



روش های افزایش امنیت و قفل گذاری بر روی فایل های دیتابیس Microsoft Office Access

پیش نمایش مقاله

تیرماه ۱۳۹۵

نگارش ۱,۰۰۰ پیش نمایش

درباره مقاله

منظور از افزایش امنیت و قفل گذاری روی فایل های دیتابیس Access، تنها بهره گیری از امکان استاندارد رمزگذاری روی این فایل ها نیست، چرا که این موضوع کاملاً مشخص است، و در متن راهنماهای مختلف توضیح داده شده است. بکارگیری فایل های Access نه تنها توسط سایر برنامه ها به عنوان اساس دیتابیس بسیار پرکاربرد است بلکه این فایل ها می توانند حاوی کدهای برنامه نویسی باشند که توسط نرم افزار Microsoft Office Access قابلیت اجرا دارند - اینکار توسط محیط VBA اکسس انجام می شود. آنچه در این مقاله بیشتر مورد توجه است استفاده از محیط VBA و تمرکز بر روی این بخش از نرم افزار Access است. در این بخش برنامه نویسی می تواند کدهایی که را که به اداره دیتابیس می پردازد را تنظیم و برنامه نویسی کند، و اطلاعات موجود در دیتابیس را برای رسیدن به هدف یا اهدافی مشخص مورد تکمیل، تحلیل و برای ارائه گزارش از آنها استفاده کند. اما همانطور که هدف مقاله مشخص شد، کدهای نوشته شده در بخش VBA باید حفاظت شود تا علاوه بر جلوگیری از به هم ریختگی برنامه های نوشته شده، حقوق برنامه نویسی نیز مورد حمایت واقع شود، در این مقاله شیوه های افزایش امنیت و قفل گذاری بخش برنامه نویسی بخش VBA دیتابیس مورد توجه واقع شده است. از آنجمله می توان به مباحث زیر اشاره کرد: استفاده از Module ها، مخفی کردن کدهای VBA، تبدیل فایل های accdb به accde، برقراری ارتباط با DLL ها، اجرای خودکار یک فرم هنگام ورود به فایل database. از کار انداختن شیفت، استفاده از نرم افزار Code Protector به منظور حذف معابر هک و کرک. استفاده از پارامترهای اجرای فایل Access، جداسازی اسکریپت ها در قالب فایل های ایستا و دیتابیس در قالب فایل های پویا و برقراری ارتباط بین دو فایل ایستا و پویا. انضمام فایل های ایستا در فایل اجرایی، وابسته کردن اجرای فایلهای اکسس به برنامه های اجرایی.

درباره نویسنده

کلیه حقوق مادی و معنوی این اثر، محفوظ است.

نام: داریوش

نام خانوادگی: طوافی اسمعیلی

پست الکترونیکی: Tavvafi@gmail.com

موقعیت: ایران، تهران.

وبسایت: Tavafi.ir

تلفن های تماس: ۰۹۳۵۱۴۴۹۹۸۳ ۰۹۱۰۵۳۶۳۴۶۳

ساختار این مقاله

در ۵ صفحه مقدمه، مطالب که در بکارگیری بخش های مختلف در ایجاد امنیت اهمیت می یابد - بررسی می شود.

الف - استفاده از Module ها برای استفاده از سرویس های برنامه نویسی در VBA Access

ب - استفاده از رمز گذاری روی پروژه ها و مخفی کردن کدهای VBA

ج - تبدیل فایل های accdb به accde توسط خود نرم افزار access

د - بهره گیری از DLL ها که سه فعالیت در این بند مشهود است:

هـ - اجرای خودکار یک فرم هنگام ورود به فایل database.

و - از کار انداختن شیفت، به منظور جلوگیری از اجرای خودکار از طریق اسکریپت های برنامه نویسی.

ز - استفاده از نرم افزار Code Protector به منظور حذف معابر هک و کرک.

ح - استفاده از پارامترهای اجرای فایل Access به منظور افزایش امنیت .

ط - جداسازی اسکریپت ها و دیتابیس ها در قالب فایل های ایستا و پویا و برقراری ارتباط بین دو فایل ایستا و پویا.

ی - ضمیمه کردن فایل های ایستا در فایل اجرایی.

ک - وابسته کردن اجرای فایل های اکسس به برنامه های اجرایی.

درباره

درباره مقاله

درباره نویسنده

ساختار این مقاله

مقدمه

مقدمه

توضیح درباره Access

توضیح درباره Access Runtime

توضیح درباره VBA

توضیح درباره VBA Compiler

امنیت در Access به روش Exclusive

سر فصل ها

توضیح درباره Module ها

استفاده از Module ها برای برنامه نویسی

رمزگذاری و مخفی کردن کدهای VBA

تبدیل فایل های accdb به accde

تبدیل فایل های mdb به mde

بهره گیری از DLL ها

کد نویسی در اکسس

کد نویسی در یک کامپایلر DLL ساز

ارتباط بین اکسس و DLL ها

اشاره به روش Declare

توضیح روش References

اجرای خودکار فرم هنگام ورود به دیتابیس

Shift و از کار انداختن آن

استفاده از نرم افزار Code Protector

استفاده از پارامترهای اجرای فایل Access

جداسازی اسکریپت ها و دیتابیس ها (ایستا، پویا)

ضمیمه کردن فایل های ایستا در فایل اجرایی

وابسته کردن فایلهای اکسس به برنامه اجرایی

تماس با من

<http://www.Tavafi.ir>

Tavvafi@gmail.com

مقدمه

آنچه در این مقاله بیشتر مورد توجه است استفاده از محیط VBA و تمرکز بر روی این بخش از نرم افزار Access است. توجه داشته باشید که اکسس طراحی شده است که اطلاعات پویا و قابل تغییر را در خود نگهداری کند، اما این مقاله درباره مهندسی ساختار و برنامه نویسی فایل های اکسس است به نحوی که خود فایل اکسس با موتور قدرتمند اکسس بتواند اهداف برنامه نویسی را محقق سازد. در این مقاله شما با مطالبی روبرو خواهید شد که درسدد بهره گیری از امکانات مربوط به عناصر داخلی اکسس و یا خدمات توسعه ایی است که برای بهره گیری از امکانات بیرونی اکسس در خدمت اکسس، تلاش می کند. باید توجه داشته باشید افزایش امنیت نرم افزار در واقع بررسی روش هایی است که موجب افزایش امنیت می شوند، در واقع برای هر روشی که موجب افزایش امنیت می شود روش یا روش هایی ساده یا سخت و یا سخت تر وجود دارد که امنیت بوجود آمده را تحت تاثیر قرار داده و یا آنرا بی اثر خواهد کرد. جدال بین امکاناتی که امنیت بوجود می آورند و ابزارها یا روش هایی که موجب از بین رفتن امنیت می شوند ادامه دارد. به عنوان مثال در بررسی هایی که به منظور جلوگیری از سرقت اتومبیل بعمل آمده است بیش از ۳۰ توصیه وجود دارد این توصیه ها هر کدام به تنهایی بنحوی سعی در جلوگیری از سرقت را در برنامه خود دارند، اما آنچه که نهایتا موجب جلوگیری از سرقت خواهد شد وقت یا زمانی است که سارق باید تلف کند تا یک به یک امکانات ضد سرقت را غیرفعال نماید. یعنی زمان است که جلوی سرقت را می گیرد نه عامل ضد سرقت؛ بنابراین بهتر است بجای بکارگیری تنهای یک روش، از روش های متعدد استفاده کنید تا امکان سرقت - یا هک نرم افزار - کاهش یابد. بکارگیری روش های تلفیقی که موجب ایجاد شبکه های متصل به هم بنحوی که در یک مجموعه تارانبکوتی بتوانند امنیت نرم افزاری را فراهم کنند مطلوب است. ما قصد نداریم از نرم افزارهای کرک و هک، در این مقاله نام ببریم، اما گهگاه به این موضوع اشاره می کنیم که مثلا فلان امکان امنیتی - که روش ایجاد و استفاده از آنها توضیح داده خواهد شد - کافی نیست!

توضیح درباره Access تا آنجا که به این مقاله مربوط می شود

می توان دیتابیس ها را به عنوان مجموعه ای از عناصر (Macro، Report، Form، Query، Table) در نظر گرفت. اگرچه اکسس به این منظور طراحی شده است که اطلاعات پویا را در خود نگهداری کند، ولی برخی از این عناصر تنها قابلیت نگهداری اطلاعاتی ایستا را دارند یعنی تنها میتوان اطلاعاتی را مانند ساختار یک گزارش یا فرم در آنها قرار داد. پس از تولید عناصر بالا در اکسس و بهره گیری از امکانات دقیق، توانمند و ارزشمند آنها، به ساختاری مهندسی شده دست خواهید یافت که بر اساس Relationships تنظیم شده شما، انجام وظیفه خواهند نمود. اگر از دیتابیس (Database) یعنی فایل اکسس می خواهید تنها به عنوان نگهدارنده عناصر اطلاعاتی که از مهمترین آنها Table است استفاده کنید. همین مقدار مهارت و تخصص و آشنایی با اکسس کافیهست. یعنی می توانید تنها با ایجاد یک و یا چند ساختار مهندسی شده Table - و اگر چند Table با ساختار مرتبط با هم ایجاد کرده اید - بکارگیری و تنظیم Relationships می توانید از امکانات فایل تولید شده خود بهره برداری کنید. سطح بعدی تخصص و مهارت بهره گیری از فایل های اکسس تنظیم Query ها است. و سطح بعد از آن اگر با UI یی که طراحی کرده اید متناسب باشد، طراحی Report است. یعنی اینکه برخی از UI ها و دیزاین های از قبیل UI هایی که اغلب تولید شرکت مایکروسافت قابلیت ارتباط با عناصر نامبرده را دارند. آنچه که در این مقاله مورد توجه است بکارگیری Access به نحوی است که مستقل از UI باشد. در این مقاله با UI به عنوان بهره گیری کننده از فایل دیتابیس روبرو نخواهیم شد، چونکه خود اکسس مجهز به موتوری است که هنگام نصب توانایی بهره برداری از امکانات طراحی شده در فایل اکسس را خواهد داشت - پس دیگر نیازی به استفاده از UI نخواهد بود.

توضیح درباره موتور Access Runtime

شما برای طراحی نرم افزار خود به بسته اکسس به طور کامل نیاز دارید، اما برای اجرای نرم افزار طراحی شده خود بر روی رایانه مشتری نیاز به نصب کامل بسته اکسس نخواهید داشت. شرکت مایکروسافت بسته ای را ارائه کرده است که قابلیت اجرای برنامه های اکسس طراحی شده در رایانه مشتری را فراهم می کند. نام این بسته Accessruntime_setup.EXE است. تفاوت اصلی بسته Accessruntime_setup.EXE با بسته Microsoft Office ، در مقدار حجم آن و سرعت نصب آن است.

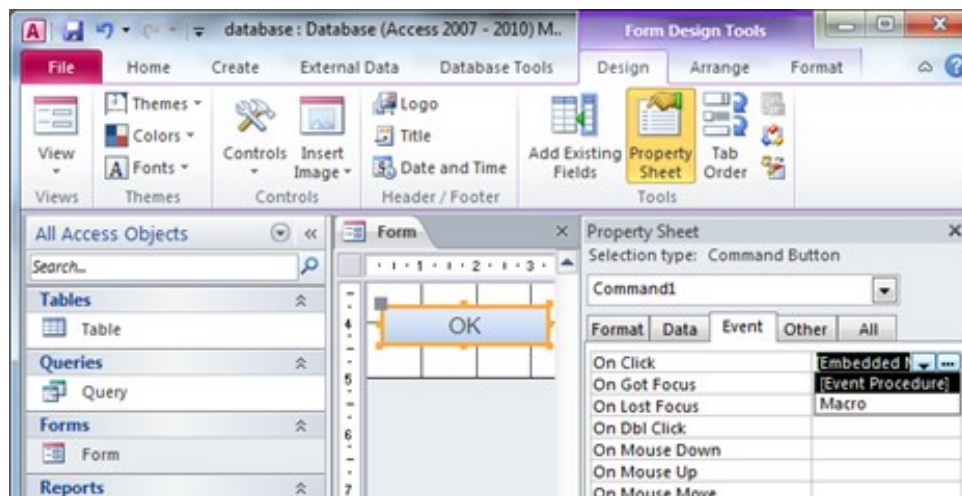
اگر بسته Microsoft Office و صورت کامل و بخصوص همراه با Access روی رایانه خود نصب کنید علاوه بر امکان بهره گیری از فایل های اکسس و اجرای آنها، می توانید نرم افزار خود را طراحی و دیزاین کنید، میتوانید وارد بخش VBA شده و کدهای خود را تغییر دهید و ...

اگر بجای اینکار بر روی ویندوزی که بسته Microsoft Office روی آن نصب نشده ، بسته Accessruntime_setup.EXE را نصب کنید. خواهید توانست از نرم افزاری که طراحی کرده اید با تمام امکانات استفاده کنید. این موتور برای کاربران نرم افزار شما امکان نصب سریعتر نرم افزارتان را فراهم می کند. اما بیشتر کاربران معمولا بسته Microsoft Office را در رایانه خود نصب کرده اند. این وضعیت یک موقعیت استراتژیک محسوب می شود چرا که شما در این وضعیت تنها با ارائه یک فایل ۲ یا ۱۰ مگابایتی نرم افزار خود (فقط در یک دقیقه) و اجرای آن در رایانه مشتری، از موتوری بهره گیری خواهید کرد که نصب آن ۵ تا ۳۰ دقیقه زمان برده است. در چنین وضعیتی کاربران درخواست یافت که نرم افزار سبک شما چه کارایی خارق العاده ای را ارائه نموده است، چرا که نرم افزار شما از موتوری بهره گیری می کند که Access نام دارد و امکانات منحصر به فرد آن همانکون در رایانه مشتری شما قرار دارد.

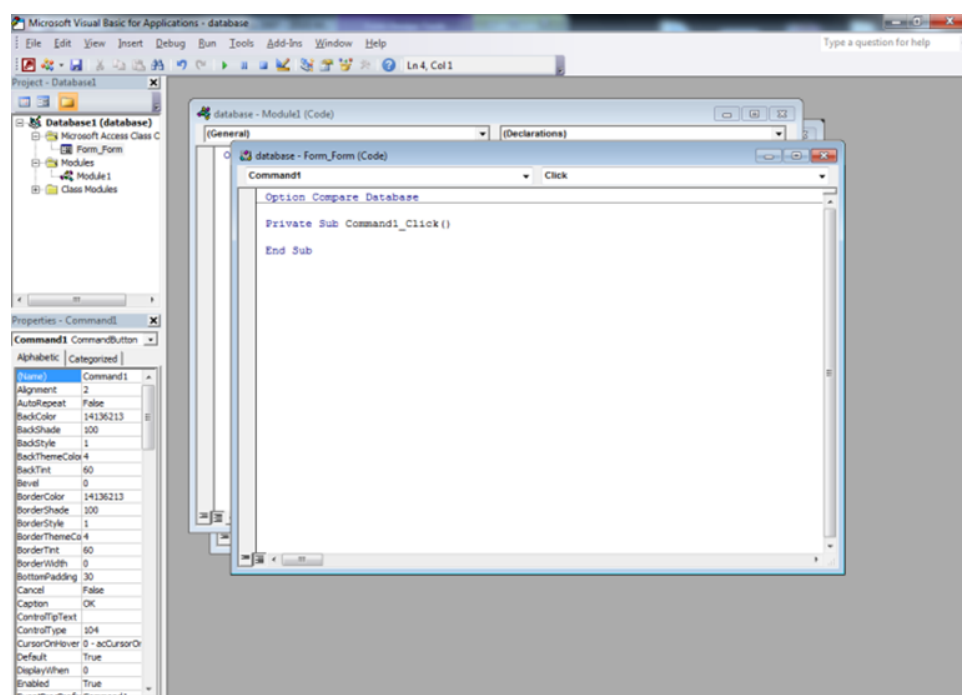
توضیح درباره VBA تا آنجا که به این مقاله مربوط می‌شود

اما برخی از خواص Properties عناصر قابلیت تغییر دارند. این عناصر نیز می‌توانند جزء دیتابیس باشند: Modules و Class ؛ آنچه که در بکارگیری دو عنصر اخیر اهمیت دارد آشنایی با مبحث برنامه‌نویسی و بخصوص برنامه‌نویسی به زبان Visual Basic است.

شروع بهره‌برداری از VBA در یک برنامه Access میتواند از هنگامی باشد که در طراحی یک فرم نیاز به دکمه‌ای داشته باشید و در پنل Property Sheet دکمه انتخاب شده Event مربوط به On click را تنظیم کنید. در این حالت می‌توانید به بخش VBA وارد شوید.



هر چند هر جا که در محیط دیزاین عناصر اکسس که قابلیت برنامه‌ریزی از طریق کدنویسی وجود داشته باشد، شما را به بخش VBA وارد می‌کند - البته بجز مواردی که تصمیم می‌گیرید اجرای دستورات در بخش Macro انجام شود. روش صحیح ورود به بخش VBA زدن همزمان دو کلید Alt + F11 صفحه کلید است.

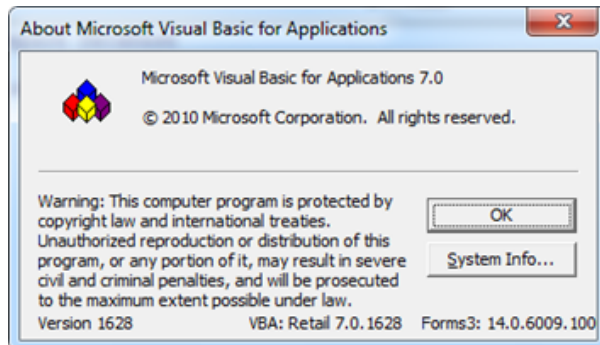


بخش VBA در اکسس یک بخش توسعه دهنده در اکسس بشمار می‌آید که در واقع کامپایلری است که تحت عنوان Microsoft Visual Basic for Application ارائه شده است. این بخش توسعه دهنده هنگام نصب موتور اکسس چه هنگام نصب Microsoft Office و یا هنگام نصب Access Runtime بر روی رایانه نصب خواهد شد.

این بخش دارای دو قسمت است که از آنها به عنوان قسمت دیزاین VBA و قسمت کامپایلر VBA نام خواهیم برد. قسمت دیزاین همان بخشی است که شما در آن کدهای برنامه‌نویسی را تنظیم و بررسی خواهید کرد این قسمت دارای اجزایی شبیه به محیط Visual Basic و یا Visual Studio است و تقریباً از تمام امکانات آن بهره‌برده است، مانند دیباگر و راهنما و اتوکامپلت و نیز List constants و List Properties و List Methods و Object Browser که به طراحی و دیزاین مهندسی شما کمک شایانی خواهد نمود.

توضیح درباره VBA Compiler

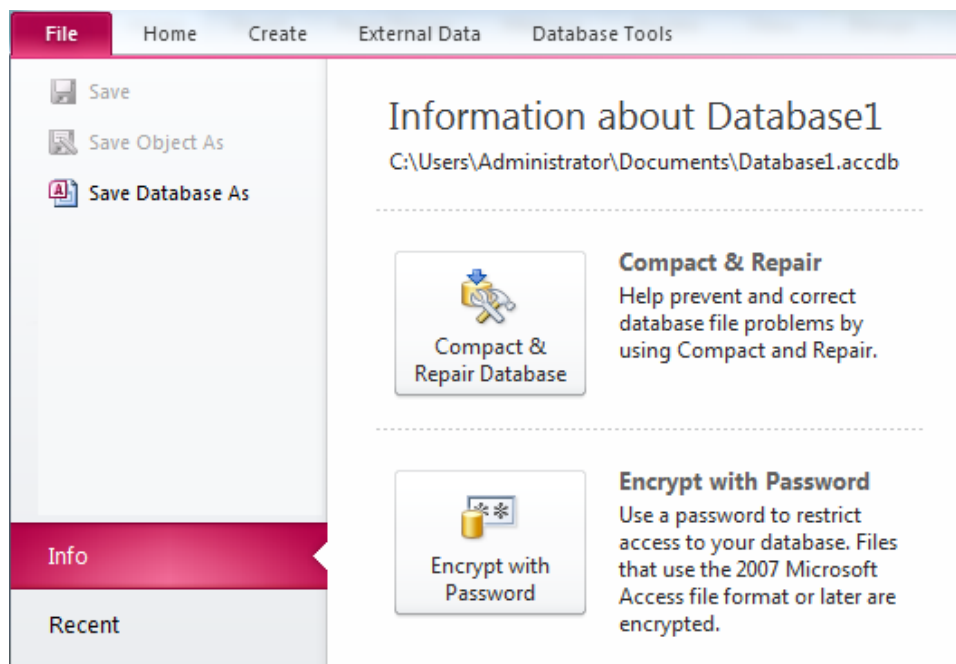
قسمت کامپایلر VBA، قسمتی است که بدون نیاز به قسمت قبلی یعنی دیزاین، کدهای موجود در فایل دیتابیس اکسس را اجرا خواهد کرد. این قسمت ظاهراً هم توانایی اجرای کدهای زبان برنامه‌نویسی Visual Basic و هم توانایی اجرای کدهای کامپایل شده را دارد. در مورد این بحث در بخش (تبدیل فایل های accdb به accde و تبدیل فایل های mdb به mde)، همین مقاله توضیح بیشتری ارائه خواهد شد.



نگارش VBA همراه با افزایش عدد نگارش Microsoft Office تغییر می‌کند، این امر موجب خواهد شد که نرم‌افزاری که در Microsoft Office Access با نگارش ۲۰۱۳ طراحی شده است با بسته‌های نگارش‌های قبلی Microsoft Office Access همخوانی نداشته باشد. اما خوشبختانه این مورد به صورت معکوس معمولاً اتفاق نمی‌افتد یعنی نگارش‌های قبلی طراحی شده در Access در نگارش‌های بعدی مورد کامپایل واقع خواهند شد. به عبارت دیگر به احتمال زیاد، نرم‌افزار دیزاین شده در Access2007 در Access2010 و Access2013 و Access2016 قابل اجرا است.

امنیت در Access به روش Exclusive

امنیت در اکسس دارای سطوح متعددی است که برخی از آنها در راهنماها توضیح داده شده‌اند، از آنجمله می‌توان به مواردی اشاره کرد که در منوی File تحت عنوان Encrypt with password مطرح است.



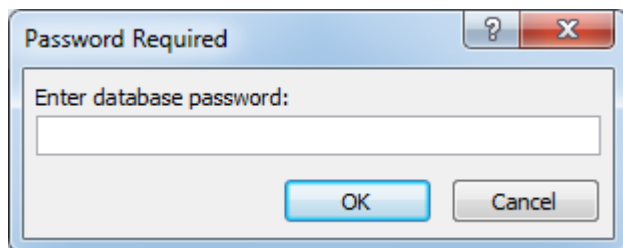
این سطح امنیت به منظور دسترسی به فایل دیتابیس، قابل تنظیم است. این رمز عبور زمانی کاربرد جدی خواهد داشت که قرار باشد دیتابیس به عنوان نگهدارنده اطلاعات پویا (Dynamic) مورد استفاده قرار گیرد. هرچند این مقاله درباره این موضوع زیاد بحث نخواهد کرد ولی از آنجا که مبحث با امنیت مرتبط است به آن اشاره می‌کنیم.

برای تعیین رمز فایل دیتابیس باید به نکاتی توجه کنیم:

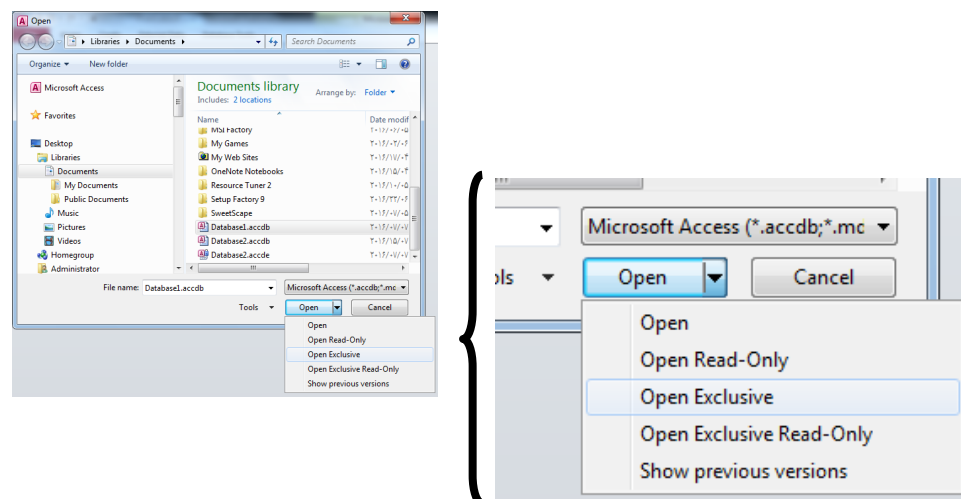
۱- در این مقاله - در بخش (ط) به آن خواهیم پرداخت - فایل‌های دیتابیس را به دو دسته ایستا و پویا دسته‌بندی می‌کنیم که این روش رمزگذاری را تنها برای فایل‌های دیتابیس پویا (Dynamic) توصیه می‌کنیم، و بهتر است برای فایل‌های دیتابیس ایستا (Static) از این رمزگذاری استفاده نکنید. درباره دلایل آن در بخش (ط) توضیح خواهیم داد.

پیش‌نمایش

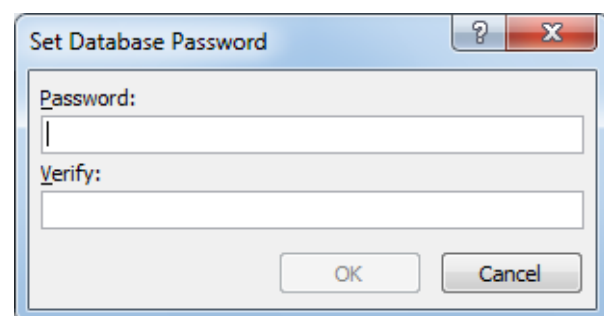
۲- این رمز توسط دیزاینرهای الایی که از دیتابیس دینامیک (پویا) استفاده می‌کنند قابل استفاده است.
۳- این رمز در دیزاینرهایی که از دیتابیس رمزگذاری شده استفاده می‌کنند (منجمله خود دیتابیس استاتیک اکسس)، می‌تواند هنگام باز شدن فایل و بهره‌گیری از امکانات و اطلاعات آن مورد استفاده واقع شود. ولی شما هنگام باز کردن فایل دیتابیزی که روی آن رمزگذاری کرده‌اید باید رمزی را که تعیین کرده‌اید در داخل کادری به شکل زیر مجدداً وارد کنید.



۴- برای رمزگذاری روی فایلی که قصد دارید به این روش بر روی آن رمزگذاری کنید، باید فایل را پس از طراحی ببندید و با گزینه Exclusive آنرا باز کنید. روش بازکردن فایل در شکل زیر نشان داده شده است.



پس از باز کردن فایل دیتابیس، می‌توانید از منوی file گزینه Encrypt with password را انتخاب کنید. سپس رمز دسترسی به فایل را تعیین کنید.



تاکید می‌کنیم:

این روش رمزگذاری برای فایل‌های دیتابیزی که دینامیک هستند کاربرد خواهد داشت، آنچه که در این مقاله به آن خواهیم پرداخت فایل‌های دیتابیس استاتیک (ایستا) هستند، که نیازی به این نوع رمزگذاری ندارند، چرا که اطلاعات موجود در فایل‌های استاتیک پس از کامپایل به کدهای باینری تبدیل می‌شود و در این مقاله بر روی افزایش امنیت اینگونه فایل‌ها توضیحاتی ارائه خواهد شد.

در بخش (ط) توضیح خواهیم داد که چگونه فایل‌های استاتیک از فایل‌های دینامیک استفاده می‌کنند و به این ترتیب فایل‌های دیتابیس استاتیک حاوی کدهای نرم‌افزاری و فایل‌های دینامیک به عنوان دیتابیس حاوی اطلاعات مورد نیاز مصرف می‌شوند.

فایل دیتابیس استاتیک: فایلی است که در آن Module ها و Class ها و Form ها و Report ها قرار دارند، این فایل‌های دیتابیس می‌توانند حاوی Table ها و Query هم باشند، اما بیشتر سعی می‌شود به Table و Query فایل دیتابیس دیگری آنها را متصل (Link) کرد.

فایل دیتابیس دینامیک: فایلی است که در آن هیچ Module یا Class یا Form یا Report ی وجود ندارد و تنها حاوی Table های اصلی و شاید Query ها است، و سعی می‌شود به Table و Query های این دیتابیس متصل (Link) شد و اطلاعات موجود در آنها استفاده کرد.

از این جای مقاله به بعد با مطالبی روبرو خواهیم بود که سعی در ایجاد امنیت در فایل‌های دیتابیس استاتیک خواهد داشت.

توضیح درباره Module ها

ماژول‌ها، قسمت کد نویسی در Access و معروف به بخش VBA هستند که می‌توانید تمام مهارت‌هایی را که از راهنماها و کتاب‌های Visual Basic کسب کرده‌اید، در آن بکار بگیرید.

بهترین کار در دیزاین کدنویس و به عبارتی مهندسی نرم‌افزار در Access طراحی Procedure هایی از قبیل Sub ها و یا Function هایی از نوع Public است که در یک Module یا یک Class واقع شده باشند، و بنحوی با پارامترهای مناسب برنامه ریزی شده باشند، که مقصود اکثر عناصر فراخواننده خود را فراهم آورند، در صورت طراحی و مهندسی درست و هنرمندانه‌ی چنین Procedure هایی، علاوه بر اینکه کدهای تولید شده بسیار کم حجم و در اجرا سریعتر خواهند بود، کنترل آنها برای ایجاد امنیت‌های برنامه ریزی شده از نوعی که در این مقاله به آن خواهیم پرداخت مناسب‌ترند. برنامه‌نویسی و دیزاین و مهندسی نرم‌افزار دارای قلق‌ها و ترفندهای بسیاری است که به دانش برنامه نویس و آگاهی او از امکانات محیط برنامه‌نویسی و خلاقیت او، مربوط خواهد شد. منظور از توضیح درباره Module ها در این مقاله جلب توجه شما خواننده محترم، به این بحث است که در ادامه به منظور تأمین امنیت و حفاظت از کدهای نوشته شده در این بخش از روش‌هایی که Access به این منظور در نظر گرفته است و نیز روش‌های دیگر ابتکاری که در این بخش‌ها (Module ها) مورد بهره‌برداری واقع خواهند شد استفاده خواهیم نمود.

تمامی امکانات سرویس‌های برنامه نویسی در VBA وجود دارد و تقریباً از تمام امکانات آن بهره‌برده است، مانند دیباگر و راهنما و اتوکامپلت و نیز List constants و List Properties و List Methods و Object Browser که به طراحی و دیزاین مهندسی شما کمک شایانی خواهد نمود.

به خاطر داشته باشید:

همانطور که توضیح داده شد کدهای برنامه نویسی باید در فایل دیتابیس متمرکز شوند که به عنوان دیتابیس ایستا در این مقاله مطرح هستند، مثلاً نام این فایل دیتابیس را Program.accdbe می‌گذاریم که در آینده به Program.accdbe تغییر نام خواهد یافت. و فایل دیتابیس که اطلاعات دینامیک را در آن ذخیره خواهیم کرد را Database.accdbe می‌گذاریم.

فایل Database.accdbe فایلی است که تنها حاوی Table ها است و ضمناً دارای امنیت Exclusive است. Encrypt with password in Exclusive method و فایل دیتابیس Program.accdbe فایلی است که حاوی تمامی کدهای برنامه نویسی است. و باید از آن حفاظت نمود. باید دقت کنید فایل دیتابیس استاتیک Program را نباید به سطح امنیت Exclusive برد؛ در عوض از روش‌هایی که در این مقاله برای افزایش سطح امنیت آن کاربرد دارد استفاده خواهیم کرد.

فایل دیتابیس استاتیک Program، دارای کدهای برنامه نویسی در ماژول‌ها است، همچنین دارای فرم‌ها و سایر اجزای مورد نیاز مانند ریپورت‌ها و ... است.

به این نکته مهم در افزایش امنیت کدهای برنامه نویسی اشاره می‌کنیم:

اگرچه می‌توان برای فرم‌ها و ریپورت‌ها و سایر ابزارهای موجود، کدهای برنامه نویسی در نظر گرفت، اما بهتر است به دو دلیل کدها را در داخل ماژول اصلی و در قالب Function ها تعریف کنید و این کدها را از داخل فرم‌ها یا ریپورت‌ها با فراخوانی Function مربوطه اجرا کنید.

دلیل اول: تمرکز برنامه نویسی از طریق فانکشن‌ها و بهره‌گیری از آرگومانها موجب سریعتر شدن و کم حجم تر شدن نرم افزار خواهد شد.

دلیل دوم: امنیت ماژول‌ها (به‌خصوص اینکه اگر تنها از یک ماژول استفاده کنید) بیشتر از امنیت فرم‌ها و سایر ابزارهای کدپذیر است.

در ادامه...

پس از توضیحاتی که در صفحات قبل ارائه شد، مشخص شد که منظور از رمزگذاری بر روی فایلها به صورت تخصصی رمزگذاری روی فایل‌های دیتابیس استاتیک است به این روش‌ها خواهیم پرداخت:

- الف - استفاده از Module ها برای استفاده از سرویس‌های برنامه‌نویسی در VBA ی Access.
- ب - استفاده از رمزگذاری روی پروژه‌ها و و مخفی کردن کدهای VBA.
- ج - تبدیل فایل‌های accdb به accde توسط خود نرم‌افزار Access.
- د - بهره‌گیری از DLL ها. که سه فعالیت در این بند مشهود است.
- هـ - اجرای خودکار یک فرم هنگام ورود به فایل database.
- و - از کار انداختن شیفت، به منظور جلوگیری از اجرای خودکار از طریق اسکریپت‌های برنامه‌نویسی.
- ز - استفاده از نرم‌افزار Code Protector به منظور حذف معابر هک و کرک.
- ح - استفاده از پارامترهای اجرای فایل Access به منظور افزایش امنیت.
- ط - جداسازی اسکریپت‌ها و دیتابیس‌ها در قالب فایل‌های ایستا و پویا و برقراری ارتباط بین دو فایل ایستا و پویا.
- ی - ضمیمه کردن فایل‌های ایستا در فایل اجرایی.
- ک - وابسته کردن اجرای فایل‌های اکسس به برنامه‌های اجرایی.

[illegible][illegible]

1. **Identify the problem.** The first step in the problem-solving process is to identify the problem. This involves understanding the situation, gathering information, and determining the goal.

2. **Analyze the problem.** Once the problem is identified, the next step is to analyze it. This involves breaking the problem down into smaller, more manageable parts and identifying the underlying causes.

3. **Generate solutions.** The third step is to generate potential solutions. This involves brainstorming ideas and considering different approaches to solving the problem.

4. **Evaluate solutions.** The fourth step is to evaluate the potential solutions. This involves comparing the solutions against the goal and the constraints of the problem.

5. **Implement the solution.** The final step is to implement the chosen solution. This involves putting the solution into action and monitoring the results.

6. **Reflect on the process.** The last step is to reflect on the problem-solving process. This involves evaluating the effectiveness of the solution and identifying areas for improvement.

[illegible][illegible]

1. **Identify the problem.** The first step in the problem-solving process is to identify the problem. This involves understanding the situation, gathering information, and determining the goal.

2. **Analyze the problem.** Once the problem is identified, the next step is to analyze it. This involves breaking the problem down into smaller, more manageable parts and identifying the underlying causes.

3. **Generate solutions.** The third step is to generate potential solutions. This involves brainstorming ideas and considering different approaches to solving the problem.

4. **Evaluate solutions.** The fourth step is to evaluate the potential solutions. This involves comparing the solutions against the goal and the constraints of the problem.

5. **Implement the solution.** The final step is to implement the chosen solution. This involves putting the solution into action and monitoring the results.

6. **Reflect on the process.** The last step is to reflect on the problem-solving process. This involves evaluating the effectiveness of the solution and identifying areas for improvement.

[illegible][illegible][illegible]