

در فصل دوم کتاب تا به الان یاد گرفتیم چگونه ماژول‌ها را کامپایل کنیم و چگونه آن‌ها را در یک اسمبلی قرار دهیم. حال وقت آن فرا رسیده است که با بسته بندی کردن (Package) و انتشار آن (Deploy) به طوری که کاربران بتوانند برنامه را اجرا کنند آشنا شویم.

نصب برنامه از طریق فروشگاه ویندوز

در فروشگاه ویندوز [Windows Store Apps](#) قوانین سخت و شدیدی برای بسته بندی کردن اسمبلی‌ها وجود دارد. ویژوال استودیو تمام اسمبلی‌های مورد نیاز برنامه را در یک فایل با پسوند appx قرار داده و آن را به سمت فروشگاه آپلود می‌کند. هر کاربری که این فایل appx را نصب کند، همه‌ی اسمبلی‌هایی را که در دایرکتوری مربوطه قرار گرفته است، توسط CLR بار شده و آیکن برنامه هم در صفحه‌ی start ویندوز قرار می‌گیرد و اگر دیگر کاربران همان سیستم هم این فایل appx را نصب کنند، از آنجا که قبلاً روی سیستم موجود هست، تنها آیکن برنامه به صفحه‌ی start اضافه می‌گردد و برای حذف هم تنها آیکن برنامه از روی این صفحه حذف می‌شود؛ مگر اینکه تنها کاربری باشد که این برنامه را نصب کرده‌است که در آن صورت کلاً همه‌ی اسمبلی‌های آن از روی سیستم حذف می‌شود.

در صورتیکه کاربرهای مختلف نسخه‌های مختلفی از همان برنامه را روی سیستم نصب کنند، برای اسمبلی‌ها هر کدام یک دایرکتوری ایجاد شده و به ازای نسخه‌ی نصب شده آن کاربر، یکی از این دایرکتوری‌ها مورد استفاده قرار می‌گیرند. کاربران مختلف می‌توانند روی سیستم به طور همزمان از نسخه‌های مختلف برنامه استفاده کنند.

روش‌های پکیج گذاری

برای برنامه‌های دسکتاپ که ربطی به فروشگاه ندارند و بین ایرانیان طرفدار زیادی دارد، نیازی به استفاده از هیچ روش خاصی نیست و یک کپی معمولی هم کفایت می‌کند. همه‌ی فایل‌های مثل اسمبلی، باید در یک دایرکتوری قرار گرفته و به روش کپی کردن آن را انتقال داد. یا برای بسته بندی از یک فایل batch کمک گرفت و آن را روی سیستم نصب کرد و نیازی به هیچ تغییری در رجیستری نیست. برای حذف برنامه هم، حذف معمولی دایرکتوری مربوطه کفایت می‌کند.

البته گزینه‌های دیگری هم برای پکیج کردن این نوع برنامه‌ها وجود دارند:

یکی از روش‌های پکیج کردن فایل‌ها به صورت cab هست که عموماً برای سناریوهای اینترنتی و فشرده سازی و کاهش زمان دانلود به کار می‌رود.

روش دوم استفاده از پکیج MSI است که توسط سرویس نصب مایکروسافت *Microsoft Installer Service* یا **MSIExec.exe** انجام می‌گیرد. فایل‌های MSI به اسمبلی‌ها اجازه می‌دهند که بر اساس زمان تقاضای CLR برای بارگیری اولیه نصب شوند. البته این ویژگی جدیدی نیست و برای فایل‌های exe یا dll مدیریت نشده هم به کار می‌رود.

استفاده از نصاب سازها

بهتر هست که برای انتشار برنامه از برنامه‌های نصاب سازی استفاده کنید که با واسطی جذاب‌تر به نصب پرداخته و امکاناتی از قبیل **shotcut**، حذف و بازیابی و نصب و .. را هم به کاربر می‌دهند.

نصاب سازهای متفاوتی وجود دارند که در زیر به تعدادی از آنها اشاره می‌کنیم:

Install Shield (+) : این برنامه نسخه‌های متفاوتی را با قیمت‌های متفاوتی، عرضه می‌کند و در این زمینه، جزء بهترین‌ها نام برده می‌شود. حتی ویژگی‌های مخصوصی هم برای ویژوال استودیو دارد. شرکت سازنده، برنامه‌ی دیگری را هم اخیر تحت نام **Install Anywhere** عرضه کرده است که اجازه می‌دهد از روی یک برنامه برای پلتفرم‌های مختلف **setup** بسازد.

NSIS : این برنامه هم در زمینه‌ی ساخت **setup** محبوبیت زیادی دارد. این برنامه به صورت متن باز منتشر شده و رایگان است. امکانات این برنامه ساده است و برای راه اندازی سریع یک **setup** و اجرای راحت آن توسط کاربر، کاملاً کاربردی است.

Tarma Installmate : این نرم افزار نسبت به **InstallShield** ساده‌تر و کم حجم‌تر است. حداقل برای برنامه‌های عادی امکانات

مناسبی دارد.

[DeployMaster](#): یک برنامه‌ی دیگر با امکانات حرفه‌ای جهت انشار برنامه‌های دسکتاپ، که از ویندوز 98 تا 8.1 را در حال حاضر پشتیبانی می‌کند.

[QSetup Installation Suite](#): یک برنامه‌ی نصب حرفه‌ای که فایل نهایی آن می‌تواند به دو فرمت exe یا MSI باشد و قابلیت‌هایی چون پشتیبانی از زبان فارسی، ورود لایسنس، سریال نرم افزار و ... را نیز پشتیبانی می‌کند.

[Inno Setup](#): این برنامه هم امکانات خوبی را برای ساخت یک نصاب ساز دارد و همچنین از زبان پاسکال جهت اسکریپت نویسی جهت توسعه امکانات بهره می‌برد.

[Visual Patch](#): وب سایت [پی سی دانلود](#) این برنامه را اینگونه توضیح می‌دهد:

نرم افزار Visual Patch یک ابزار توسعه یافته‌ی نرم افزاری برای ساخت پچ و آپدیت برنامه‌ها می‌باشد. این سازنده پچ باینری، استفاده از فشرده سازی داده DeltaMAX برای سریع‌تر کردن توسعه‌ی نرم افزار، یکپارچگی با نصب نرم افزار و ابزارهای مدیریت پچ از فروشندگانی نظیر Installshield, Lumension, Patchlink, Shavlik, Indigo Rose و ...، را به طور برجسته نمایان ساخته است.

با استفاده از این ابزار پچ کردن برنامه‌ها که برای توسعه دهندگان نرم افزار و برنامه نویسان طراحی شده است، توزیع نرم افزار و سیستم گسترش پچ بهبود می‌یابد. Visual Patch الگوریتم‌های فشرده سازی و state-of- the-art binary differencing را نمایان می‌سازد و این کمک می‌کند که شما به کوچکتر شدن و بهتر شدن پچ‌های نرم افزار اطمینان داشته باشید. و ...

انتشار توسط ویژوال استودیو

ویژوال استودیو هم امکانات خوبی برای انتشار در بخش Properties پروژه، برگه‌ی publish ارائه می‌کند و فایل MSI نتیجه را به سمت وب سرور، FTP Server یا روی دیسک ارسال می‌کند. یکی از خصوصیات خوب این روش این است که می‌تواند پیش نیازهایی مانند فریم ورک دات نت یا sql server Express را به سیستم اضافه کنید؛ در نهایت با مزیت آپدیت و نصب تک کلیک، کاربر، برنامه را بر روی سیستم نصب کند.

اسمبلی‌های انتشار یافته اختصاصی

در روش‌هایی که ذکر کردیم، از آنجا که اسمبلی‌ها در همان شاخه یا دایرکتوری برنامه قرار گرفته‌اند و نمی‌توان آن‌ها را با برنامه‌های دیگر به اشتراک گذاشت (مگر اینکه برنامه دیگری را هم در همان دایرکتوری قرار داد) به این روش *Privately Deployed Assemblies* می‌گویند. این روش برگ برنده بزرگی برای برنامه نویسان، کاربران و مدیران سیستم‌ها محسوب می‌شود. زیرا که جابجایی آنها راحت بوده و CLR در همانجا اسمبلی‌ها را در حافظه بار کرده و اجرا می‌کند. در این نوع برنامه‌ها عملیات نصب/جابجایی/حذف به راحتی صورتی می‌گیرد و نیازی به تنظیمات خارجی مانند رجیستری ندارد. یکی از خصوصیات مهمی که دارد این هست که جداول متادیتا به اسمبلی اشاره می‌کنند که برنامه بر پایه آن ساخته شده و با آن تست شده است؛ نه با اسمبلی موجود دیگر در سیستم که شاید نام نوع مورد استفاده آن یا اسمبلی آن به طور تصادفی با آن یکی است. در مقالات آتی در مورد اشتراک گذاری اسمبلی‌ها بین چند برنامه مفصل‌تر صحبت خواهیم کرد.