

Introduzione

Questo testo è rivolto a chi utilizza le versioni di Excel dalla 2007 alla 2019 per il proprio lavoro. È un utente che ha una certa dimestichezza con i comandi; sa utilizzare le formule, almeno le più semplici; sa ricavare un grafico dai dati. Non è un esperto informatico, piuttosto un “artigiano”: ha imparato a usare una tabella pivot dai colleghi, sa lavorare con un foglio di bilancio, oppure è in grado di eseguire una semplice analisi dati, magari utilizzando medie e percentuali. Se non sa fare qualcosa, prova a cercare la soluzione su Internet e in genere risolve il problema. Insomma, stiamo parlando dell’utente medio professionale dei fogli elettronici. In generale non sa programmare e non sa esattamente che cosa significhi questa parola in ambito informatico. Questo è un libro di *programmazione* per Excel, non insegna a usare Excel, lo dà abbastanza per scontato. Chi desiderasse imparare a usare Excel potrà contare su migliaia di testi che riempiono le librerie e anche Amazon.

Convenzioni utilizzate in questo libro

Prima di proseguire descriviamo alcuni termini usuali per chi lavora con Excel. Sono termini conosciuti, ma preferiamo descriverli per offrire a tutti una base di conoscenza. Per maggiore chiarezza, per questi elementi useremo un formato di stampa specifico, in modo che si capisca subito a che cosa ci stiamo riferendo.

- Excel usa delle parole specifiche per descrivere i comandi, le icone, le videate e così via. Sono le parole che descrivono gli elementi di interfaccia di Excel: noi le scriveremo in corsivo, mantenendone la stessa forma di maiuscolo e minuscolo proposta sulla videata da Excel. Per esempio, scriveremo *Barra multifunzione*, *Scheda sviluppo*, *Registratore Macro*, per indicare quegli elementi che vedrete sul monitor.
- Quasi sempre i comandi usano delle combinazioni di tasti e quindi se vogliamo indicare le azioni di *Copia* e *Incolla* scriveremo Ctrl + C e Ctrl + V per indicare di tenere premuto il tasto Ctrl (Control) e poi contemporaneamente il tasto C e così via. Excel accetta molte combinazioni di tasti utilizzando i tasti Ctrl, Alt, Shift (maiuscolo), anche in combinazione tra loro. Per esempio, la combinazione Ctrl + Maiusc + L serve a eseguire il comando *Filtro*.

- Per dare un comando, Excel utilizza una sequenza di azioni da eseguire su icone, videate o righe di comando. Se, per esempio, vogliamo centrare in orizzontale un testo in una cella utilizzeremo la forma seguente: *Home > Allinea al centro orizzontale*. In altre parole, una sequenza di comandi sarà scritta in corsivo e ciascun elemento sarà separato dal segno >.
- Questo è un libro di programmazione e come tale sarà pieno di righe di codice, come le seguenti:

```
Sub Somma()  
    a = 2  
    b = 3  
    Sum = a + b  
    MsgBox( "La somma è: " & Sum)  
End Sub
```

Quando scriveremo delle istruzioni per il computer, useremo questo tipo di carattere, in modo che sia facilmente distinguibile dal resto del testo.

Compatibilità delle versioni Excel con questo libro

Esistono molte versioni di Excel: dall'anno 2000 al 2019 ogni tre anni circa Microsoft ha fatto uscire una nuova versione dell'applicativo, in genere abbastanza compatibile con quelle precedenti. Abbastanza, ma non del tutto. Ci sono stati cambiamenti anche profondi nel corso di tutti questi anni. Per esempio, prima del 2007 un foglio di Excel conteneva 65.536 righe e la sua ultima cella aveva indirizzo (colonna scritta in lettere, riga scritta in cifre) IV65.536. Dal 2007 le righe e le colonne sono aumentate tantissimo, oggi abbiamo 1.048.576 righe e 16.384 colonne e l'ultima cella avrà indirizzo XDF1.048.576 (sono più di 17 miliardi di celle). Quindi, se utilizzate una versione precedente al 2007 siete eroi della resistenza, ma questo libro non fa per voi. Se utilizzate le versioni 2007 o 2010 alcuni degli aggiornamenti di VBA non possono funzionare su quelle versioni e quindi le macro che verranno proposte qui potrebbero bloccarsi. Le versioni 2013, 2016 e 2019 vanno bene. Alla fine dell'Introduzione troverete un breve paragrafo che insegna a riconoscere quale versione di Excel state utilizzando.

Programmare Excel con VBA e le macro

VBA (*Visual Basic for Applications*) è un sottoinsieme del linguaggio Visual Basic (oggi è Visual Basic 2019) che Microsoft ha creato per sviluppare speciali programmi chiamati *macro* da utilizzare con Excel ma anche con tutti gli altri applicativi del pacchetto Microsoft Office, come per esempio, Word, Access, PowerPoint e così via. Acquistando Office, VBA è già compreso nel pacchetto, ma non può lavorare al di fuori di Office. Il linguaggio di programmazione VBA viene utilizzato per creare funzioni di Excel, per eseguire i comandi di tutti gli applicativi di Office, per eseguire qualunque operazione in modo automatico, esattamente come un qualsiasi linguaggio di programmazione, di quelli utilizzati dai programmatori per creare tutti i programmi che quotidianamente si

utilizzano sui PC. In questo libro utilizzeremo VBA esclusivamente con Excel, creando procedure chiamate *macro*, che permetteranno di rendere automatiche alcune operazioni che sceglieremo. Prima di proseguire, chiariamo un aspetto importante: questo libro non è una guida completa alla programmazione in VBA. Tratteremo soltanto un sottoinsieme del linguaggio, la parte più comune, ma con questa sarete in grado di risolvere la maggior parte dei problemi che affronterete nel lavoro. Una guida completa presupporrebbe una competenza di gran lunga superiore a quella che chiediamo qui. Per risolvere un problema di calcolo o per automatizzare un'operazione servono due competenze principali:

- saper analizzare il problema da un punto di vista razionale e logico, scomponendolo in operazioni semplici, ciascuna delle quali composta da una successione ordinata di singole operazioni, che, eseguite una dopo l'altra, consentono di arrivare all'obiettivo proposto;
- tradurre le operazioni semplici in istruzioni di una procedura, scritte utilizzando un linguaggio di programmazione.

Anche se questo secondo punto può sembrare il più difficile dei due, in realtà è la prima parte quella più critica. In questo libro analizzeremo e illustreremo come lavorare con entrambi questi aspetti. Nella prima parte entreremo in dettaglio del linguaggio VBA, che, per fortuna, è un linguaggio semplice da utilizzare, alla portata di quasi tutti; in più gli autori cercheranno il più possibile di renderlo immediato ed elementare. Partiremo da zero, non dando nulla per scontato, in quanto ci rivolgiamo a non programmatori. Lo scopo del testo è quello di fornire tutte quelle conoscenze di base necessarie per diventare entro poco tempo autonomi nello sviluppare un progetto di piccole o medie dimensioni. Man mano che andremo avanti con i capitoli sarà il primo aspetto a diventare protagonista, perché programmare vuol dire prima di tutto analizzare il problema, costruendone uno schema concettuale composto da blocchi più semplici, da tradurre poi in sequenze operative di istruzioni. Nell'ultima parte del testo tratteremo alcuni problemi che gli autori hanno dovuto affrontare nella loro attività di consulenti, tutti problemi risolti con la partecipazione attiva e consapevole degli utenti che poi hanno utilizzato i sistemi prodotti.

NOTA

Le macro *si eseguono* oppure *si lanciano* o ancora *si fanno partire* o *si richiamano*. Potete usare uno qualunque di questi termini. Nel libro verranno utilizzati tutti.

Che cosa sono le macro in Excel e perché utilizzarle

Excel è uno strumento di grande versatilità, utilizzato per le attività più disparate; se facessimo un censimento tra i lettori di questo libro ne scopriremmo tanti usi differenti, il cui elenco potrebbe riempire un intero capitolo. Anche se i lettori hanno vari tipi di aspettative, anche se Excel è usato in una pluralità di modi, tutti i lettori hanno sicuramente la stessa esigenza e il desiderio di poter automatizzare almeno una parte delle proprie attività. Se vi capita spesso di prendere una tabella, ricavarne un grafico e poi di stamparlo, potreste voler automatizzare questi tre passaggi in modo che vengano eseguiti in pochi istanti solo premendo un tasto. Una macro può automatizzare processi ripetitivi come impostare una tabella, scrivere una fattura, compilare un modulo, preparare un grafico o

una tabella pivot, ricavare un bilancio da una tabella dati e moltissime altre cose ancora. Possiamo dire che *una macro è costituita da un blocco di istruzioni scritte nel linguaggio VBA in grado di automatizzare un compito specifico eseguendolo in pochi istanti*. Certo, per scrivere una macro bisogna imparare a programmare, ma come abbiamo detto, partendo da zero cercheremo di rendere quest'aspetto accessibile a tutti e facilmente comprensibile.

NOTA

I termini *macro*, *procedura*, *blocco di istruzioni*, *programma* in questo testo vengono considerati come equivalenti e verranno usati sovente uno al posto dell'altro.

Che cosa potete fare con VBA e le macro

- Chi utilizza Excel, spesso ripete continuamente le stesse operazioni, impiegando abbastanza tempo: copiare dati in una tabella e formattarla, ricavare dati di sintesi e di consolidamento da tante tabelle diverse, riordinare i dati dell'archivio del magazzino, preparare un report delle vendite o degli acquisti corredandolo con i grafici opportuni, predisporre un diagramma di Gantt per lo sviluppo di un progetto e tantissime altre cose.
- Le macro possono automatizzare tutte queste operazioni che si svolgono ripetutamente.
- Spesso può capitare di dover svolgere una stessa operazione su molte cartelle diverse: per esempio un controllo sui dati, un'operazione di conteggio dei casi, l'eliminazione di righe o colonne non essenziali e via di seguito.
- Le macro possono automatizzare queste operazioni ripetitive.
- Dovete eseguire una stessa sequenza di comandi più volte o periodicamente?
- Potete costruire una macro che esegua tutta la sequenza dei comandi, semplicemente facendo clic su un'icona.
- Potete costruire una macro che crei una funzione personalizzata con una formula creata da voi.

Tutti questi sono esempi, ma comprendono abbastanza bene il campo di utilizzo delle macro, un campo che virtualmente non ha confini, perché riguarda tutte le potenzialità di Excel. Abbiate la pazienza di seguirci nei prossimi capitoli, dove svilupperemo tanti strumenti da utilizzare così come sono, oppure da inserire in procedure più complesse costituite da una successione di macro.

Principali vantaggi nell'impiego delle macro

Programmare le macro in Excel presenta molti vantaggi. Eccone alcuni.

- Una macro esegue il proprio compito sempre nello stesso modo, sempre nello stesso tempo, senza mai sbagliare nulla.
- Una macro esegue il proprio compito in pochi istanti, un tempo molto inferiore a quello impiegato anche dal "campione mondiale di Excel".

- Una macro corretta può essere eseguita da chiunque, nel vostro ufficio: la potete passare anche a chi non sa utilizzare Excel e la macro eseguirà lo stesso il proprio compito.
- Lanciate una macro e mentre lei lavora voi potete occuparvi d'altro. La macro eseguirà lo stesso il suo compito.

Se pensate che questo sia una sorta di paradiso, passiamo subito in rassegna alcuni (pochi per fortuna) degli svantaggi nell'uso delle macro, come dire il loro "lato oscuro".

- Bisogna imparare a programmare. Questo può spaventare qualcuno, ma vedrete che non è così difficile come sembra e soprattutto gli autori faranno di tutto per rendere l'apprendimento leggero e semplice.
- Una macro funziona per quello per cui è stata programmata: se cambiate una cella, un foglio, un valore, senza averlo previsto prima, la macro può andare in errore. Dovremo imparare quindi a eseguire il controllo degli errori (gli informatici dicono *fare il debug*), eseguire la macro un passo per volta fino a trovare l'errore. In questo, un aiuto prezioso verrà dato dal VBE.
- Costruita una macro con una versione di Excel, potrebbe non funzionare più con le versioni successive. Microsoft fa evolvere continuamente Excel e il VBA, da una versione all'altra, in genere mantenendo sempre una grande compatibilità, ma poi il tempo passa e dopo due o tre versioni, può diventare necessario intervenire daccapo sulla macro per modificarne alcune parti e renderle compatibili con il nuovo Excel. Ma, ripetiamo, ci vogliono anni per avere cambiamenti significativi, state tranquilli.

Come è fatta una macro?

Sì, va bene, abbiamo capito che esistono queste "macro", che è possibile utilizzarle per programmare Excel. Ma... in pratica come è fatta una macro? Da che cosa è composta? Una macro è sempre composta da istruzioni, in genere poste una per riga: inizia con la parola *Sub* seguita dal nome della macro e da una coppia di parentesi tonde. Una macro termina sempre con le parole *End Sub*.

```
Sub Somma()
  a = 2
  b = 3
  Sum = a + b
  MsgBox( "La somma è: " & Sum)
End Sub
```

Questa macro si chiama *Somma()* e non fa altro che assegnare il valore 2 alla variabile *a* ($a = 2$) e il valore 3 alla variabile *b* ($b = 3$). Poi assegna il risultato della somma alla variabile *Sum* ($Sum = a + b$) e infine apre una finestra (*MsgBox* dall'inglese *Message Box*) in cui scriverà "La somma è 5". Si tratta di una macro che esegue un compito elementare, ma qui il nostro scopo è solo quello di mostrare in modo pragmatico che cosa sia e come si presenta una macro.

L'ambiente di sviluppo delle macro di Excel: il VBE, Visual Basic Editor.

Il *Visual Basic for Applications* o *VBA* è il linguaggio di programmazione per sviluppare le macro in Excel e più in generale in tutti gli applicativi del pacchetto Office. Ma per programmare non basta disporre di un linguaggio di programmazione: è estremamente importante avere uno strumento, un applicativo che consenta di scrivere facilmente i programmi, che consenta di provarli mano a mano che vengono scritti, che corregga la sintassi mettendo in evidenza gli errori di battitura, che permetta di individuare gli errori logici di programmazione e infine che inserisca la macro all'interno di Excel e ne consenta l'esecuzione. Questo fondamentale applicativo per lo sviluppo delle macro si chiama *Visual Basic Editor* o *VBE* ed è compreso nel pacchetto Office. Non si può usare il linguaggio di programmazione VBA senza disporre dell'editor VBE. Il VBE è un programma a sé stante che esiste solo quando Excel è attivo e si chiude quando chiudiamo Excel. In un Excel impostato normalmente, l'ambiente di sviluppo VBE, anche se è presente, è nascosto e occorre attivarlo per inserirlo nella *Barra multifunzione* (Figura I.1). Fate clic sul comando *File*, il primo comando all'estrema sinistra mostrato nella Figura I.1, della *Barra multifunzione*. Si aprirà la finestra rappresentata nella Figura I.2, che contiene molti comandi per l'apertura, il salvataggio, la stampa delle cartelle di lavoro e anche il comando *Opzioni* per la configurazione di Excel.

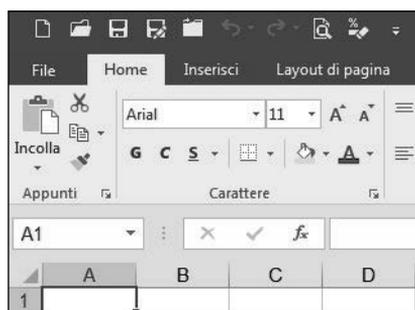


Figura I.1 Nella Barra multifunzione il primo comando a sinistra è il comando File, che consente di aprire la finestra rappresentata nella Figura I.2.

Dopo aver individuato il comando, fate clic su di esso per far aprire la finestra rappresentata nella Figura I.3, che contiene tutte le possibili impostazioni di Excel.

Una delle più importanti e che riguarda proprio il nostro argomento è la riga *Personalizzazione barra multifunzione*, che farà comparire nella parte destra del video due schede, utilizzabili per inserire o togliere comandi nelle varie schede della *Barra multifunzione*. Nella parte destra sono elencate le schede esistenti (ogni utente può costruire e aggiungere le proprie schede facendo clic su *Nuovo*). Tra queste, la scheda *Sviluppo* in genere non è attiva: fate clic sul quadrato per apporre un segno di spunta e attivare la scheda *Sviluppo*, che vediamo mostrata nella Figura I.4. All'estrema sinistra si trova l'icona *Visual Basic*: facendo clic su di essa si apre la finestra del *Visual Basic Editor*, il VBE che utilizzeremo in tutto il testo come strumento base per creare, modificare ed eseguire le macro che svilupperemo.

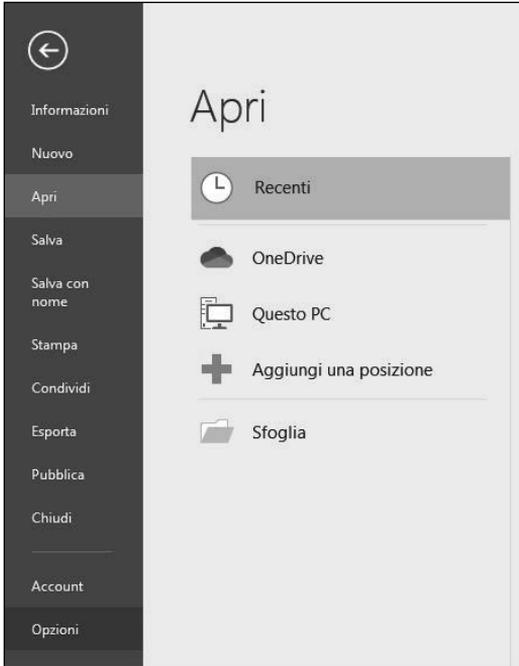


Figura I.2 La finestra contiene molte azioni per il salvataggio, la stampa e soprattutto per la configurazione di Excel.

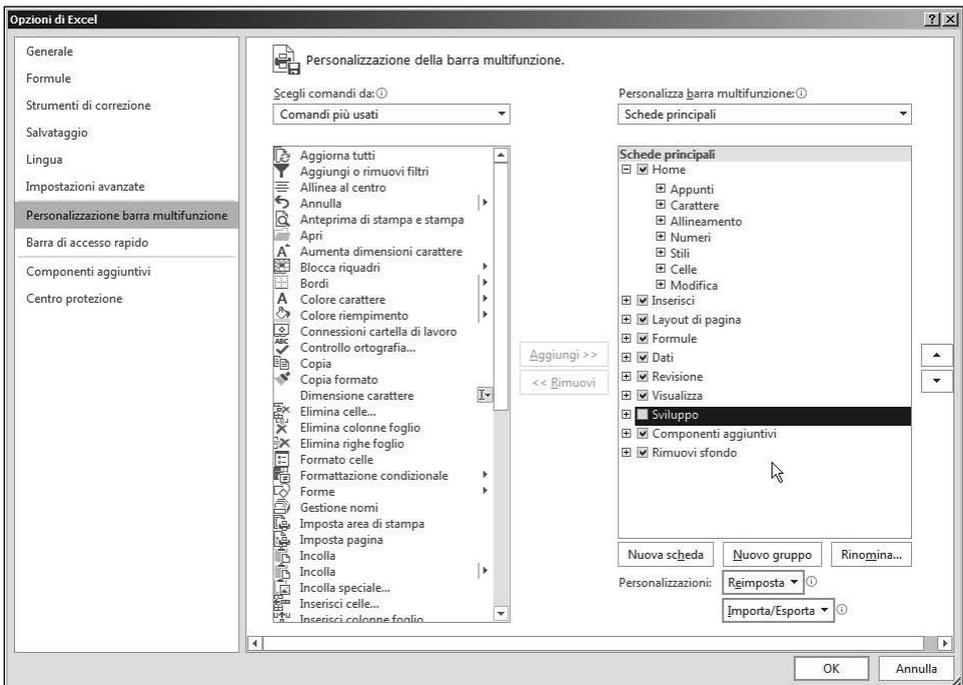


Figura I.3 La finestra per la configurazione generale di Excel.

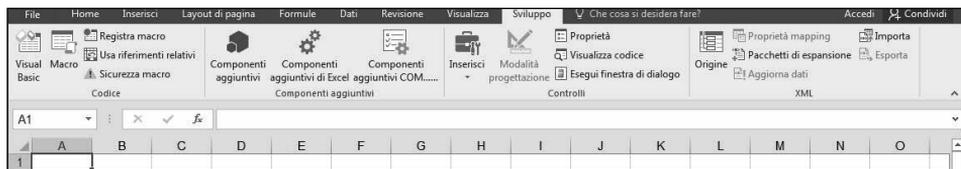


Figura I.4 La scheda Sviluppo consente di creare, modificare ed eseguire le macro. Contiene molti strumenti utili per la programmazione.

La Figura I.5 mostra la videata del VBE, composta da tre parti, ognuna dedicata a un aspetto della programmazione. A destra si trova la finestra del codice, dove verranno scritti i programmi da eseguire. In questa finestra si trova il codice della semplice macro presentata nel paragrafo precedente. Da notare come la scrittura di un programma segua una particolare *indentazione*, con alcune righe rientrate, alcune righe vuote e altro ancora. Una disposizione così articolata rappresenta un aspetto importante per rendere più semplice la lettura di un programma. Nel Capitolo 5 verranno spiegate le regole di formattazione che sono valide per qualunque tipo di programma e linguaggio di programmazione. La finestra in alto a sinistra è chiamata *Progetto*: qui sono elencate tutte le cartelle di lavoro che Excel utilizza per la sua sessione di lavoro. Come spiegheremo nel Capitolo 1 e soprattutto nel Capitolo 2 esaminando a fondo il contenuto di questa finestra, Excel per il suo funzionamento ha bisogno di alcune cartelle, la maggior parte nascoste all'utente, che contengono tutte le funzioni e istruzioni. La finestra *Progetto* può contenere molti altri oggetti, tutti indispensabili per programmare le macro. Con un po' di pazienza impareremo a crearli e usarli. La finestra in basso a sinistra è *Proprietà*: elenca per ciascuno degli oggetti presenti nella finestra *Progetto* le proprietà che possono essere modificate direttamente dall'utente. Per un approfondimento di quanto detto rimandiamo al Capitolo 3.

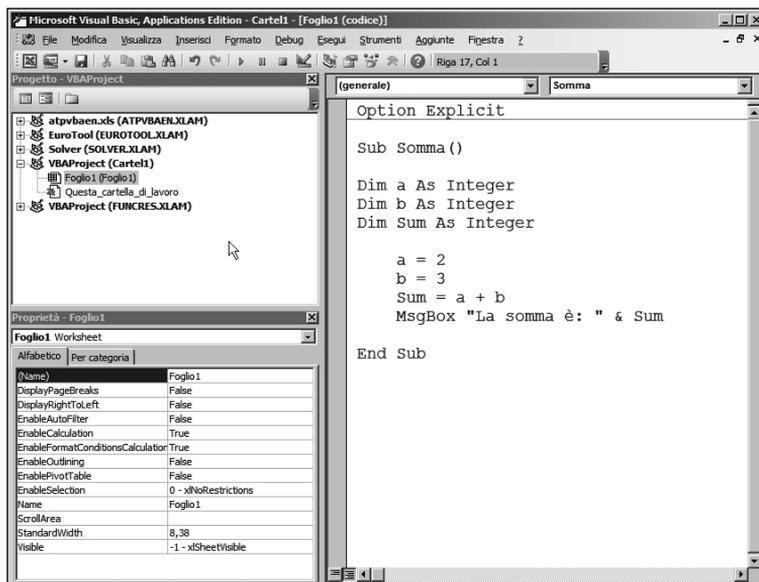


Figura I.5 La finestra del VBE con la sua suddivisione in componenti.

Un aspetto che si deve ulteriormente sottolineare è che il VBE, è un applicativo a sé stante, che viene eseguito dando il comando *Visual Basic* come abbiamo visto prima. Può lavorare con uno qualunque dei pacchetti dell'applicativo Office, ma non può esistere senza uno di essi: prima si deve avviare Excel, poi VBE e quando si chiude Excel anche VBE verrà automaticamente chiuso. In commercio esistono molti programmi di *editing* il cui scopo è quello di fornire un ambiente di sviluppo per la costruzione, modifica e correzione dei programmi. In genere possono lavorare con molti linguaggi (ne esistono decine) dal Visual Basic, al C++, a Python a PHP e così via. L'editor VBE è costruito per lavorare esclusivamente con VBA.

Preparativi necessari per utilizzare il VBA

Prima di iniziare a costruire la prima macro dobbiamo imparare a eseguire alcuni passaggi obbligatori per l'uso del codice VBA, altrimenti Excel si bloccherà. Prima abbiamo detto che le macro sono molto utili nel lavoro quotidiano e hanno molti aspetti positivi; questo è vero, ma sono anche potenziali portatrici di virus. Non certo quelle che costruirà il lettore, ma quelle che possono arrivarci dall'esterno, da un messaggio di posta elettronica o dal Web. Le macro sono programmi che Excel (o Word o Access e così via) possono eseguire in modo automatico e di nascosto, appena viene attivato l'applicativo e possono arrecare danni consistenti al contenuto dell'hard disk del vostro computer. Per risolvere almeno parzialmente questo problema, Microsoft ha imposto due condizioni: la prima riguarda il tipo di estensione del file Excel che contiene macro; la seconda sono le impostazioni di sicurezza dell'applicativo. Vediamole separatamente.

Salvare una cartella di lavoro che contiene macro

Quando salvate una cartella di lavoro, dalla versione 2007 Excel propone l'estensione .xlsx. Se tentate di salvare una cartella di lavoro contenente macro, Excel farà comparire il messaggio di errore rappresentato nella Figura I.6.

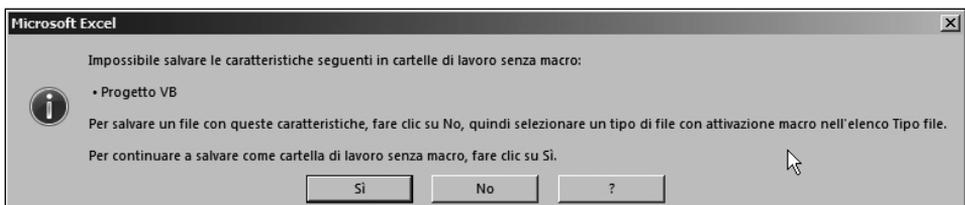


Figura I.6 Il tentativo di salvare una cartella contenente macro con l'estensione .xlsx farà comparire questa finestra d'errore. Forzando il salvataggio (facendo clic su Sì), Excel salverà la cartella di lavoro, ma senza le macro presenti nel VBE.

Se forzaste il salvataggio facendo clic su Sì, Excel utilizzerrebbe l'estensione .xlsx, ma eliminerebbe tutte le macro presenti: meglio non farlo. Per includere le macro in una cartella Excel bisogna scegliere il formato .xlsm (dove la m sta per *macro*) utilizzando il comando *Salva con nome* che mostrerà la schermata rappresentata nella Figura I.7. Facendo clic sul menu a tendina della seconda riga compariranno tutte le estensioni possibili

associate ai diversi formati di salvataggio di una cartella di lavoro. Nella seconda riga compare l'estensione `.xlsm`, che consente di salvare insieme alla cartella anche le macro presenti nell'editor VBE. Nella Figura I.8 potete osservare le due differenti icone che Excel utilizza per conservare le cartelle contenenti macro (a sinistra), contrassegnate da un punto esclamativo come segno di potenziale pericolo e le cartelle senza macro (a destra).



Figura I.7 Utilizzando il comando Salva con nome potete agire sulla seconda riga, dove compare l'elenco di tutti i formati di salvataggio possibili. Selezionate `.xlsm` per salvare anche le macro insieme alla cartella di lavoro.

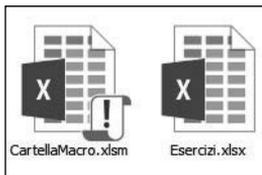


Figura I.8 I due tipi di icone che contraddistinguono le cartelle con macro (a sinistra) e senza macro (a destra).

Gestire le impostazioni di sicurezza

Il secondo aspetto della gestione della sicurezza riguarda la possibilità di bloccare le macro e la loro eseguibilità direttamente in lettura. Per controllare e modificare le impostazioni di sicurezza fate clic sul comando *File* (vedi la Figura I.1) all'estrema sinistra della *Barra multifunzione* per far comparire la finestra rappresentata nella Figura I.2: selezionate la riga *Opzioni* per far comparire la finestra rappresentata nella Figura I.3 e fate clic sulla riga *Centro protezione*. Comparirà la finestra rappresentata nella Figura I.9, dove dovete fare clic sul pulsante *Impostazioni Centro protezione*.

Al termine di questi passaggi si aprirà finalmente la finestra rappresentata nella Figura I.11 che presenterà quattro possibili opzioni per la gestione delle macro. La configurazione base di Excel è impostata come si vede in figura: la prima opzione blocca l'apertura di tutte le macro e non avvisa nemmeno l'utente della loro presenza.

Una via più immediata per arrivare alla finestra rappresentata nella Figura I.11 consiste nel fare clic sul pulsante triangolare giallo *Sicurezza macro* che trovate sulla sinistra della scheda *Sviluppo* (Figura I.10).

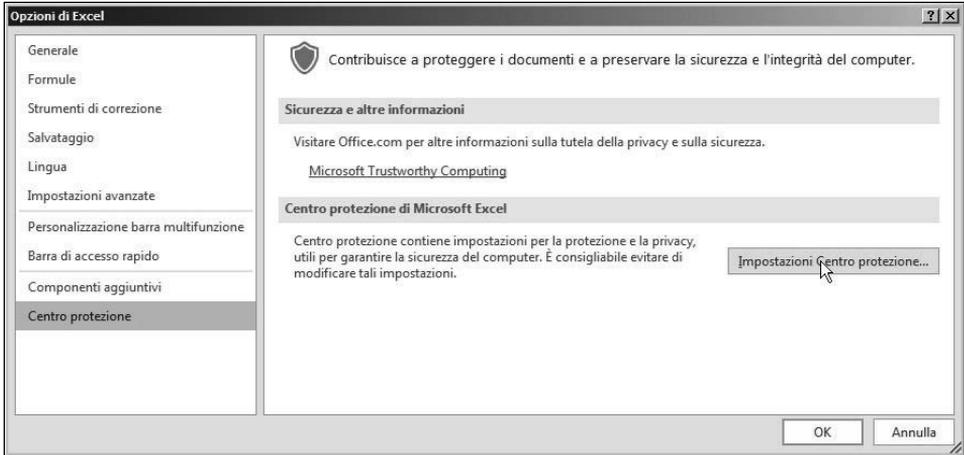


Figura I.9 La finestra che consente di aprire il Centro protezione.

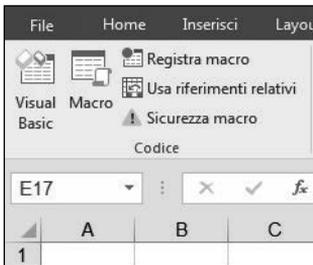


Figura I.10 Sulla Barra multifunzione > Sviluppo (scheda Codice) trovate il pulsante Sicurezza macro. Facendo clic su di esso aprirete direttamente la finestra Centro protezione, rappresentata nella Figura I.11.

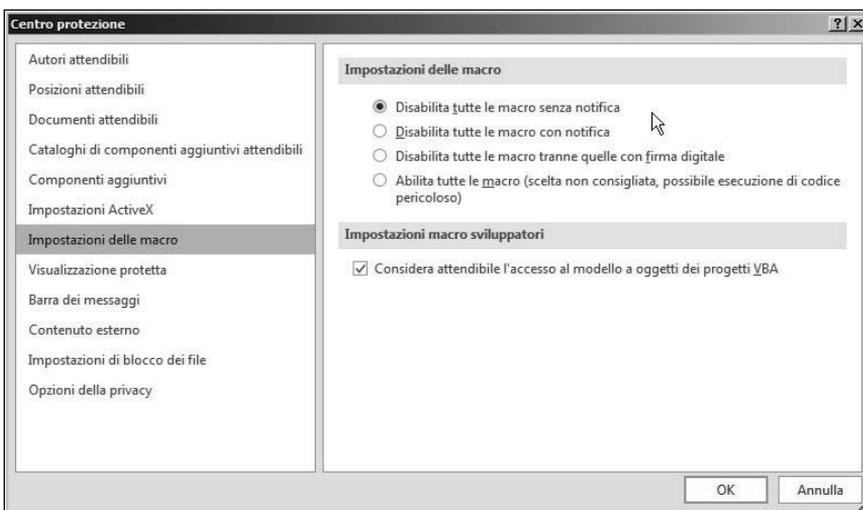


Figura I.11 La finestra Centro protezione contiene le opzioni per la gestione della sicurezza delle macro. La figura mostra l'impostazione base.

Osservando la Figura I.11, le opzioni possibili sono quattro, vediamo il significato.

- *Disabilita tutte le macro senza notifica*: le macro non funzioneranno e non verrà dato nessun messaggio sulla loro presenza. Questa è l'opzione predefinita di Excel. Impedisce alle macro di funzionare: massima sicurezza in cambio di assoluta rigidità.
- *Disabilita le macro con notifica*: questa opzione consente di lavorare con le macro ed è quella consigliata. Se tentate di aprire un file che contiene macro, compare il messaggio rappresentato nella Figura I.12: sarete avvisati della presenza di macro e potete, solo se lo ritenete sicuro, dare l'autorizzazione alla loro esecuzione. Questa è l'opzione consigliata per seguire questo libro. Naturalmente per fare clic su *Abilita contenuto* dovete essere sicuri dell'origine della cartella Excel: o perché è un vostro prodotto o perché siete certi della sua provenienza.
- *Disabilita tutte le macro tranne quelle con firma digitale*: una cartella di lavoro può essere firmata digitalmente, se avete la possibilità di accedere alla firma digitale. In questo caso chi riceve la cartella di lavoro può considerarla sicura.
- *Abilita tutte le macro (scelta non consigliata...)*: questa è la scelta più pericolosa, perché Excel non effettua nessun controllo e potrebbe far partire in automatico una macro contenente codice pericoloso.

Questo sistema di opzioni può creare qualche problema: per esempio dopo aver costruito le vostre macro sul vostro computer, volete passarle a qualche vostro collega, che vi dirà subito che la vostra macro non funziona. In realtà andrà solo abilitata l'esecuzione delle macro. Dovrebbe comparire la segnalazione rappresentata nella Figura I.13, che avvisa solo del fatto che state tentando di eseguire una macro. Per proseguire basta cambiare l'impostazione nella finestra rappresentata nella Figura I.11.

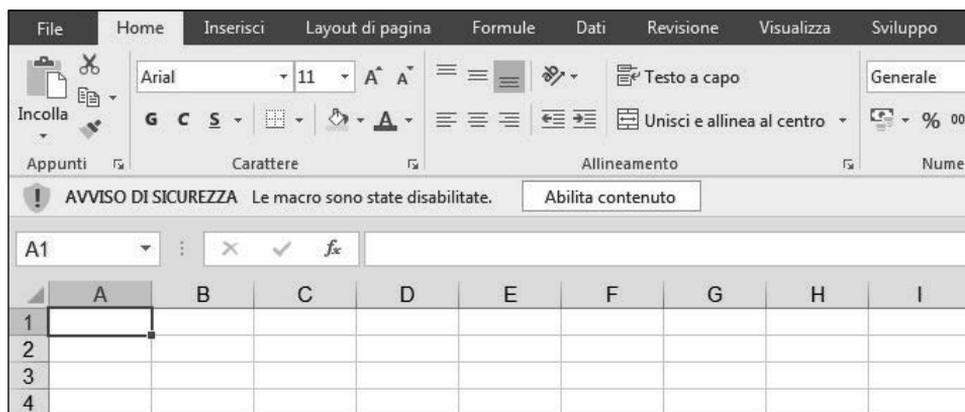


Figura I.12 Se è attivata la seconda opzione mostrata nella Figura I.11, aprendo una cartella di lavoro contenente macro compare la segnalazione mostrata. Se siete sicuri della provenienza del file, potete fare clic su *Abilita contenuto*, per consentire l'esecuzione delle macro.

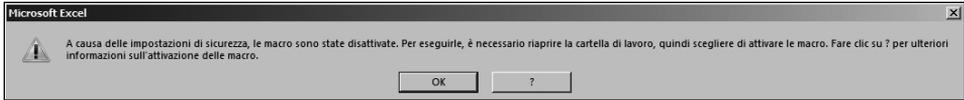


Figura I.13 Tentando di aprire una cartella di lavoro con le macro disabilitate (opzione 1) compare questo avvertimento.

Esiste ancora un'opzione molto valida da poter utilizzare: creare una cartella sicura dove poter riversare i propri file Excel (per esempio quelli scaricati dal sito di questo libro) senza dover cambiare le impostazioni di ciascun file. Per farlo osservate la Figura I.14; fate clic, a sinistra, su *Posizioni attendibili*, per far comparire l'elenco delle cartelle del computer che Excel utilizza per i propri file di sistema. A questo elenco potete aggiungere una o più cartelle facendo clic sul comando *Aggiungi nuova posizione* e poi scegliendo la cartella. Tutti i file contenuti in questa cartella verranno considerati sicuri da Excel. Per esempio, se scaricate dal sito web di Apogeo i file degli esercizi, potete inserirli in questa cartella, senza dover cambiare le impostazioni di ogni singolo file.

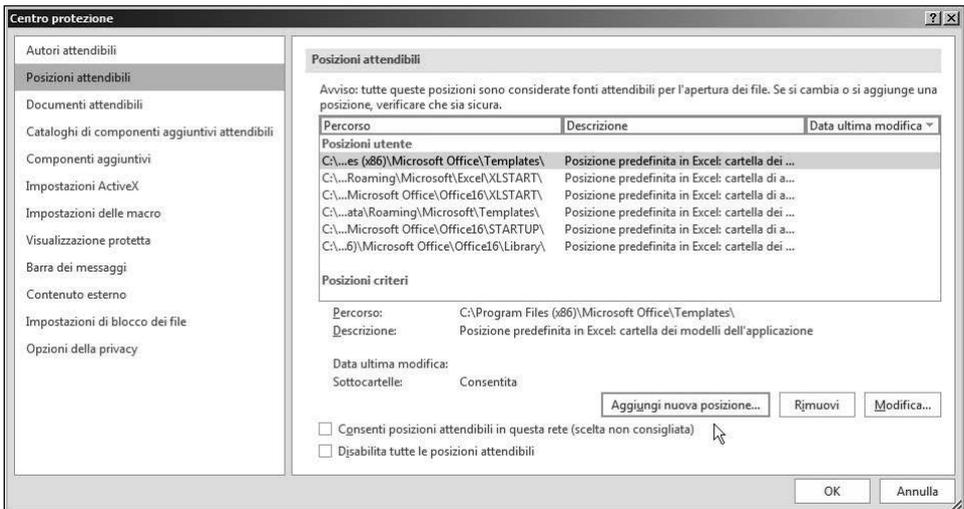


Figura I.14 La riga Posizioni attendibili mostra l'elenco delle cartelle che Excel utilizza per i propri file di sistema.

Riconoscere la versione di Excel in uso

Sapere con quale versione di Excel si sta lavorando diventa importante per capire quali vincoli si avranno rispetto alla programmazione con VBA. Conoscere la versione utilizzata è molto semplice: facendo clic su *File > Account* comparirà la finestra rappresentata nella Figura I.15, che contiene la configurazione e le impostazioni dell'applicativo Office. Sulla parte destra potete osservare la riga contenente la versione di Excel che utilizzate e l'anno di edizione. Un'altra possibilità si ha premendo il tasto *F1* per attivare la Guida di Excel, la cui pagina iniziale è mostrata nella Figura I.16: nella prima riga è riportato l'anno di edizione di Excel.

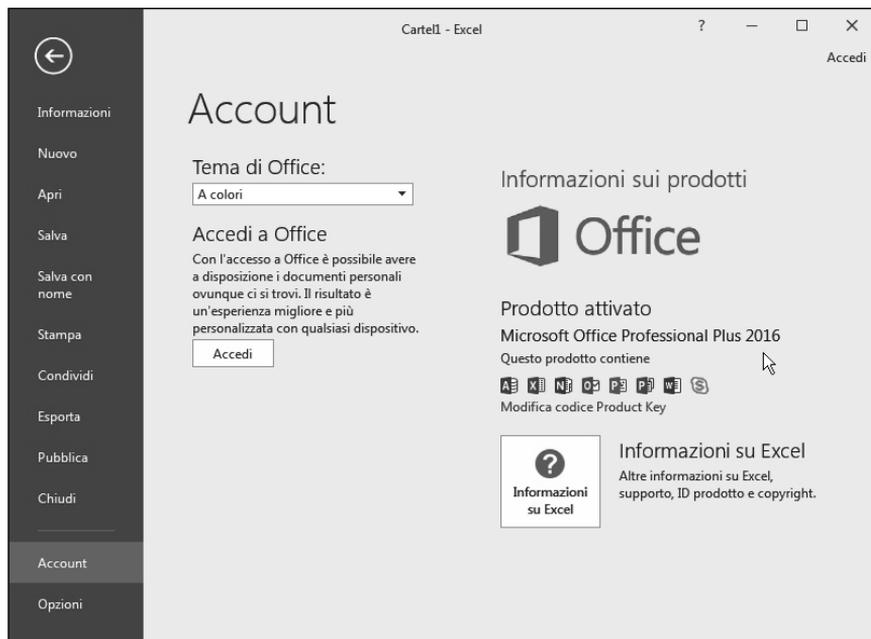


Figura I.15 La finestra Account contiene indicazioni sulla configurazione e le impostazioni di Excel. A destra notate la versione e l'anno di edizione.



Figura I.16 Premendo il tasto F1 aprirete la pagina iniziale della Guida, che riporta l'anno di edizione.

In questa introduzione

- Convenzioni tipografiche e di stile utilizzate in questo testo.
- Versioni di Excel compatibili con questo testo.
- Che cosa potete fare con le macro.
- Vantaggi e svantaggi nell'uso delle macro.
- Prima presentazione dell'ambiente di sviluppo delle macro, VBE.
- Salvataggio di una cartella di lavoro con macro.
- Impostazioni di sicurezza di una cartella di lavoro.
- Riconoscere la versione di Excel in uso.