

برنامه نویسی VBA در اکسل

نویسنده: فرشید میدانی

WWW.FARSARAN.COM

بخش اول، اصول برنامه نویسی VBA

با مطالعه این کتاب سلولهای اکسل
دیگر برای شما مستطیل نخواهد بود.



بخش اول، ویرایش یکم

قیمت بخش اول: ۱۴ هزار تومان

کلیه حقوق مادی و معنوی این کتاب متعلق به موسسه فرساران تفکر است.

پیشاپیش از رفتار شرافتمندانه رعایت حقوق مادی و معنوی این کتاب و حمایتی که از آن می‌نمایید، سپاسگذاریم.

کلیه حقوق مادی و معنوی این کتاب متعلق به موسسه فرساران است

اگر کتاب به هر شکلی بجز خرید مستقیم از وب سایت فرساران به دست شما رسیده است، لطفاً با مراجعه به وب سایت فرساران به آدرس زیر، وجه کتاب را پرداخت نمایید:

www.farsaran.com

مشخصات کتاب

عنوان: برنامه نویسی VBA در اکسل (بخش اول)

نویسنده: فرشید میدانی

ویرایش: یکم (اسفند ۱۳۹۵)

قیمت بخش اول: ۱۴ هزار تومان

ناشر نسخه الکترونیک: موسسه فرساران تفکر

اجازه استفاده از این کتاب فقط متعلق به خریدار می‌باشد و خریدار می‌تواند کتاب را برای استفاده شخصی خود در یک نسخه پیرینت نماید. اگر دوستان، همکاران و یا هر فرد دیگری خواستار مطالعه این کتاب است، لطفاً از آنها بخواهید که کتاب را از طریق وب سایت فرساران (بخش محصولات) خریداری نمایند.

توجه داشته باشید که فروش، دانلود، چاپ، پخش، به اشتراک گذاری و همچنین هر گونه استفاده دیگری از این کتاب و یا بخشی از آن به هر شکل و یا هر روشی غیر مجاز می‌باشد.

همچنین این کتاب به مدت یکسال دارای گارانتی عودت وجه پرداختی می‌باشد.

درباره بخش اول کتاب:

در بخش اول این کتاب که در حال حاضر پیش روی شما است، اصول برنامه نویسی VBA در اکسل را خواهید آموخت. این اصول کلی هستند و چه با اکسل و یا حتی اکسس برنامه نویسی کنید، باید بر این اصول مسلط باشید.

قصد دارم در **بخش دوم کتاب** به صورت ویژه برنامه نویسی در اکسل را آموزش دهم. اما هنوز بخش دوم آماده نیست.

برنامه من آن است که در نیمه اول سال ۹۶ نگارش بخش دوم را آماده سازم و به شما ارائه کنم. اما واقعا نگارش کتاب کاری سخت است و اصلا مطمئن نیستم که اتمام آن بخش چه موقعی خواهد بود.

به همین دلیل از ابتدا تصمیم داشتم که این کتاب را مرحله به مرحله کامل نمایم و آنچه انجام شده است و را در هر مرحله از طریق وب سایت فرساران ارائه نمایم.

گمان می‌کنم که بخش اول این کتاب و بدون بخش دوم می‌تواند مفید واقع شود و شروع خوبی برای یادگیری برنامه نویسی اکسل باشد.

بنابراین منتظر بخش دوم کتاب باشید و همچنین مرا از نقدها و نظرات خودتان آگاه نمایید.

فرشید میدانی

f.meidani@farsaran.com

فهرست مطالب

مقدمه	۹
تقدیم این کتاب	۹
قبل از شروع	۹
تصمیم گرفتی که برنامه نویسی بشی	۱۰
نگرانی من	۱۱
پیش فرض های من درباره شما	۱۲
قوائد نگارشی کتاب	۱۲
مثال های کتاب	۱۳
حمایت از این کتاب	۱۳
فصل صفرم - شروع داستان VBA	۱۴
معرفی زبان VBA	۱۴
زبان VBA به چه دردی می خورد؟	۱۵
چند مزیت و ایراد VBA	۱۵
فصل دوم - آشنایی با دنیای VBA	۱۷
معرفی واژه ماکرو	۱۷
تنظیمات امنیتی ماکروها	۱۷
فعال کردن یک ماکرو	۱۹
آشنایی با VISUAL BASIC EDITOR	۲۱
وارد شدن به محیط VBE	۲۱
اجزای محیط VBE	۲۴
تنظیمات محیط VBE	۲۶
معرفی VBAPROJECT	۲۷
تنظیمات VBAPROJECT	۲۸
بررسی قفل گذاشتن روی VBPROJECT	۲۹
معرفی قسمت IMMEDIATE	۲۹
ساختن MODULE یک قدم کوچک و مهم	۳۰
پروسیجر چیست ؟	۳۲
ریشه شناسی واژه PROCEDURE	۳۳
قوانین نامگذاری پروسیجرها	۳۵
نوشتن اولین برنامه - HELLO WORLD	۳۶
اجرای یک پروسیجر	۳۷
روش اول - استفاده از کلید F5	۳۷
روش دوم - دکمه RUN	۳۸
روش سوم - منوی RUN	۳۸
روش چهارم - از داخل EXCEL	۳۸
روش چهارم و نیم - اجرا با کیبورد	۳۹
روش پنجم - با کلیک بر روی یک چیز	۴۰

۴۰.....	روش ششم - اجرا از داخل یک پروسیجر دیگر.....
۴۱.....	ذخیره فایل.....
۴۳.....	باز کردن یک فایل دارای ماکرو.....

فصل سوم - دستورات پایه زبان VBA..... ۴۴

۴۴.....	قوانین و قوانین کلی دستورات.....
۴۴.....	منظور ما از دستور چیست.....
۴۵.....	حروف بزرگ و حرف کوچک در دستورات.....
۴۶.....	رنگ دستورات.....
۴۶.....	گذاشتن SPACE بین دستورات.....
۴۷.....	استفاده از TAB برای تو رفتگی دستورات.....
۴۸.....	استفاده از CTRL+SPACE.....
۴۸.....	توضیحات یا COMMENT.....
۵۰.....	یک سطرری کردن.....
۵۰.....	چند سطرری کردن.....
۵۱.....	معرفی دستور DEBUG.PRINT.....
۵۳.....	عملگرها و تقدم عملیات ها.....
۵۴.....	متغیر (VARIABLE) چیست.....
۵۶.....	نامگذاری متغیر ها.....
۵۷.....	آشنایی با DATA TYPE.....
۵۹.....	تعریف یک متغیر.....
۶۰.....	مزایای DECLARE (تعریف) کردن یک متغیر.....
۶۱.....	اجباری کردن تعریف متغیرها.....
۶۲.....	تخصیص یک مقدار به یک متغیر.....
۶۲.....	میدان دید متغیرها.....
۶۳.....	میدان دید محدود به یک پروسیجر.....
۶۴.....	میدان دید محدود به ماژول.....
۶۵.....	میدان دید در همه ماژول ها.....
۶۵.....	تعریف مقادیر ثابت (CONSTANTS).....
۶۶.....	استفاده از مقادیر ثابت پیش فرض.....
۶۷.....	تاریخ میلادی.....
۶۷.....	استفاده از توابع VBA.....
۶۸.....	معرفی تابع MsgBox.....
۷۱.....	معرفی تابع InputBox.....

فصل چهارم - دستورات شرطی و کنترل برنامه..... ۷۲

۷۲.....	معرفی دستور GOTO.....
۷۳.....	با کلید F8 آشنا شوید.....
۷۴.....	آشنایی با IF.....
۷۵.....	مثال ۱ دستور IF - قبول شدید.....
۷۵.....	مثال ۲ دستور IF - ازدواج کردی.....
۷۶.....	مثال ۳ دستور IF - عدد زوج یا فرد.....
۷۷.....	مثال ۴ دستور IF - بخش پذیری.....
۷۷.....	آشنایی با عملگرهای منطقی.....

۷۷.....	آشنایی با AND
۷۸.....	مثال ۵ دستور IF – مثلث متساوی الاضلاع
۷۹.....	آشنایی با OR
۷۹.....	مثال ۵ دستور IF – مثلث متساوی الساقین
۸۰.....	مثال ۶ دستور IF – هوای مناسب
۸۰.....	آشنایی با NOT
۸۰.....	مثال ۷ دستور IF – خاموش / روشن کردن
۸۱.....	سایر عملگرها
۸۲.....	حروف کوچک و بزرگ در IF
۸۳.....	دستور IF بلوکی
۸۴.....	مثال ۸ دستور IF – رسم مستطیل یا مثلث
۸۴.....	استفاده از ELSEIF
۸۵.....	مثال ۹ دستور IF – مقایسه دو عدد
۸۶.....	مثال ۱۰ دستور IF – جایزه سیب
۸۷.....	IF های تو در تو
۸۷.....	آشنایی با BOOLEAN
۸۸.....	تابع IIF
۸۹.....	دستور SELECT CASE
۸۹.....	مثال ۲ دستور SELECT CASE – فصل سال
۹۲.....	مثال ۲ دستور SELECT CASE – جایزه سیب
۹۳.....	انواع شرطهای SELECT CASE
۹۵.....	فصل پنجم – حلقه‌ها
۹۵.....	ایجاد تکرار با GoTo - روشی اشتباه
۹۷.....	تفسیر دستور $X = X + 1$
۱۰۰.....	مشاهده مقدار متغیر با TOOL TIP
۱۰۰.....	مشاهده مقدار متغیر با WATCH
۱۰۱.....	چرا استفاده از GOTO اشتباه است
۱۰۳.....	دستور FOR – NEXT
۱۰۴.....	نقش متغیر در دستور FOR – NEXT
۱۰۵.....	مثال ۱ دستور FOR NEXT – جذر
۱۰۵.....	مثال ۲ دستور FOR NEXT – اعداد زوج
۱۰۶.....	استفاده از STEP در FOR – NEXT
۱۰۷.....	مثال ۳ دستور FOR NEXT – خروج از حلقه
۱۰۸.....	مثال ۴ دستور FOR NEXT – جمع اعداد
۱۱۰.....	دستور DO – LOOP
۱۱۰.....	مثال ۱ دستور DO – LOOP – پیغام سلام
۱۱۱.....	معرفی عبارت WHILE در دستور DO – LOOP
۱۱۱.....	مثال ۲ دستور DO – LOOP – برنده شدید
۱۱۲.....	مثال ۳ دستور DO – LOOP – اعداد دو رقمی
۱۱۲.....	شرط در ابتدا یا انتهای بلوک DO – LOOP
۱۱۳.....	مثال ۴ دستور DO – LOOP – خواندن یک فایل
۱۱۵.....	مثال ۵ دستور DO – LOOP – فاکتوریل
۱۱۷.....	دستور EXIT DO
۱۱۷.....	مثال ۶ دستور DO – LOOP – ک م م
۱۲۰.....	آشنایی با عبارت UNTIL در دستور DO – LOOP

۱۲۱.....	جمع بندی دستور DO - LOOP
۱۲۲.....	حلقه‌های تو در تو
۱۲۲.....	مثال ۱ حلقه‌های تو در تو - مستطیل
۱۲۳.....	مثال ۲ حلقه‌های تو در تو - جدول ضرب
۱۲۵.....	مثال ۳ حلقه‌های تو در تو - اعداد اول
۱۲۶.....	مثال ۴ حلقه‌های تو در تو - چاپ آرام زیگزاگ

فصل ششم - خطاها و مدیریت آنها..... ۱۲۹

۱۳۰.....	خطاهای نوشتاری - SYNTAX ERROR
۱۳۲.....	خطاهای در هنگام کار برنامه - RUN TIME ERROR
۱۳۲.....	مثال ۱ ERROR HANDLING - تقسیم بر صفر
۱۳۴.....	چند توصیه برای پیشگیری از خطاها
۱۳۴.....	مدیریت خطاها با دستور ON ERROR GOTO
۱۳۵.....	مثال ۲ ERROR HANDLING - دستور ON ERROR GOTO
۱۳۷.....	دستور ON ERROR GOTO 0
۱۳۷.....	بی خیال خطاها - ON ERROR RESUME NEXT
۱۳۸.....	مثال ۳ ERROR HANDLING - حذف شیت اکسل
۱۳۹.....	مروری بر ابزارهای DEBUGGING در VBA
۱۳۹.....	دستور STOP
۱۴۰.....	استفاده از BREAK POINT

فصل هفتم - پروسیجرها و توابع..... ۱۴۱

۱۴۱.....	پروسیجرها و برنامه نویسی ساخت یافته
۱۴۲.....	یادآوری مطالبی قبلی
۱۴۲.....	میدان دید پروسیجرها
۱۴۲.....	تعریف پروسیجر از نوع PUBLIC
۱۴۳.....	تعریف پروسیجر از نوع PRIVATE
۱۴۳.....	دادن ورودی به پروسیجر
۱۴۴.....	کاربرد پروسیجرها با ورودی
۱۴۵.....	با توابع آشنا شوید
۱۴۶.....	مقایسه توابع و پروسیجرها
۱۴۶.....	کلیات ساخت و استفاده از یک تابع
۱۴۷.....	CALL کردن توابع
۱۴۷.....	مثال ۱ توابع - تابعی بدون ورودی و بدون خروجی
۱۴۹.....	میدان دید توابع
۱۴۹.....	تعریف خروجی یک تابع
۱۵۰.....	مثال ۲ توابع - اسم من
۱۵۱.....	مثال ۳ توابع - فقط عدد طبیعی
۱۵۱.....	ورودی توابع
۱۵۱.....	مثال ۴ توابع - مکعب
۱۵۲.....	مثال ۵ توابع - طول وتر مثلث
۱۵۳.....	ورودی اختیاری توابع
۱۵۴.....	مثال ۶ توابع - فصل های سال

۱۵۵.....	تابع با تعداد ورودی زیاد.....
۱۵۵.....	مثال ۶ توابع - ادغام متن سلولها.....
۱۵۷.....	مثالهای کاربردی از توابع.....
۱۵۷.....	مثال ۷ - تابعی برای تجزیه یک متن به حروف.....
۱۵۸.....	مثال ۸ - تابعی برای استخراج یک عدد از متن.....
۱۵۹.....	مثال ۹- حذف حروف صدا دار.....
۱۵۹.....	معرفی عملگر LIKE.....
۱۶۱.....	ادامه مثال ۹ - حذف حروف صدا دار.....

فصل هشتم - ARRAY..... ۱۶۳

۱۶۴.....	تعریف متغیری از نوع آرایه.....
۱۶۵.....	کار با اعضای یک آرایه، تکنیک FOR - NEXT.....
۱۶۷.....	کار با اعضای یک آرایه، تکنیک FOR - EACH.....
۱۶۷.....	مثال ۱ آرایهها - بزرگترین عدد.....
۱۶۸.....	آرایههای داینامیک- متغیری از نوع VARIANT.....
۱۶۹.....	مثال ۲ آرایهها - تجزیه یک متن.....
۱۷۰.....	مثال ۳ آرایهها - استخراج کد ملی از شرح.....
۱۷۱.....	آرایههای داینامیک.....
۱۷۲.....	آرایههای دو بعدی (چند بعدی).....
۱۷۴.....	آرایهها و سلولهای اکسل.....
۱۷۴.....	مثال ۴ آرایهها- خواندن و نوشتن در سلولها.....
۱۷۶.....	آرایهها به عنوان ورودی یک تابع.....
۱۷۷.....	ورودی تابع از نوع VARIANT.....
۱۷۸.....	مثال ۴ آرایهها - مرتب سازی.....

مقدمه

قبل از هر چیز از شما برای خریداری این کتاب تشکر می‌کنم و امیدوارم که آموزش‌های آن بتواند سریع و دقیق شما را به مقصودتان برساند. نگران نباشید اگر شما با دنیای برنامه نویسی آشنا نیستید. در این کتاب سعی می‌کنیم که این دنیای جالب را به شما با حوصله نشان دهیم و اگر از واژه‌ای فنی استفاده می‌کنم تلاش خواهیم کرد که آنرا با مثال‌های روزمره برای شما شفاف کنم.

تقدیم این کتاب

سال‌ها پیش وقتی هنگامی که من در دبیرستان درس می‌خواندم، دوست بسیار عزیزی و گرامی داشتم به نام کریم رعنا حسینی. فردی دوست داشتنی و کتاب‌خوان.

روزی با کریم به شوخی و جدی قرار گذاشتیم که هر کسی کتابی نوشت، به دیگری تقدیم کند. اگر چه کریم سال‌هاست که دیگر در این دنیا نیست، اما همواره من به یادش هستم و به قول شاعر "من از یادت نمی‌کاهم".

حال این کتاب را که اولین کتاب مهم و جدی است که نوشته‌ام را به او تقدیم می‌کنم.

قبل از شروع

کتاب‌های فارسی برنامه نویسی VBA و برنامه نویسی اکسل انگشت شمار هستند و با هدف نگارش کتابی قابل فهم برای کاربران اکسل اقدام به نوشتن این کتاب کردم. برخی از ویژگی‌های این کتاب به شرح زیر است:

- سالها تجربه کار و تدریس نویسنده (یعنی من) با اکسل در آن منعکس شده است
- لازم نیست که شما هیچ پیش زمینه‌ی برنامه نویسی داشته باشید
- کتاب به سبکی آموزشی ما ایرانی‌ها نگاشته شده است
- از چندین منبع انگلیسی برای نوشتن آن تا حد ممکن تقلب کرده‌ام
- غلط‌گیری و به روزرسانی سریع و ساده آن
- مطالب با حوصله و جامع گفته شده است
- مطالب را با علاقه و انگیزه زیادی تدریس کرده‌ام

من تجربه سالها برنامه نویسی و انجام پروژه‌های متفاوتی را در اکسل دارم و بارها دوره‌های برنامه نویسی حضوری را در شرکت‌ها تدریس کرده‌ام و تجربه تدریس نزدیک به ۲۰ سال تدریس از زمان DOS تا کنون را سعی کرده‌ام در این کتاب پیاده کنم.

کتاب‌های انگلیسی بی‌نظیر هستند و من قطعا از بهترین‌ها استفاده کرده‌ام تا مطالب را انسجام بیشتری ببخشم و کتابی کامل‌تری را برای شما مهیا سازم.

در ضمن اینکه این کتاب به صورت PDF است و قاعدتا اگر استقبال و حمایت شود، می‌توانم آنرا به روزرسانی و تکمیل‌تر کنم پس لطفاً از آن با هر شکلی که ممکن است حمایت کنید.

اما در خصوص اینکه مطالب کتاب جامع، کامل و باحوصله است باید بگویم که خود من از روی اینترنت کتابهای فارسی زیادی را دانلود کرده‌ام و البته برخی از آنها مانند این کتاب رایگان هم

نبودند و مشکلی که همه آنها داشتند یک چیز بود! در اوایل کتاب نویسندگان با حوصله مطالب را درس می‌دهد اما هر چه کتاب جلوتر می‌رود و مطالب سخت‌تر و پیچیده‌تر می‌شوند، نویسندگان بی حوصله می‌شود و خیلی اجمالی مطالب را می‌گویند. من در این کتاب دقیقا سعی کردم که همواره توضیحاتم را کامل بدهم حتی اگر موفق نشوم که کتاب را هیچوقت تکمیل کنم.

شاید افرادی بگویند که این روش نوشتن کتاب که همه چیز را توضیح بدهند خوب نیست و باعث می‌شود که حجم مطالب زیاد شود و شاید خواننده خسته، ناامید و ... بشود.

من با این رویکرد خلاصه گفتن مطالب، در حوزه برنامه نویسی کاملا مخالفم، در واقع شاید در برخی از علوم بتوانیم کلیاتی را مطرح کنیم اما در برنامه نویسی اینکار باعث می‌شود که در ابتدا خواننده از پیشرفت سریع خودش خوشنود شود اما در اولین قدم‌ها، با مسائل پیچیده‌ای مواجه می‌شود که نمی‌تواند آنها را تفسیر کند و کاملا ممکن است برای همیشه برنامه نویسی را رها کند.

در ضمن اینکه اشاره می‌کنم در کتابهای برنامه نویسی پرفروش انگلیسی، نویسندگان هیچ وقت از گفتن همه جانبه مطالب و ریزه‌کاری‌ها طفره نمی‌رود و یا بی حوصله نمی‌شود و تا حد ممکن به آنها می‌پردازد.

تصمیم گرفتی که برنامه نویسی بشی ...

برنامه نویسی از اصولی ساده و پایه ای ساخته می‌شود که یادگیری آن سخت نیست و هر کسی می‌تواند به سادگی در طی چند ماه یا حداکثر یک سال به سطحی برسد که از عهده کارهای خودش برآید. اما روی دیگر سکه آن است که برنامه نویسی اکسل به شما کمک می‌کند تا کارهای و وظایف سازمانی خود را سریع تر انجام دهید، بدون اشتباه و این احتمالا یعنی شما بتوانید به موقع محل کارتان را ترک کنید و به منزل بروید و با خانواده باشید و بنابراین برنامه نویسی می‌تواند به زندگی شخصی شما هم کمک کند.

واقعا اعتقاد دارم که برنامه‌هایی که شما در سازمان های بزرگی مثل ایران مثل تامین اجتماعی، دارایی، بانک‌ها، شهرداری و ... می‌نویسید می‌تواند کارها را سریع و دقیق کند و در نهایت کشوری منظم تر داشته باشیم و مردمی خوشحال تر داشته باشیم.

خیلی از مواقع ممکن است یک فرد در چند ماه یک برنامه کوچک تولید کند و آن برنامه سالها در آن سازمان بکار گرفته شود و کارهای تکراری حذف شود و دقت بالا برود.

شاید بدانید که من برنامه تاریخ هجری در اکسل را از سال ۸۵ بر روی وبلاگم قرار دادم و شاید این برنامه ساده چقدر توانست مفید واقع شود تقریبا همه جا از آن استفاده می‌کنند.

پس به خاطر خودتان یا خانواده تان و یا کشورتان لطفا کمی برنامه نویسی اکسل را یاد بگیرید .

البته برنامه نویسی اکسل مزایای زیر را نیز دارد:

- باعث می‌شود که در سازمان فردی کارآمدتر جلوه کنید
- می‌توانید باعث صرفه جویی در منابع مالی سازمان شود یا حتی در منابع انسانی
- حس شادی ناشی از خلاقیت و نوشتن یک برنامه دست کم نگیرید.

نگرانی من

واقعیت آن است که شما قرار است یک تکنیک و یک قدرت بالایی را در دنیای کامپیوتر کسب کنید. تا الان کاربر عادی / User بوده اید و حالا قرار است که یک کاربر حرفه ای شود و این یعنی باید چیزهای زیادی را بیاموزید.

این چیزها واقعا سخت نیستند اما حجمشان کمی زیاد است و بجز حوصله چاره دیگری ندارید. من مجبورم که مفاهیم زیادی را به شما بیاموزم و از آنجایی که احتمال دارد شما خسته و یا ناامید شوید، مرا نگران می‌کند.

اگر در فصل‌هایی خسته و یا ناامید شدید، خیلی خوب مطالب را نفهمیدید، اصلا نگران نباشید این یک پدیده کاملا عادی است و می‌توانیم پیشنهاد بدهم که با سرعت کمتری مطالب را بخوانید و فصل را مرور کنید و یادداشت بنویسید و یا حتی چند روزی مطالعه را متوقف کنید.

اگر در فهم مطلبی مشکلی داشتید و آنرا دو یا سه بار خواندید و باز هم متوجه نشدید، حق ندارید به هوش و یا استعداد خودتان شک کنید بلکه بهترین شک آن است که من به عنوان نویسنده و معلم شما نتوانسته‌ام آن مطلب را درست آموزش دهم. پس اولاً به من ایمیل بزنید و شکایت کنید و حتما پیگیری کنید تا آنرا درست کنم و دوماً تا آن زمان بهترین کار آن است که چند ویدئوی آموزش از سایت youtube در خصوص آن مطلب را ببینید و یا اینکه یک کتاب انگلیسی مرجع را بخوانید.

نگرانی دیگری که دارم است آن است که این کتاب را فقط مطالعه کنید و مثال‌ها را انجام ندهید. واقعیت آن است که باید مثال‌ها را هر چند که بسیار در نظر شما ساده باشد، مجدداً خودتان تایپ و اجرا کنید. مثال‌ها را تغییر بدهید و نتیجه‌ها را آزمایش کنید تا تصویر مبهم برنامه نویسی در ذهن شما به تدریج شفاف و کامل شود.

تاکید می‌کنم که قرار است شما مهارت سطح بالایی را یاد بگیرید و یادگیری بجز با حوصله، زحمت و پشتکار هیچ راه میانبری ندارد.

پیش فرض های من درباره شما

خوب حداقل باید موارد زیر را داشته باشید که بتوانید از کتاب استفاده کنید:

- قطعا اکسل را در حد متوسط باید بدانید و در غیر اینصورت بهتر است که ابتدا اکسل را یاد بگیرید. (از دوره های اینترنتی فرساران، دوره های حضوری و یا کتاب های ما برای اینکار استفاده کنید)
- کاربر دائمی اکسل هستید و در محیط کار به صورت مرتب با اکسل سر و کار دارید.
- قول می دهید که حداقل هفته ای ۱ ساعت مطالعه داشته باشید
- همت لازم برای تمرین و به اتمام رساندن این کتاب را داشته باشید.
- حوصله چند بار خواندن یک مطلب را دارید.
- مثال ها و تمرین های هر قسمت را حتی اگر خیلی ساده هم باشند، حتما اجرا می کنید.
- لازم نیست که قبلا برنامه نویسی کار کرده باشید.
- نسخه اکسل شما مهم نیست.

قوائد نگارشی کتاب

من از فونت Consolas که فونت خوانایی است برای نوشتن برنامه ها استفاده میکنم و به تفاوت های عدد یک و صفر با حروف L و O انگلیسی دقت کنید.

```
Wowoo0o0 10+20  
Salam Leila jan
```

ما در برنامه نویسی می توانیم که یک سطر را در چندین سطر بشکانیم و از آنجایی که ممکن است نوشتن یک سطر بلندتر از عرض صفحه کاغذ شود، من از این تکنیک یعنی شکاندن سطر استفاده می کنم. برای آنکه یک سطر را به قطعات کوچکتری تقسیم کنیم باید از یک علامت _ (آندر اسکور یا آندر لاین) که قبلش یک فاصله (اسپیس) گذاشته ایم استفاده نماییم:

```
Selection.PasteSpecial Paste:=xlValues, _  
Operation:=xlNone, SkipBlanks:=False, _  
Transpose:=False
```

قسمت های مهم یک برنامه را که باید به آن توجه کنید، برای شما به صورت Bold و Italic نمایش داده ام: (کلمه Like در مثال زیر و همچنین علامت های ستاره)

```
Debug.Print "Shahr Shiraz ziba" Like "*Shiraz"
```

در هر قسمت از درس برای تاکید بر موضوعی و یا طبقه بندی ذهن شما نکته ها و توجه هایی درج شده است.

مثال های کتاب

مثال های حل شده هر فصل به صورت جداگانه ای برای شما آماده شده است که با همین کتاب آن فایل ها را دریافت کرده اید.

هر مثال یک نام دارد که در ابتدای نام مثال، نام آن فصل از کتاب که مثال در آن زده شده است، آمده است. مثلاً تمامی مثال های فصل پنجم کتاب با Chapter5 شروع می شوند و نام اولین مثال فصل پنجم Chapter5_Ex1 است و دومین مثال فصل پنجم Chapter5_Ex2 خواهد بود. در برخی از موارد نام مثال را یک واژه قابل فهم گذاشته ایم.

توجه: فایل های مثال از چندین ماژول تشکیل شده است.

حمایت از این کتاب

اگر از این کتاب خوشتان آمد و آنرا مفید یافتید لطفاً آنرا به هر شکلی به دیگران معرفی کنید مثلاً لینک معرفی آنرا در وبلاگتان قرار دهید و یا یک ایمیل به همکاران سازمانتان بزنید.

توجه داشته باشید که نسخه اصل آن فقط متعلق به شماست و آنرا با دیگران به اشتراک نگذارید و اگر فردی از شما کتاب را خواست از او بخواهید که با خرید از سایت فرساران این کتاب را تهیه کند. در ضمن اگر مایل هستید می توانید فقط برای استفاده شخصی خودتان این کتاب را پیرینت نمایید.

معرفی این کتاب در **LinkedIn** نیز مزید امتنان خواهد بود. ☺



فصل صفرم - شروع داستان VBA

در این فصل می خواهیم اولین قدم ها را برای آشنایی با VBA انجام دهیم و قبل از هر چیز مختصری بگوییم که VBA چیست؟

معرفی زبان VBA

شرکت مایکروسافت سال ها یک زبان برنامه نویسی استاندارد به نام Visual Basic را تولید می کرده و به فروش می رسانده است . از روی زبان برنامه نویسی Visual Basic یک نسخه خاص به نام VBA نیز ساخت . بنابراین VBA فرزند زبان برنامه نویسی Visual Basic است.

VBA یک زبان برنامه نویسی است.

حالا که متوجه شدید VBA فرزند Visual Basic است طبیعی است که بگوییم تمام اصول و قواعد زبان مادری را به ارث برده است بنابراین هر کسی که اصول VBA را یاد بگیرد قطعاً اصول زبان مادری را یاد گرفته است و همچنین بالعکس.

VBA مخفف واژه Visual Basic For Application است و هدف از این نامگذاری آن است که برنامه های VBA در داخل یک نرم افزار (Application) «میزبان» مانند اکسل، اجرا می شوند و بدون این «میزبان» ما نمی توانیم برنامه های نوشته شده به زبان VBA را اجرا کنیم.

نکته ۱: لازم نیست کسی VBA را نصب یا داندلود کند. VBA در «داخل» سایر نرم افزار به رایگان گذاشته است. مثلاً هنگامی که نرم افزار Excel را نصب می کنید، در داخل آن VBA نیز نصب شده است.)

نکته ۲: من از واژه «در داخل» استفاده کردم و باید به شما بگویم که VBA به صورت مستقلی اجرا نمی شود و اجرای آن مستلزم آن است که شما اکسل را باز کرده باشید و سپس می توانید از داخل اکسل وارد محیط VBA شوید و برنامه نویسی کنید پس به دنبال گزینه VBA در داخل منوی Start ویندوز نگردید.

نکته ۳: شما در PowerPoint اسلاید دارید، در Word صفحه و در Excel شیت . خوب اگر شما با VBA برنامه ای بنویسید که در داخل Word با صفحات کار کند، کاملاً بدیهی است که آن برنامه در داخل Excel اجرا نشود چون اکسل اساساً صفحه ندارد.

بنابراین اگرچه اصول کلی زبان VBA در تمامی برنامه ها یکی است اما برنامه ای که برای شیت های Excel نوشته شده است را نمی توانید در داخل PowerPoint کپی کنید و انتظار داشته باشید که با اسلایدهای پاورپوینت کار کند.

تذکر: بررسی یک زبان برنامه نویسی، بحثی پیچیده و فنی است و من در اینجا با رویکرد ساده گرایانه ای، ویژگی های VBA را برای شما توصیف کرده ام و برخی از موارد گفته شده در این چند سطر از نظر علمی اعتبار چندانی ندارد.

زبان VBA به چه دردی می خورد؟

بدیهی است که در اکسل شما هزاران قابلیت دارید و می توانید بسیاری از کارها را با اکسل و بدون VBA انجام دهید و حالا باید به شما بگوییم که چرا قرار است VBA هم یاد بگیرید.

الف) سپردن انجام کارهای که شما معمولاً انجام می دهید

فرض کنید که یک گزارش هر ماه باید تهیه و به چند نفر ارسال شود، می توانیم خیلی از این فرآیند را برنامه نویسی کنیم و شما درگیر آن کارها نشوید و برنامه برای شما آنها را انجام دهد.

ب) تکرار یک عملیات تکراری

مثلاً قرار است که بر روی ۱۲ شیت اکسل یک کاری را انجام دهید، خوب واقعا سخت نیست که برنامه ای بنویسید که اینکار را انجام دهد. در ضمن اگر موفق شدید که برای ۱۲ شیت اینکار را انجام دهید قطعاً اکسل برای انجام این کار بر روی ۱۲۰ شیت و یا ۱۲۰۰ شیت اصلاً خسته و ناراحت نخواهد شد.

ج) خلق ابزاری جدید

بالاخره ابزارهای اکسل محدود هستند و شما ممکن است یک تابع جدید و یا یک ابزار برای یک کار خاصی را لازم داشته باشید که اکسل آنرا ندارد.

د) کنترل سایر برنامه ها

با کد نویسی در داخل اکسل می توانید سایر برنامه ها را نیز کنترل کنید مثلاً از اکسل به Outlook فرمان دهید که یک ایمیل برای کاوه ارسال کند و در داخل آن ایمیل مقدار آخرین موجودی انبارها را بنویس.

چند مزیت و ایراد VBA

الف) واقعا برای یادگیری VBA ساده است .

البته شما حرف من را شاید خیلی قبول نکنید که ساده است اما هنگامی که با مفاهیمی عجیبی در سایر زبان های برنامه نویسی مواجه می شود آنوقت است که می فهمیم VBA چقدر ساده است.

ب) استفاده از قابلیت های اکسل

خیلی از وقت ها ما کارها را برنامه نویسی نمی کنیم مثلاً اگر بخواهیم یک لیست را Sort کنیم لازم نیست که برویم و الگوریتم های مرتب سازی استفاده کنیم . بهتر است سری به اینترنت بزنید تا متوجه شوید که چندین روش مختلف برای مرتب کردن یک لیست داریم .

فقط کافی است که از ابزار Sort اکسل استفاده کنیم و خود اکسل کارها را انجام می دهد.

ج) نیاز به فایل اکسل برای اجرا (عیب)

همواره یک برنامه VBA نیاز دارد تا اکسل بر روی سیستم نصب باشد تا اجرا شود و نمی توان فایل های اجرایی مانند exe ساخت.

د) ساده است و گاهی محدود

همانطور که گفتیم VBA واقعا ساده است و خوب این ساده بودن به دلیل آن است که قرار نبوده کارهای عجیب و غریب با VBA انجام دهیم و در نتیجه باید بدانید که انجام کارهای «خاص» در VBA یا خیلی سخت و یا غیر ممکن است. پس بهتر است از VBA انتظارات درستی داشته باشیم. مثلا انتظار اینکه شما بتوانید اطلاعات یک دستگاه بارکد خوان را با VBA بگیریم می تواند یک مثال خوب از کارهای سخت است. (نگفتم که نمی شود!)

ه) غیر قابل بودن امکانات فروش و یا توزیع برنامه

خیلی این سوال می شود که چطور می توانم بر روی برنامه اکسلی که نوشته ام پسورد بگذارم و آنرا قفل کنم و یا محدودیت زمانی به آن بدهم!

اساسا اکسل و برنامه نویسی آن برای تولید نرم افزارهای اجرایی و سپس فروش و ... ساخته نشده است و طراح این سوالات در حوزه اکسل اشتباه است.

اگر در نظر دارید که برنامه ای با اکسل بنویسید و سپس روی آن قفل بگذارید و بعدش به فروش برسانید، همین الان بروید به سراغ سایر زبان های برنامه نویسی حرفه ای که برای اینکار ساخته شده اند و این کتاب را ببندید!

فصل دوم - آشنایی با دنیای VBA

حالا کم کم باید وارد کارهای عملی تری شویم و با دنیای VBA آشنا شویم.

معرفی واژه ماکرو

هدف اصلی ما در این کتاب نوشتن یک «برنامه» است و البته واژه های دیگری هم برای توصیف اینکار یعنی «برنامه» به کار می رود که معادل همان واژه برنامه یا برنامه نویسی است. یکی از آن واژه ها ماکرو / Macro است. پس اگر جایی من گفتم یک ماکرو داریم یعنی یک برنامه داریم.

البته معادل واژه «برنامه»، واژه های دیگری را در کتاب ها، مقالات و ... مشاهده می کنید که برخی از مشهورترین آنها را برای شما در زیر لیست کرده ام:

- برنامه = ماکرو = پروسیجر = کد = سابروتین
- برنامه نویسی = ماکرو نویسی = نوشتن یک پروسیجر = کد نویسی = ایجاد یک سابروتین

تنظیمات امنیتی ماکروها

مجبورم این تنظیم را در همین ابتدای کتاب بگویم.

فرض کنید که یک نفری یک فایل اکسل رو به شما داده است و شما هم از همه جا بی خبرید که آن فایل ماکرو (برنامه) دارد و فایل رو باز می کنید و به یکباره می بینید که کامپیوتر شما خاموش شد!!!

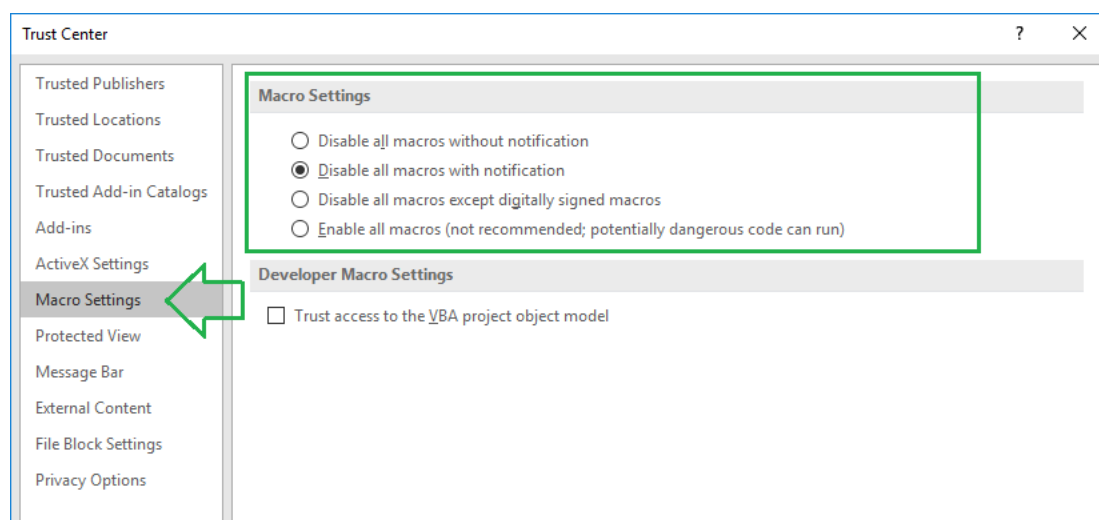
خیلی اتفاق عجیبی نیست وقتی که یک فایل برنامه/ماکرو دارد، قطعاً می تواند کارهای مخربی را روی دستگاه شما انجام دهد.

به همین دلیل اکسل و سایر برنامه های آفیس به صورت پیش فرض هر فایلی را که ماکرو/برنامه داشته باشه رو قرنطینه یا Block می کند و برنامه های آن نتوانند اجرا شوند. به همین دلیل شما این فرصت را خواهید داشت که متن برنامه را مطالعه کنید و سپس تصمیم بگیرید که آن برنامه بتواند اجرا شود و یا خیر.

این تنظیمات یعنی آیا قرنطینه شدن انجام بشود یا نه، در تنظیمات اکسل (Options)، در جایی به نام Macro Settings وجود دارد که از مسیر زیر قابل دستیابی است:

File → Options → Trust Center → Trust Center Settings → Macro Settings

تصویر تنظیمات امنیتی ماکروها



نکته: این تنظیمات هنگامی که یک فایل را باز می کنید تاثیر دارند و تعیین می کنند که آیا ماکروهایی در فایلی موجود است، هنگام باز شدن، در حالت قرنطینه / غیر فعال / یا Block قرار گیرد یا خیر.

در ادامه شرح هر یک از گزینه ها آورده شده است:

1) Disable all macros without notification:

یعنی همه ماکروها رو غیر فعال کند و به کاربر هیچ هشدار یا پیغامی مبنی بر اینکه ماکروها غیر فعال هستند، را نمایش نخواهد داد.

2) Disable all macros with notification:

این تنظیم پیش فرض اکسل است و می گوید همه ماکروها غیر فعال باشند اما به کاربر پیغامی نمایش بده که بگوید ماکروهای فایل، غیر فعال شده اند و قابل اجرا نمی باشند.

3) Disable all macros except digitally signed macros:

همه ماکروها غیر فعال باشند بجز آنهایی که امضا دیجیتالی شده اند. من در ایران ندیدم که سازمانی امضای دیجیتال مایکروسافت را پیاده سازی کرده باشد. در واقع این امضای دیجیتال توسط واحد IT قابل پیاده سازی است و افراد سازمان می توانند فایل های خود را امضا دیجیتالی کنند و البته آن امضا فقط در داخل آن سازمان اعتبار دارد. دقت داشته باشید که ما امضای دیجیتال بین المللی و .. هم داریم اما بهتر است این بحث رو ما ادامه ندهیم زیرا در دنیای اکسل تا آنجایی که من دیده ام، اصلا متداول نیست.

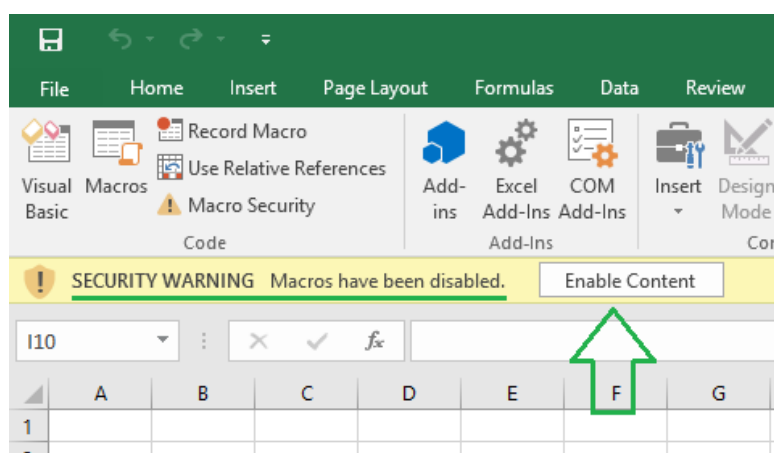
4) Enable all macros (not recommended; potentially dangerous code can run):

این گزینه یعنی ماکروهای همه فایل‌ها همواره فعال باشند و غیر فعال نباشد. به شما هشدار داده است از این گزینه استفاده نکنید زیرا که ممکن است کدهای خطرناکی اجرا بشوند.

فعال کردن یک ماکرو

در دلیل تنظیمات امنیتی ماکروها، اگر گزینه Disable all macros with notification را انتخاب کرده باشد، بلافاصله بعد از اینکه فایلی که ماکرو دارد را باز کنید، پیغام زرد رنگی مانند تصویر زیر به شما نمایش داده می‌شود:

هشدار قرنطینه/غیرفعال بودن ماکروی یک فایل

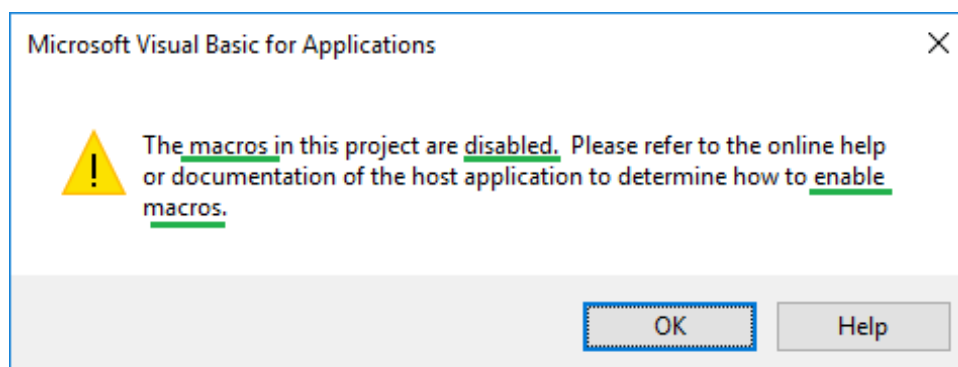


پیغام زرد رنگ رو اگر دقیقاً بخوانید به شما گفته است که Macros have been disabled. و معنای آن این است که :

- ۱) این فایل ماکرو دارد
- ۲) این ماکرو ممکن است که خطرناک باشد (Security Warning و علامت زرد سپر)
- ۳) ماکروی این فایل غیر فعال / قرنطینه شده است.
- ۴) چون ماکرو غیر فعال است بنابراین نمی‌توانید آن را اجرا کنید.
- ۵) اگر می‌خواهید با زدن کلید ALT+F11 وارد محیط VBE شوید و آنرا مطالعه کنید (آن جمله را خودم اضافه کردم و در پیغام این نوشته نشده است ☺)
- ۶) اگر هم که مطمئن هستید که این فایل معتبر است می‌توانید با زدن کلید Enable Macro، ماکرو را از حالت قرنطینه/غیرفعال خارج کنید و سپس آنرا اجرا نمایید.

اگر که شما به این پیغام زرد رنگ بی‌توجه باشید و گزینه Enable را نزنید، سپس بخواهید که یکی از ماکروهای این فایل را اجرا کنید، پیام زیر را به شما نمایش داده می‌شود.

تصویر پیغامی که به دلیل غیر فعال بودن ماکرو، قابل اجرا شدن نیست



ترجمه این پیغام:

"ماکروهای این پروژه (فایل) غیر فعال هستند. لطفاً به راهنمای نرم افزار جهت فعال کردن ماکروها مراجعه کنید."

نکته ۱: از Excel 2010 به بعد، اگر یکبار برای فایلی Enable Macro را بزنید، دیگر از شما این پیام را در مورد آن فایل نمی‌پرسد و در خاطرش می‌ماند که این فایل معتبر است.

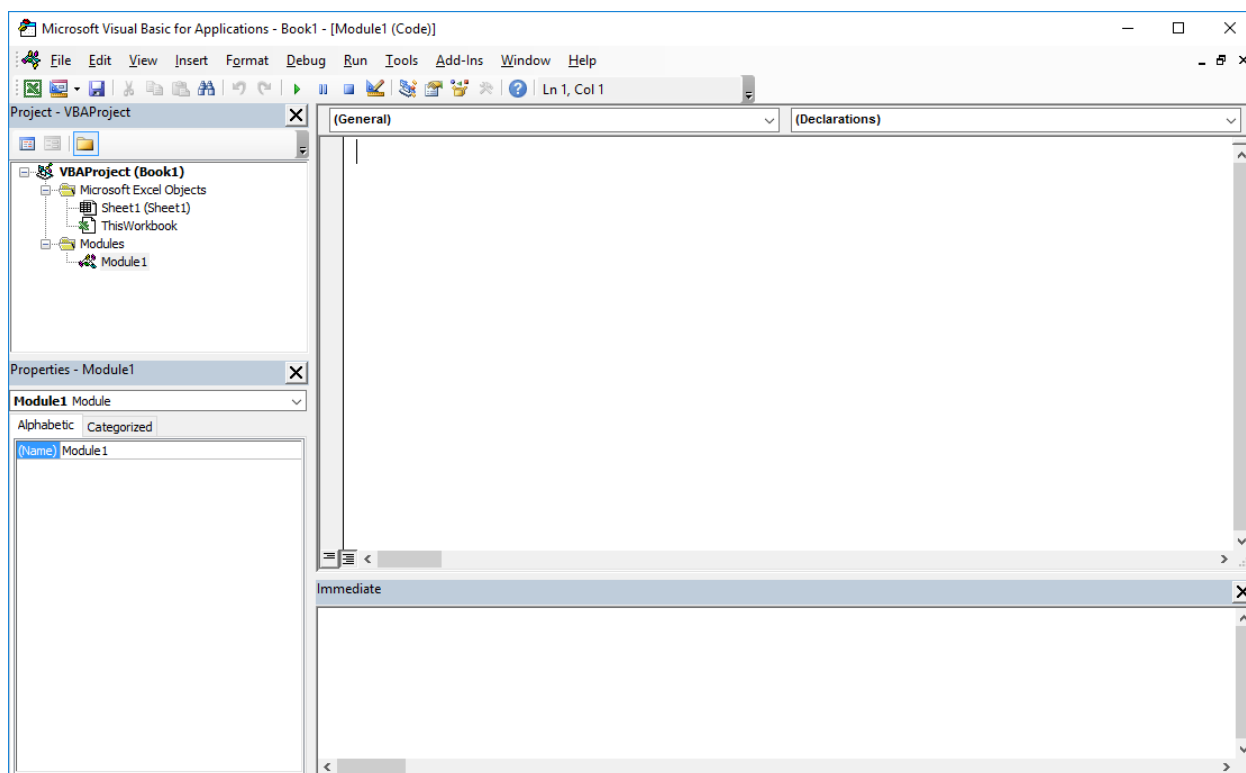
نکته ۲: اگر بلافاصله بعد از باز کردن یک فایل ماکروها را Enable نکنید، دیگر نمی‌توانید آنرا فعال کنید و باید فایل را ببندید و مجدد باز کنید تا پیغام برای شما نمایش داده شود.

توصیه: اگر هر روز با فقط با چند فایل که کاملاً معتبر هستند کار دارید، توصیه می‌کنم که تنظیمات ماکرو رو در حالت همیشه فعال قرار دهید در ضمن آنکه توجه داشته باشید که ویندوز، آنتی ویروس/ فایروال ها و تنظیماتی که واحد IT سازمان شما در نظر گرفته اند جلوی بسیاری از آسیب ها را خواهند گرفت.

آشنایی با Visual Basic Editor

باید بگوییم که VBE همان جایی که شما باید برنامه‌های خود را تایپ خواهید کرد و VBE در واقع Visual Basic Editor است و در دنیای کامپیوتر یعنی جایی که می‌توانید تایپ کنید.

عکس محیط VBE که در آن برنامه‌ها را تایپ خواهیم کرد



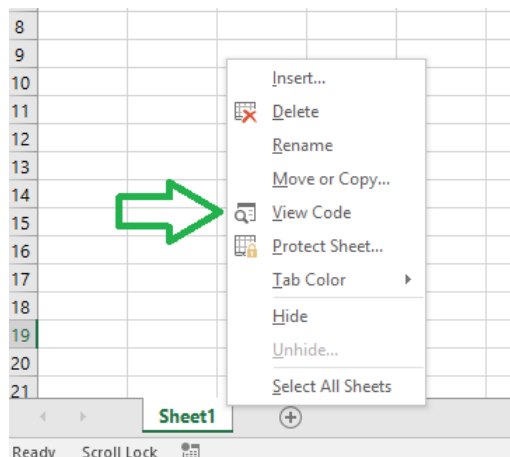
وارد شدن به محیط VBE

برای وارد شدن به محیط VBE ابتدا اکسل یا اکسس را اجرا می‌کنیم و سپس برای رفتن به محیط VBE چندین روش داریم :

- ۱) با زدن کلید ALT+F11 وارد محیط VBE خواهید شد.
توجه: اگر شما لپ تاپ دارید ممکن است که سازنده لپ تاپ شما استاندارد کلید F11 را عوض کرده باشد مثلاً زدن F11 باعث بلند شدن صدا می‌شود و برای زدن کلید F11 واقعی باید کلید دیگری را نگه دارید . معمولاً این کلید Fn است.

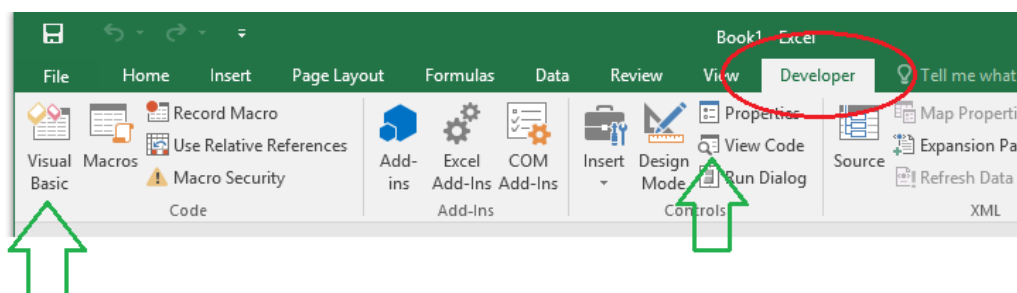
۲) اگر در اکسل روی یک نام شیت R-Click کنید و گزینه View Code را بزنید، وارد VBE خواهید شد.

عکس ورود به محیط VBE با R-Click روی یک نام شیت



۳) در اکسل از Developer Tab گزینه Visual Basic را انتخاب می‌کنیم.

عکس ورود به محیط VBE از Developer Tab

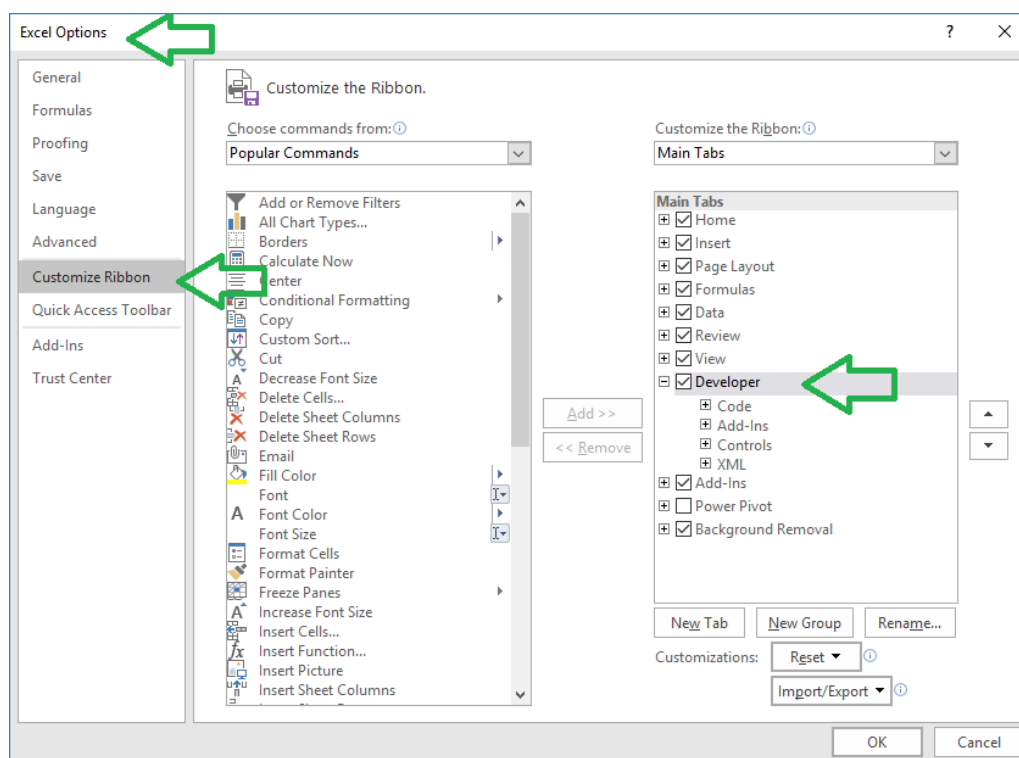


برای نمایش Developer Tab از مسیر زیر آنرا انتخاب کنید:

File → Options → Customize Ribbon

در نسخه Excel 2007 برای فعال کردن Developer Tab باید مسیری کمی متفاوت را طی کنید. برای یافتن آن یک گوگل کنید. (یعنی در گوگل جستجو کنید).

تصویر تنظیم فعال کردن نمایش Developer Tab در Ribbon



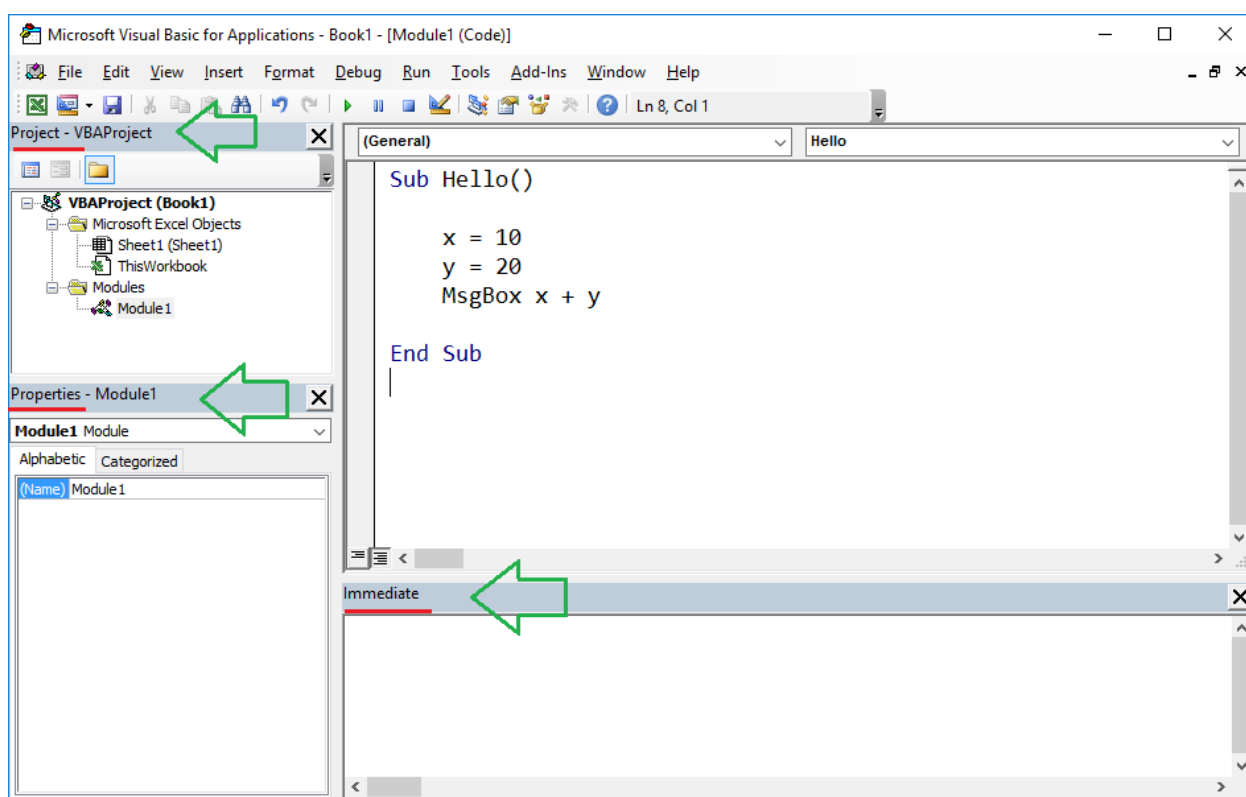
اجزای محیط VBE

در تصویر زیر من سه فلش گذاشته‌ام که هر کدام عنوان یک قسمت از محیط VBE هستند:

- ۱- قسمت Project
- ۲- قسمت Properties
- ۳- قسمت Immediate

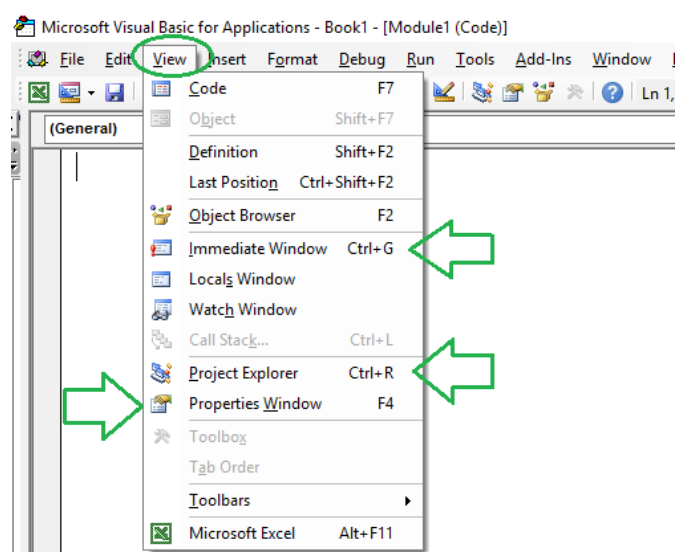
اگر خوب نگاه کنید هر یک از پنجره‌ها یک دکمه «ضربدر» که می‌توانیم آن قسمت را ببینیم و اگر آن قسمت را بستیم و خواستیم که مجدد آن را بیاوریم باید از منوی View این کار را انجام دهیم.

تصویری از محیط VBE



نکته: اگر احتمالاً یکی از این قسمت‌ها را بر روی دستگاه خود مشاهده نمی‌کنید، می‌توانید از منوی View که در تصویر بعدی مشاهده می‌کنید، بر روی آنها کلیک کنید تا فعال شوند.

تصویری از منوی View



تمرین ۱: لطفا یک نگاهی به همه منوها محیط VBE بیندازید. نگران نباشید این محیط بسیار ساده است.

تمرین ۲: پنجره Immediate را ببینید و سپس با کلید میانبر (shortcut key) مجدداً آنرا باز کنید. (یعنی ممکنه کسی باشه که بپرسه که کلید میانبرش چیه؟؟؟)