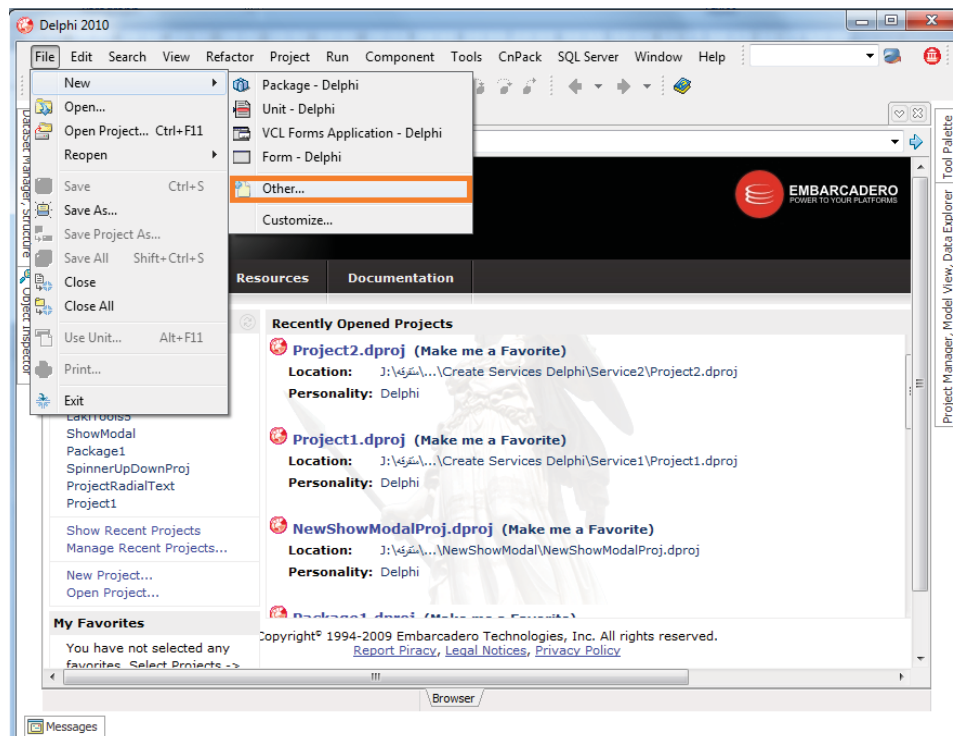
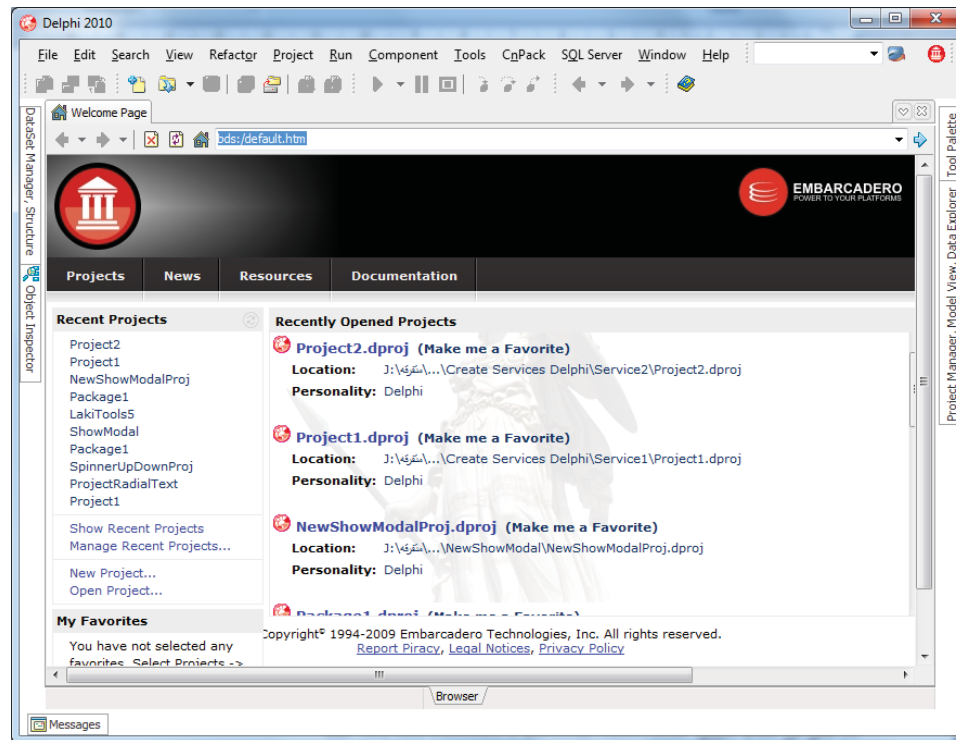
توضیح تب Dependencies :

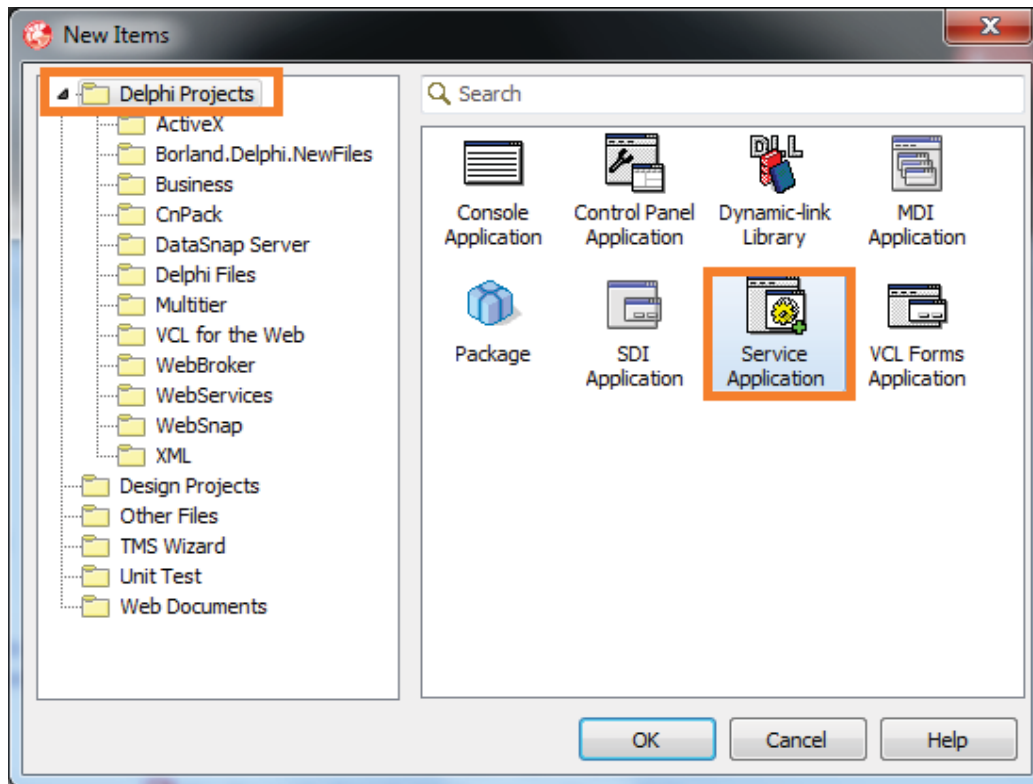
برخی از سرویس ها به سرویس های دیگر وابسته هستند ، مانند درایور های سیستمی .
در صورتی که یک سرویس متوقف شود یا اصلاً اجرا نشده باشد ، سرویس های وابسته اش تحت تاثیر آن سرویس قرار می گیرند .
در این تصویر ، سرویس Acronis ، وابسته به سرویس (Remote Procedure Call (RPC)) می باشد .



ساخت ، نصب و اجرای سرویس توسط دلفی ۲۰۱۰:

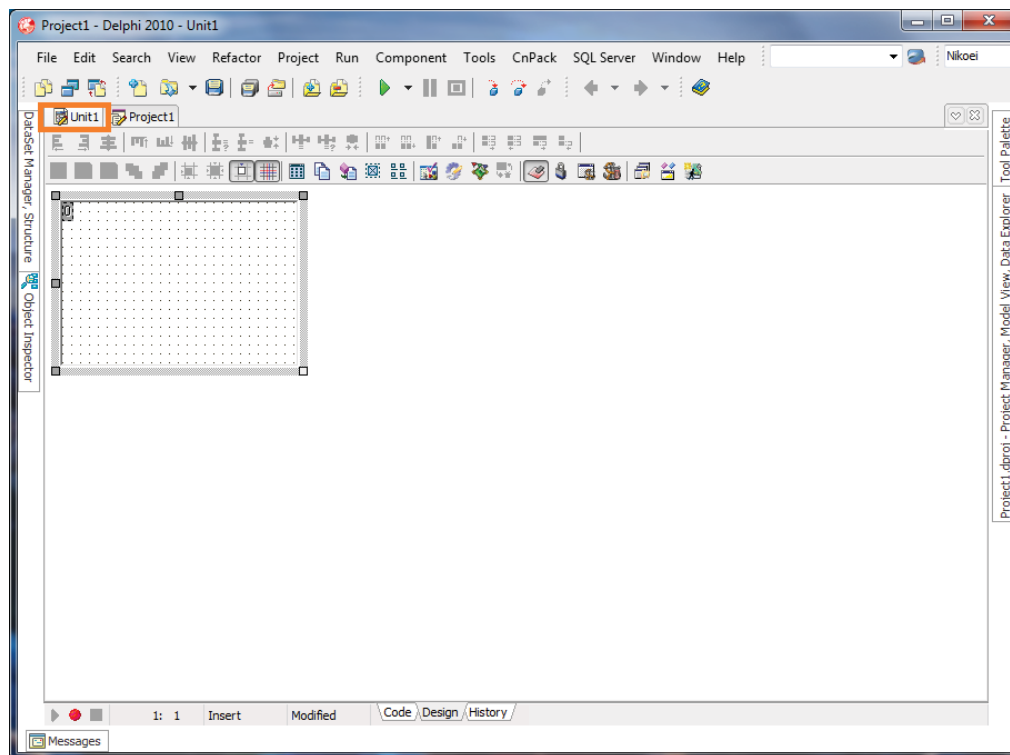
برای اضافه کردن یک Windows Service نرم افزار Delphi 2010 را باز می کنیم و سپس وارد منوی File > New > Other شوید ، از آیتم های سمت چپ Delphi Projects را انتخاب کنید و سپس از سمت راست گزینه Service Applications را انتخاب کنید و روی آن دابل کلیک کنید .

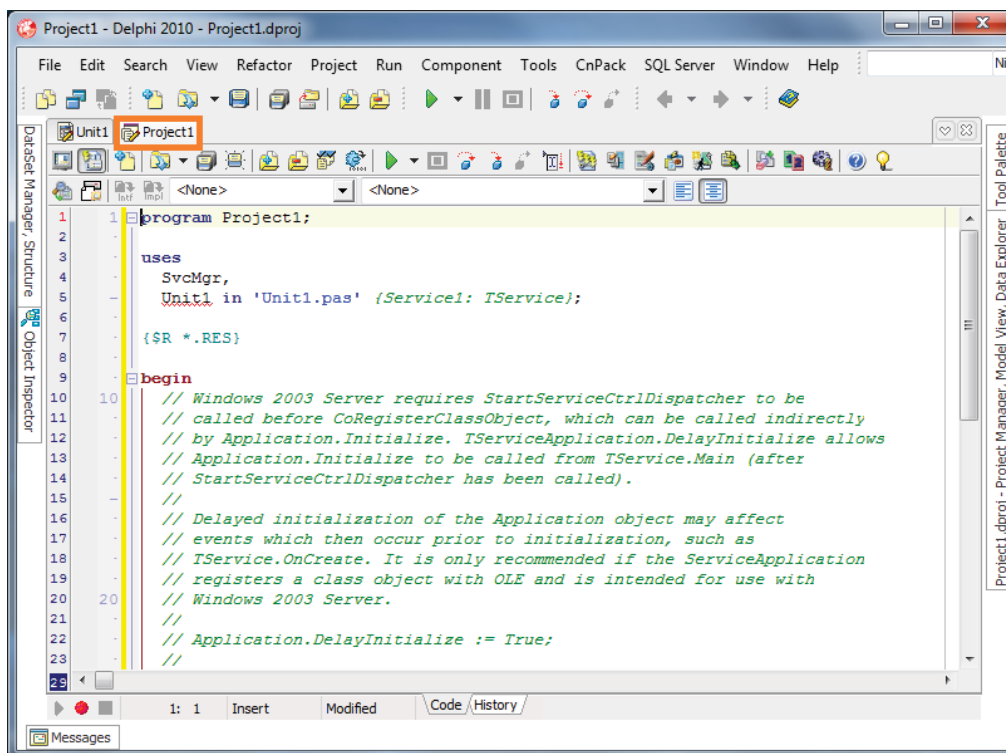




به این ترتیب یک پروژه سرویس در دلفی ایجاد شد .

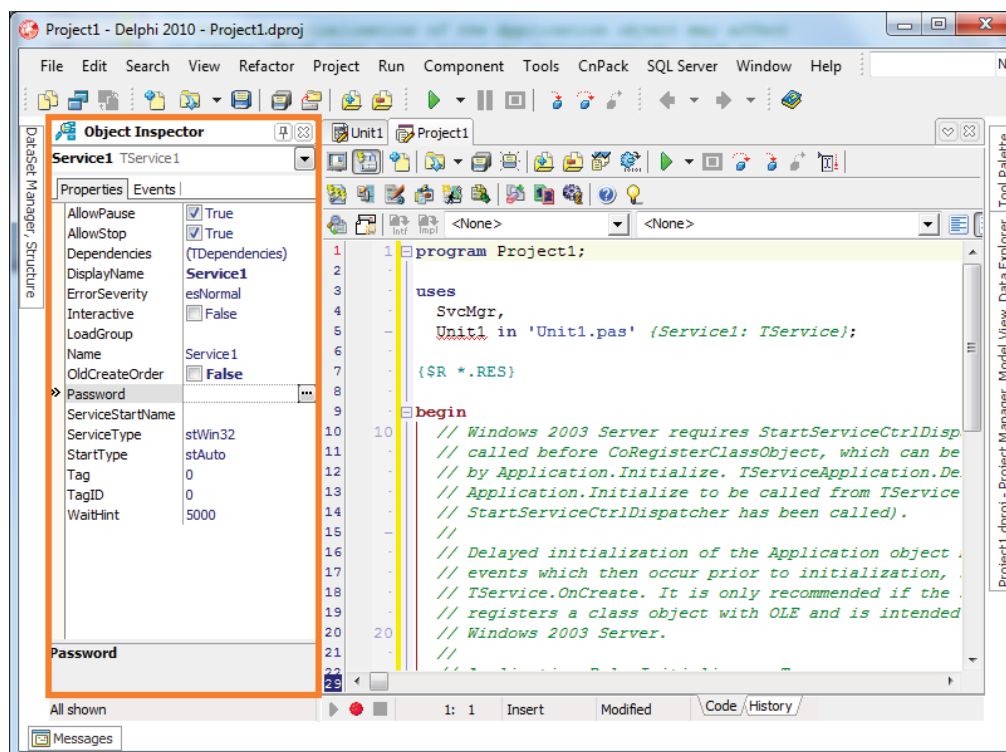
یک یونیت به نام Unit1.pas و یک پروژه به نام Project1.dpr ایجاد می شود :





اکنون پروژه را Save (ذخیره) می کنیم . با انتخاب File > Save Project AS

حال به توضیح مختصری در رابطه با تنظیم خصوصیات این سرویس می پردازم (object Inspector):



توضیحات راجع به object Inspector :

۱. AllowPause و AllowStop = TRUE

مجوز توقف لحظه ای و کامل برای سرویس را می دهد

۲. Dependencies

قبلاً توضیح دادم

۳. Error Severity

نحوه برخورد با خطای احتمالی از سرویس :

فکر کنم مثلاً ،

esCritical : دست از برخورد بر نمی داره تا بلاخره یا کاری از پیش ببره و خطا رفع بشه یا سیستم هنگ کنه

esNormal : برخورد با خطا همراه با عطلت

esIgnore : خطا رو نادیده می گیره

esSevere : به شدت با خطا برخورد می کنه شاید در حد برخورد فیزیکی (تنبیه بدنی)

۴. Interactive = TRUE

مشخص می کند سرویس توانایی ارتباط با دسکتاپ را داشته باشد یا خیر؟

به فرض مثال = سرویسی که در آن از ShowMessage استفاده شده باشد ، این به نوع سرویسی که می خواهیم ایجاد کنیم

بستگی داره ، برای سرویس های معمولی که ما نیازی به ارتباط و تعامل با دسکتاپ نداریم نیازی به فعال کردن این قسمت نیست

۵. LoadGroup

۶. Name

همان نامی ست که در TaskManager\Services در Tab ای به نام Name مشخص شده

۷. OldCreateOrder

۸. Password و ServiceStartName (همانند Log On As هستش که قبلاً توضیح دادم)

مشخص کننده حساب کاربری و رمز عبور کاربری هست که سرویس بایستی در توسط آن اجرا شود

مثل LocalSystem و A. با پسورد ۱۰

LocalSystem. Pass = "

.\A Pass = 10

NT AUTHORITY\LocalService. Pass = "

NT AUTHORITY\NetworkService. Pass = "

ServiceType ۹.

مقدار این خصوصیت = stWin32 است برای سرویس های Win32

StartType ۱۰.

می تونه یکی از انتخاب های زیر باشه stAuto, stBoot, stDisabled, stManual یا stSystem ولی در سرویس های

Win32 فقط می توان از stAuto, stManual یا stSystem استفاده کرد

Tag ۱۱.

TagID ۱۲.

WaitHint ۱۳.

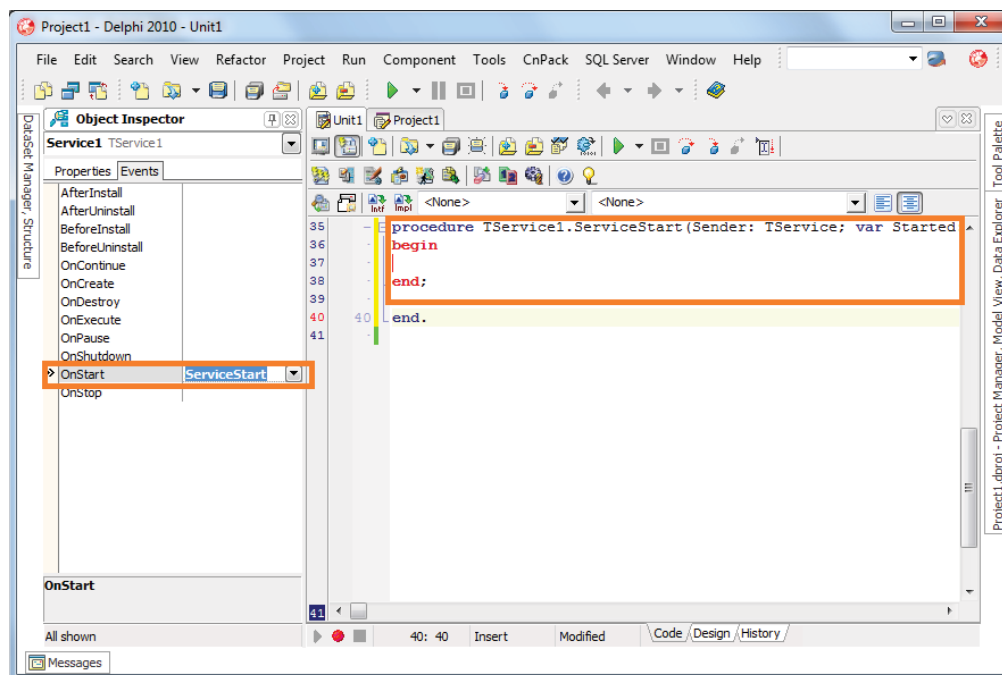
پروژه شماره ۱ -

می خواهیم سرویسی ایجاد کنیم که پس از نصب شدن آن ، زمانی که سرویس متوقف شد ، تاریخ و ساعت توقف آن را در یک Log Text در درایور C ذخیره کند .

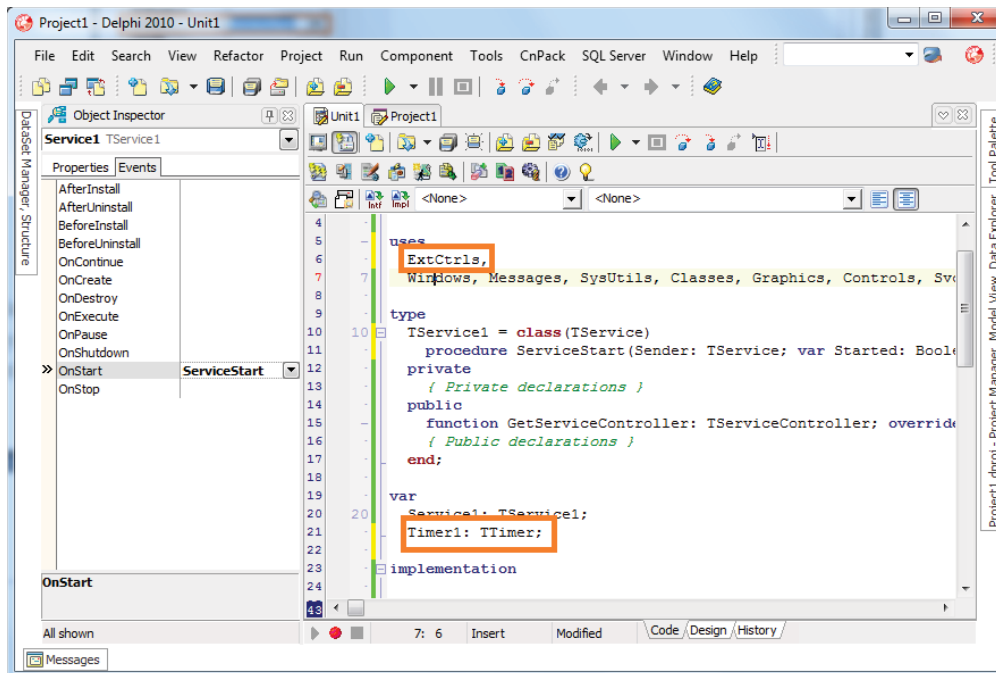
در ساخت این سرویس ، نیاز به یک تایمر داریم ، می دانید که استفاده از تایمر بسیار ساده است کفایت یک کنترل از نوع تایمر در صفحه قرار داده و خاصیت interval آن را ست نمایید اما متاسفانه شما نمی توانید از این کنترل در Windows Service ها استفاده کنید ، این یکی از کلیدی ترین نکات ساخت سرویس های ویندوز است . بنابراین برای اضافه نمودن تایمر به صورت زیر عمل نمایید :

۱- در قسمت Object Inspector ، بر روی خاصیت On Start از سرویس مورد نظر دابل کلیک می کنیم تا پروسیجر آن نمایان شود

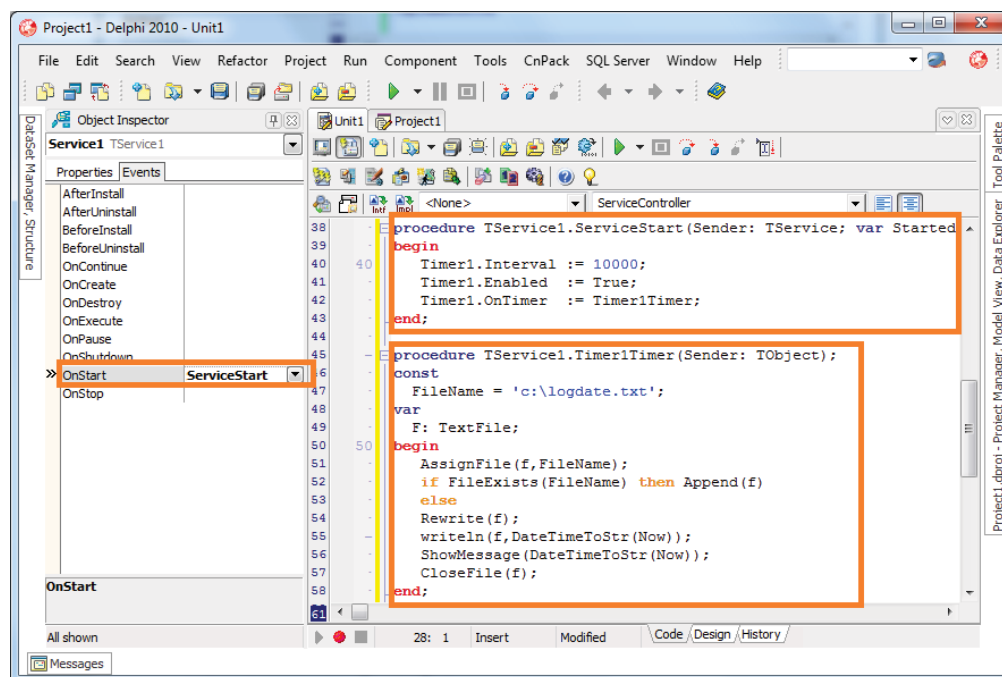
، احتمالاً کدی مشابه زیر ملاحظه خواهید کرد:



۲- یک متغیر Global از نوع Timer اضافه می کنیم ، ابتدا باید ExtCtrls را Use کرد.



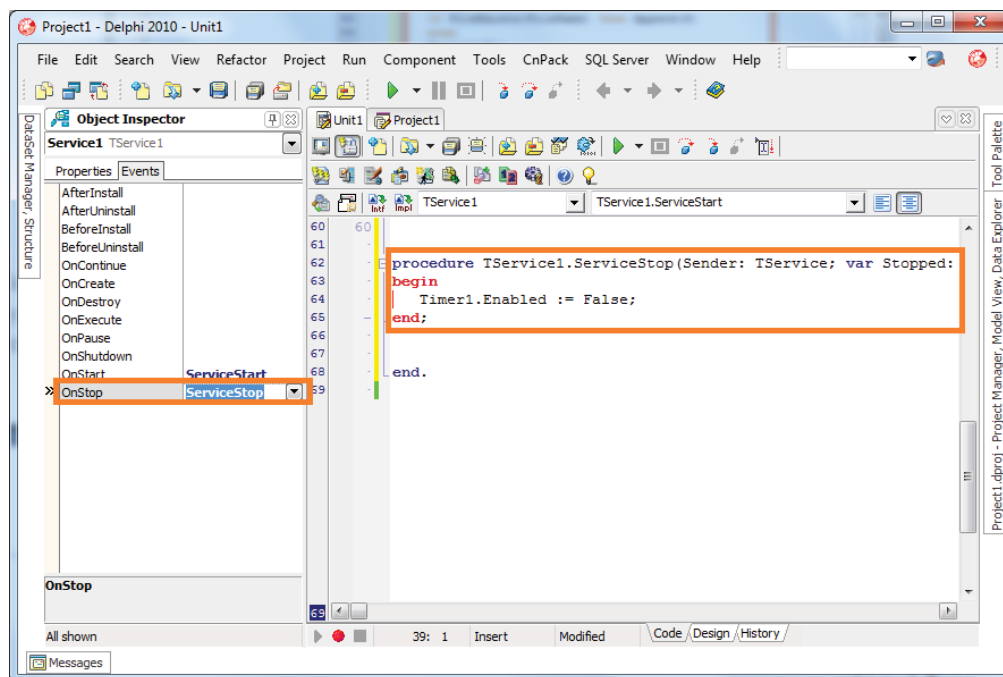
۳- در رویداد OnStart ابتدا تایمر را فعال کرده و سپس مقدار (Interval دوره بازگشت تایمر) را تعیین نمایید ، و سپس رویداد اجرا شدن هر دوره را به Timer نسبت می دهیم. (یعنی پروسیجر Timer1Timer را می نویسیم و به OnTimer از Timer1 نسبت می دهیم).



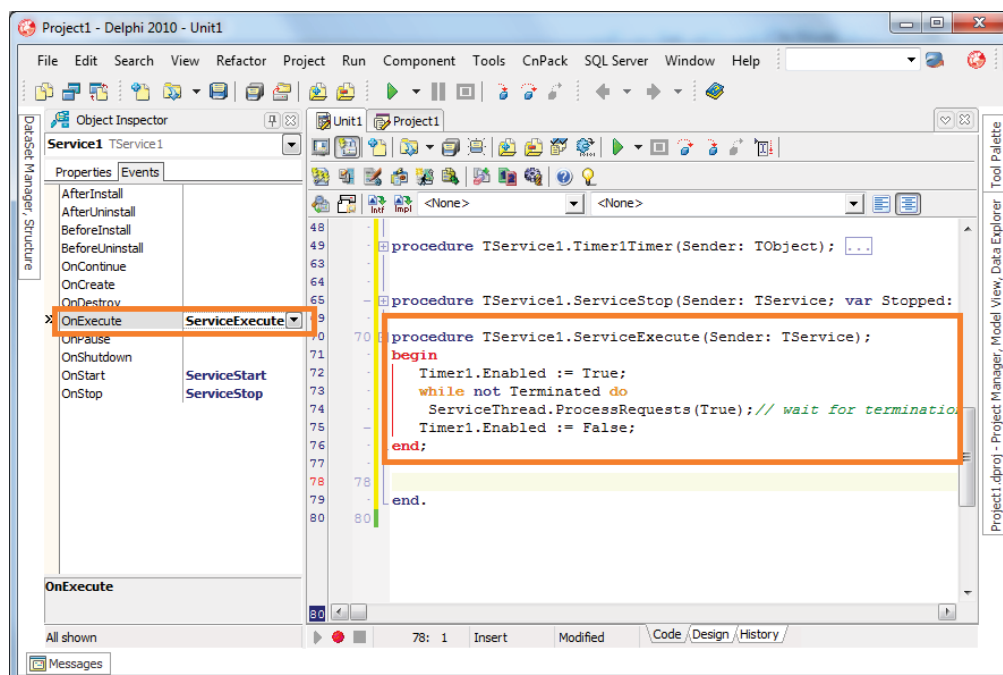
```
private
{ Private declarations }
procedure Timer1Timer(Sender: TObject);
```


نکته : خاصیت Interval از نوع int32 بوده و بر حسب میلی ثانیه می باشد ، بنابراین اگر میخواهید دوره هر ۱ دقیق یکبار انجام شود باید مقدار ۶۰۰۰۰ را ست نمایید. راستی تا یادم نرفته ، در قسمت Properties از Object Inspector ، گزینه InterActive رو مساوی TRUE قرار بدین .

۴- در رویداد OnStop تایمر را غیر فعال می کنیم



۵- در قسمت OnExecute این کد را می نویسیم :




```

1  unit Unit1;
2
3  interface
4
5  uses
6      ExtCtrls,
7      Windows, Messages, SysUtils, Classes, Graphics, Controls, SvcMgr, Dialogs;
8
9  type
10
11  TService1 = class(TService)
12      procedure ServiceStart(Sender: TService; var Started: Boolean);
13      procedure ServiceStop(Sender: TService; var Stopped: Boolean);
14      procedure ServiceExecute(Sender: TService);
15  private
16      { Private declarations }
17      procedure Timer1Timer(Sender: TObject);
18  public
19      function GetServiceController: TServiceController; override;
20      { Public declarations }
21  end;
22
23  var
24      Service1: TService1;
25      Timer1: TTimer;
26
27  implementation
28
29      {$R *.DFM}
30
31  procedure ServiceController(CtrlCode: DWord); stdcall;
32  begin
33      Service1.Controller(CtrlCode);
34  end;
35
36  function TService1.GetServiceController: TServiceController;
37  begin
38      Result := ServiceController;
39  end;
40
41  procedure TService1.ServiceStart(Sender: TService; var Started: Boolean);
42  begin
43      Timer1.Interval := 10000;
44      Timer1.Enabled := True;
45      Timer1.OnTimer := Timer1Timer;
46  end;
47
48  procedure TService1.Timer1Timer(Sender: TObject);
49  const
50      FileName = 'c:\logdate.txt';
51  var
52      F: TextFile;
53  begin
54      AssignFile(f, FileName);
55      if FileExists(FileName) then Append(f)
56      else
57          Rewrite(f);
58      writeln(f, DateTimeToStr(Now));
59      ShowMessage(DateTimeToStr(Now));
60      CloseFile(f);
61  end;
62
63  procedure TService1.ServiceStop(Sender: TService; var Stopped: Boolean);
64  begin
65      Timer1.Enabled := False;
66  end;
67
68  procedure TService1.ServiceExecute(Sender: TService);
69  begin
70      Timer1.Enabled := True;
71      while not Terminated do
72          ServiceThread.ProcessRequests(True); // wait for termination
73      Timer1.Enabled := False;
74  end;
75
76  end.

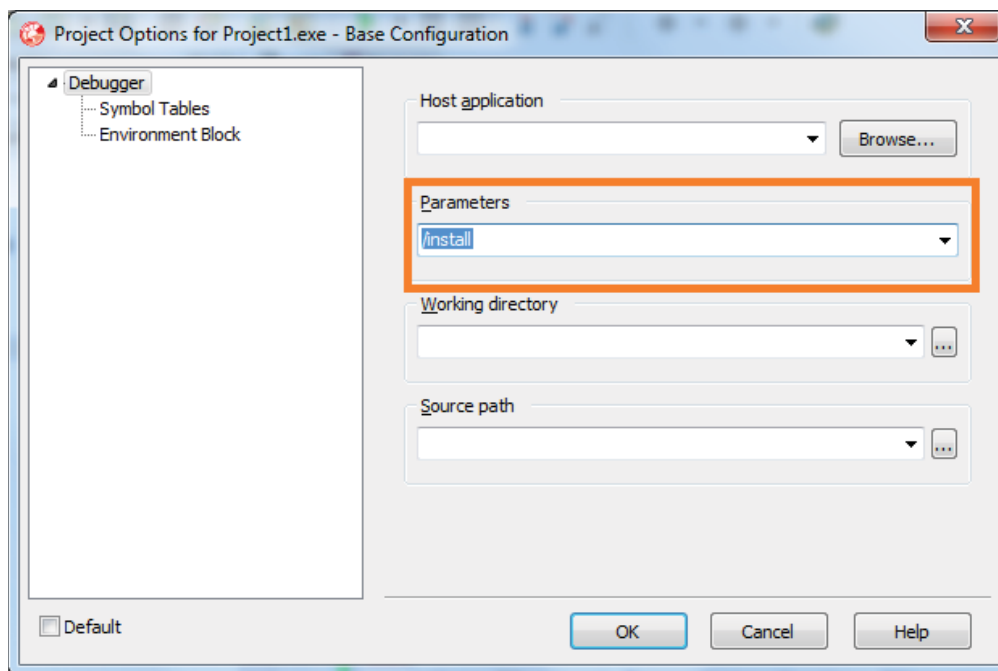
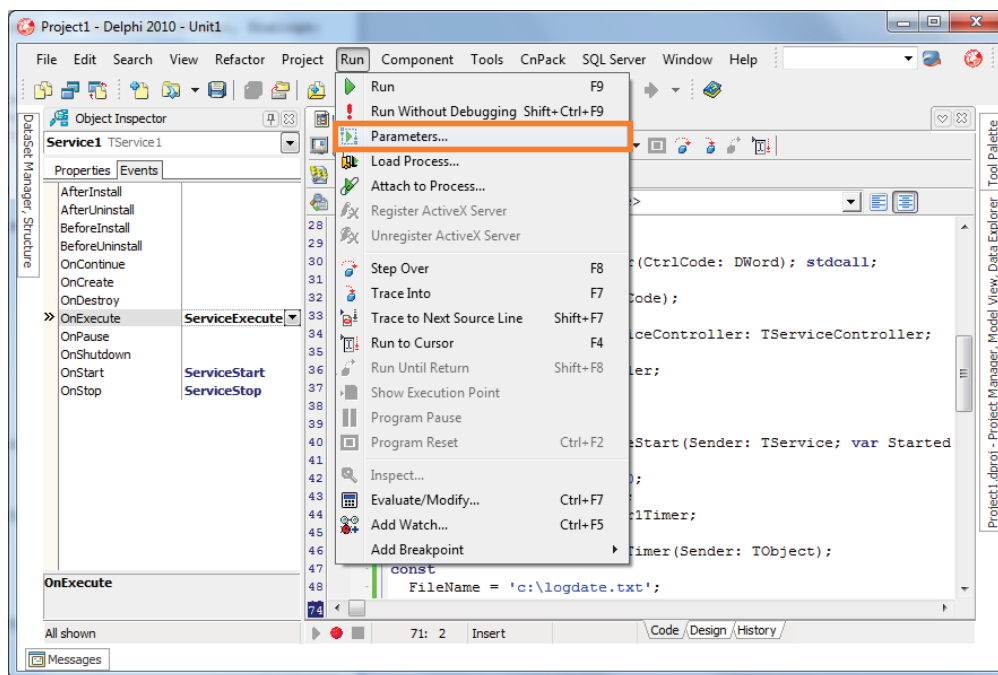
```

ویندوز سرویس شما
آماده است ، در نهایت
کد ها چیزی شبیه کد
های روبرو خواهد بود :

ایجاد نصاب ویندوز سرویس (Windows Service Installer)

۱- از طریق RUN دلفی :

از منوی RUN گزینه Parameters را انتخاب می کنیم ، سپس در قسمت Parameters ، پارامتر /install را وارد کرده سپس بر روی OK کلیک می کنیم .



سپس کلید F9 یا از منوی RUN > RUN سرویس را نصب می کنیم ، در این روش سرویس استارت نمی باشد و فقط نصب شده است.

۲- از طریق Batch File :

این کد را در یک NotePad تایپ کرده و سپس آن را با پسوند Bat. ذخیره می کنیم و کنار فایل اجرایی ، آن را اجرا می کنیم .

```
@echo off
"/Install /silent" "نام فایل اجرایی"
NET START "نام سرویس"
```

```
@echo off
Project1.exe /Install /silent
NET START Service1
```

```
@echo off
NET STOP "نام سرویس"
"/UnInstall /silent" "نام فایل اجرایی"
```

```
@echo off
NET STOP Service1
Project1.exe / UnInstall /silent
```

توجه توجه :

در صورتی که این سرویس را نصب کردید در حالیکه مدت زمان Interval تایمر را کم تنظیم کرده اید ، با مشکل بزرگی روبرو خواهید شد چرا که هر زمان که وارد سیستم می شوید تایمر به سرعت مسیح هایی به شما نشان می دهد که قابل کنترل نیستند .

راه حل :

فقط از طریق SafeMode می توان این سرویس را متوقف کرد ، چون در این حالت هسته سرویس ها که برای بالا اومدن ویندوز مورد نیاز هستند به صورت پیش فرض اجرا می شوند (بدون در نظر گرفتن هر گونه تغییر که در تنظیمات سرویس ها انجام شده است .)

وقتی سیستم به حالت SafeMode رفت ، می توان تنظیمات سرویس را تغییر داد یا سرویس را Restore کرد.

جهت کانفیگ کردن سرویس با دستورات Command این لینک را مشاهده کنید :

<http://technet.microsoft.com/en-us/library/807191da-4a33-4149-addf-c11ded938b5f>

این هم یک فانکشن که با آن می توان چک کرد که سرویس ما نصب می باشد یا خیر ؟

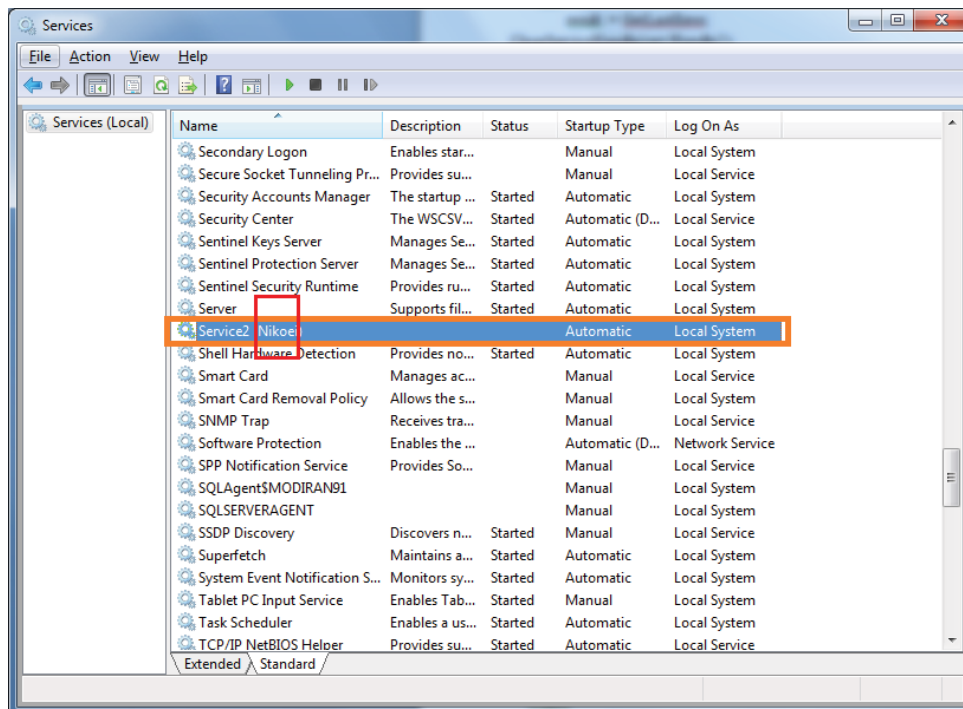
```
//- Check if service is installed -----
function checkService(servicename : String) : Integer;
var
    servHandle,
    servHandle2 : SC_HANDLE;
begin
    servHandle := OpenSCManager(nil, nil, SC_MANAGER_ALL_ACCESS);
    if servHandle > 0 then
        begin
            servHandle2 := OpenService(servHandle, PChar(Trim(serviceName)), SERVICE_ALL_ACCESS);
            if servHandle2 > 0 then
                result := 0
            else
                result := GetLastError;
        end
    else
        result := GetLastError;
    CloseServiceHandle(servHandle2);
end;
else
    result := GetLastError;
    CloseServiceHandle(servHandle);
end;
```

پروژه شماره ۲ -

تفاوت این پروژه با دوپروژه اول و سوم در این است که ، این پروژه هنگام نصب ، یک پارامتر ورودی دریافت می کند.

Project2.exe /install -instance="Nikoei" /Silent

در این مثال ، پارامتر ورودی "Nikoei" است



پروژه شماره ۳ -

یک سرویس که در هنگام نصب ، اطلاعات خود را در رجیستری ذخیره می کند ، زمانی که سرویس استارت می شود یک Thread ایجاد می کند و در قسمت Execute می توانید کد مورد نظر خودتون رو وارد کنید .

```
procedure TService3.ServiceExecute(Sender: TService);
const
  SecBetweenRuns = 10;
var
  Count: Integer;
begin
  Count := 0;
  while not Terminated do
  begin
    Inc(Count);
    if Count >= SecBetweenRuns then
    begin
      Count := 0;
      { place your service code here }
      { this is where the action happens }
      //-----SomeProcedureInAnotherUnit;
    end;
    Sleep(1000);
    ServiceThread.ProcessRequests(False);
  end;
end;
```

پروژه شماره ۴ -

<http://stackoverflow.com/questions/5561919/create-a-small-and-concise-windows-service-using-delphi>

این مقاله رو به عشق شما دوستان سایت برنامه نویس ، به پاس همکاری های بی منت تک تک تون در ساعت ۰۲:۳۷ بامداد پنج شنبه اول تیر ماه ۱۳۹۱ به پایان رسوندم . امیدوارم گوشه ای از محبت های بی دریغتون رو جبران کرده باشم.

همین جا می خوام از تعدادی از دوستان خوبم با آوردن اسمشون تشکر کنم :



GOLD



مجتبی ناجیک



hossein_h6



Mahmood_M



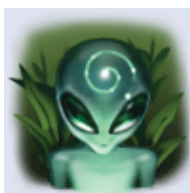
cayberfox

شاهین
عشایری alimanam

noorsoft



SAASTN



AMIBCT



You-See



علی کشاورز



Ananas



IranDelphi



Babak869 DelphiAssistant



حمیدرضا صدقیان



mbshare



Pr0grammer

References :

<http://www.majidhosseinpour.5u.com>

<http://www.Barnamenevis.org>

<http://www.nikamooz.com>

<http://social.technet.microsoft.com>

<http://stackoverflow.com>

<http://Delphi.about.com>

تصویر گروهی برنامه نویسان در نمایشگاه کتاب سال ۱۳۸۸



تصویر بنده در جشن فارغ التحصیلی سال ۱۳۸۹ (خاطرات ...)
از سمت چپ سومی عینک آفتابی زدم و نیم آستین پوشیدم (هه هه)



خدانگهدار همتون
یا علی