



TÓPICOS DE EXCEL (Versión 2019)

**FILTROS - FUNCIONES - TABLAS DINÁMICAS
BUSINESS INTELLIGENCE - MACROS - VBA**



Diego A. Parrás



cece uba - nuevo espacio



@ceceuba



ceceubanuevoespacio

Material gratuito para ser distribuido exclusivamente por:

Nuevo Espacio

Los estudiantes que hacemos del CECE el mejor Centro de la UBA





Contenido

1.	Introducción.....	4
1.1.	Tipos de datos en las planillas de cálculo	4
1.2.	Celdas, Rangos y Tablas	6
1.3.	Los formatos y sus propiedades	8
1.4.	Formatos condicionales.....	9
1.5.	Filtros	10
1.5.1.	Filtros de texto.....	10
1.5.2.	Filtros de Números	11
1.5.3.	Filtros por color.....	11
1.6.	Filtros Avanzados.....	12
1.7.	Subtotales.....	14
2.	¿Qué son las funciones en Excel?	15
2.1.	¿Cómo invocar a una función?	15
2.2.	Tipos de funciones.....	17
2.2.1.	Funciones de Búsqueda y Referencia.....	17
2.2.2.	Funciones de Fecha y Hora	18
2.2.3.	Funciones de Texto.....	20
2.2.4.	Funciones Estadísticas	21
2.2.5.	Funciones Financieras.....	26
2.2.6.	Funciones Lógicas	28
2.2.7.	Funciones Matemáticas y Trigonométricas.....	29
2.3.	Diferencia entre funciones y fórmulas	31
2.4.	Caracteres de TEXTO dentro de las funciones y fórmulas.....	31
2.5.	Errores en las Funciones y Fórmulas en Excel	32



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



2.6.	Funciones recursivas y dinámicas.....	35
2.6.1.	Funciones recursivas.....	35
2.6.2.	Funciones dinámicas.....	37
2.6.3.	Ejemplo del uso de la función AGREGAR.....	39
3.	Tablas dinámicas y Herramientas Business Intelligence en Excel	41
3.1.	¿Qué son las tablas dinámicas en Excel?.....	41
3.2.	¿Cómo activar la Herramienta Tablas Dinámicas?	42
3.2.1.	Ejemplo de aplicación.....	44
3.3.	Tipos de Informes Dinámicos	47
3.4.	Personalizando los cálculos	47
3.5.	Tipo de funciones de resumen de datos en tablas dinámicas.....	48
3.5.1.	Mostrando valores relativos.....	49
3.6.	Personalizando la apariencia de los informes	50
3.7.	Gráficos dinámicos y Datos segmentados.....	53
3.7.1.	Gráficos dinámicos.....	53
3.7.2.	Segmentación de datos	54
3.8.	Modelos de Datos.....	55
3.8.1.	Modelos de datos relacionales.....	56
3.9.	Herramientas “Power” en Excel	57
3.9.1.	Power Pivot.....	59
3.10.	Power Query (Obtener y Transformar desde Excel 2016).....	62
3.10.1.	Utilizando Power Query.....	62
3.11.	Power Maps (Mapas 3D desde Excel 2016).....	65
3.11.1.	Generar un Power Map	65
3.12.	Power BI. Más allá de Excel	69
3.12.1.	Utilizando Power BI Desktop	70
4.	Grabación de Macros y Programación VBA.....	73
4.1.	Introducción.....	73
4.2.	Grabación de macros en Excel con VBA	73
4.2.1.	Traduciendo nuestras acciones en código VBA: la grabadora de macros.....	74
4.2.2.	Grabando nuestra primera macro.....	74



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



4.2.3.	Ejemplo (Referencias Absolutas).....	76
4.2.4.	Examinando el Código de una macro	77
4.2.5.	Ejemplo (Referencias Relativas)	82
4.2.6.	Optimizando el código generado por la macro	83
4.2.7.	Ejemplo: Pegar Valores con atajo de teclado	85
4.3.	Personalizando los mensajes de error.....	87
4.3.1.	La función MSGBOX.....	87
4.4.	On Error Resume Next.....	90
4.5.	La instrucción Select CASE	91
4.6.	¿Qué es la programación orientada a objetos?	92
4.6.1.	Jerarquía	92
4.6.2.	Colecciones	93
4.6.3.	Propiedades	93
4.6.4.	Métodos.....	93
4.6.5.	Eventos	93
4.7.	Explorando los métodos y propiedades de un objeto.....	94
4.7.1.	Variables y Funciones en VBA.....	94
4.8.	El bucle For – Next	98
4.9.	Uso de la función Offset en VBA.....	99
4.10.	Funciones personalizadas en Excel – VBA	100
4.10.1.	Creando nuestras primeras funciones en VBA	100
4.10.2.	El bucle For Each – Next	101
4.10.3.	Funciones personalizadas que recalculen automáticamente	102
5.	Conclusiones.....	103
6.	Bibliografía	103



1. Introducción

Microsoft Excel © (en adelante Excel) es el software para la creación, gestión y edición de planillas de cálculo más conocido y utilizado a nivel mundial. Los orígenes de este programa datan de los primeros años de la década del 80.

Todas las personas que realizan tareas relacionadas con las ciencias económicas deberían dominar Excel; pero la realidad es que, aun siendo tan utilizado y tener tantos años de vigencia, existe mucho desconocimiento acerca de las características de este poderoso software.

Esperamos poder contribuir con este material teórico (basado en las versiones 2016 y 2019 de Excel) a generar la inquietud en los lectores sobre el estudio de Excel en particular y de la teoría de las planillas de cálculo en general.

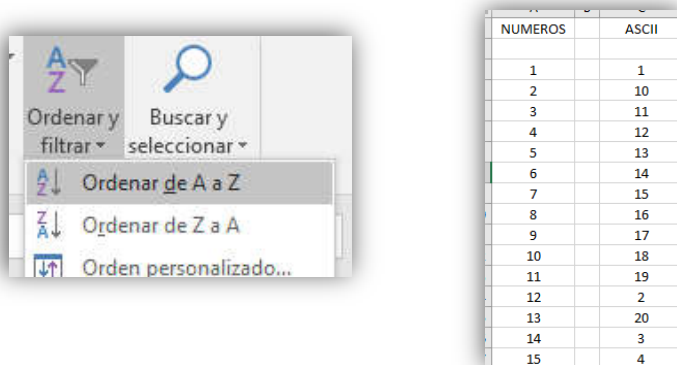
1.1. Tipos de datos en las planillas de cálculo

Excel puede gestionar tres tipos de datos, los cuales se definen como datos **NUMÉRICOS** (estos son los que pueden operarse matemáticamente), datos **ASCII** (datos que representan cadenas de texto) y datos **FÓRMULA** (instrucciones que contienen procedimientos operacionales integrados a Excel).

Más adelante, veremos que los datos pueden representarse visualmente al usuario con distintos formatos. **Lo más importante en este punto es que el usuario comprenda que, sin importar cómo se muestren en pantalla los datos, internamente Excel procesa tres tipos de datos: NUMÉRICOS, ASCII o FÓRMULA.**

Es importante destacar que los datos se ordenarán de distintas formas dependiendo los tipos de datos que contenga un determinado conjunto de celdas.

Observemos la siguiente representación gráfica de una pantalla de Excel en donde tenemos, por un lado, una serie de datos NUMÉRICOS y, por el otro, una serie de datos ASCII. A ambas series de datos se les aplico un orden con la opción:



Observemos ahora como, internamente, Excel diferencia un carácter numérico ASCII de un dato NUMÉRICO. Lo hace anteponiendo un apostrofe (') al dato.



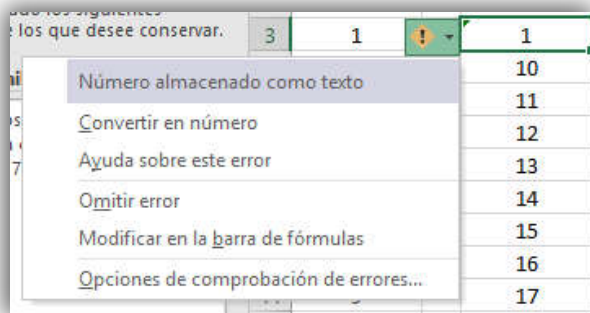
TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



A	B	C
NUMEROS		ASCII
1		1

Por defecto, desde las últimas versiones de Excel, al escribir un carácter numérico ASCII, se marcará la celda con un llamado, informando que Excel no tomará ese dato como un número y no podrá operar con la cantidad numérica que éste representa.



Por último, observemos que es lo que ocurre cuando queremos aplicar una fórmula a un conjunto de datos ASCII.

NUMEROS		ASCII	
1		1	
2		10	
3		11	
4		12	
5		13	
6		14	
21	=SUMA(A3:A8)	0	=SUMA(D3:D8)

Si nos preguntamos por qué ocurre esto, la respuesta es muy sencilla: al ser datos ASCII, Excel entiende que cada celda tiene un valor numérico de CERO.



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



1.2. Celdas, Rangos y Tablas

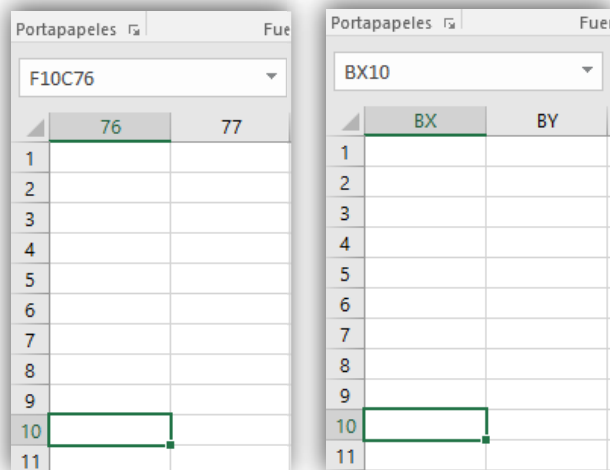
En este punto abordaremos la forma en los archivos de Excel estructura los datos que estos contienen.

Las planillas de Excel se componen por **LIBROS** (cada archivo de Excel se denomina Libro) con una determinada cantidad de **HOJAS** (desde la versión 2007 de Excel, el número de hojas posibles en un libro está limitado solamente por la memoria de la computadora) Estas **HOJAS** pueden tener un total de 1.048.576 filas por 16384 columnas.

La unidad mínima de información en Excel se denomina **CELDA** y, en principio, se identifica por su posición vertical y su posición horizontal, por ejemplo, la celda A1 es la que corresponde a la intersección de la primera fila con la primera columna, de un total de 17.179.869.184 posibles intersecciones distintas, es decir, unos 17 mil millones de celdas posibles.

Excel permite identificar las columnas por letras o por su posición numérica. Para ello se debe realizar los siguientes pasos **ARCHIVO** → **OPCIONES** → **FORMULAS** → **ESTILO DE REFERENCIA F1C1**.

En principio puede resultar algo confuso ya que la celda A1 se denominará F1C1 (Fila 1, Columna 1), pero en planillas grandes (con más de 30 columnas) resultará muy fácil identificar la posición vertical de cada celda. Observe estos casos...



El segundo tipo de agrupamiento de datos se da con los llamados **RANGOS**. Un rango en Excel es un conjunto de celdas contiguas en forma vertical, como en forma horizontal. Dos o más celdas contiguas forman un rango. Es importante aclarar que Excel no considera a los rangos como una estructura de información a menos que se les dé un nombre a los mismos.



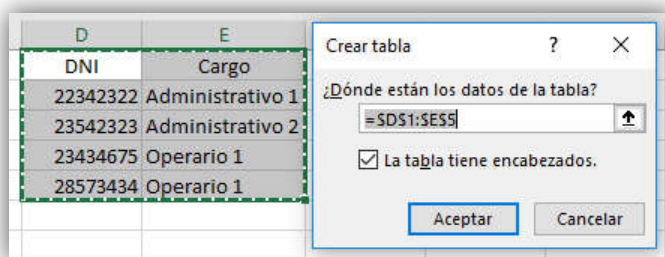
TÓPICOS DE EXCEL

(Material teórico gratuito 2019)

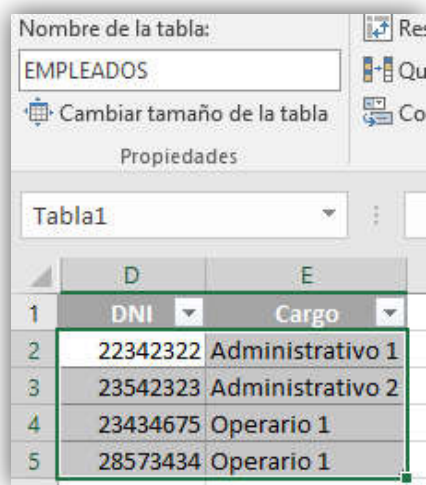


	DNI	Cargo
1		
2	22342322	Administrativo 1
3	23542323	Administrativo 2
4	23434675	Operario 1
5	28573434	Operario 1

El tercer agrupamiento de datos se da con las llamadas **TABLAS**. Una tabla en Excel es una estructura ampliada de información que permite una integración adecuada de fórmulas y formatos, además de filtros y ordenamientos entre los mismos.



Con la combinación de teclado **CTRL + t** se puede convertir un rango en tabla de una manera rápida y fácil. Al trabajar con tablas se consiguen una serie de ventajas en lo que respecta a velocidad y precisión en la carga de datos, así como también en lo referente a la integridad de la lista contenida en la tabla.





TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



Las ventajas de las tablas por sobre los rangos también se pueden apreciar en la construcción de fórmulas, ya que al ingresar una fórmula en una celda, la misma se replicará en la totalidad de la columna correspondiente.

	D	E	F	G	H
1	DNI	Cargo	Columna		
2	22342322	Administrativo 1	=IZQUIERDA([@Cargo];3)		
3	23542323	Administrativo 2	A IZQUIERDA(texto; [núm_de_caracteres])		
4	23434675	Operario 1	Ope		
5	28573434	Operario 1	Ope		
6					

1.3. Los formatos y sus propiedades

Los llamados formatos en Excel permiten “enmascarar” los datos, presentándolos de diferentes maneras.

Es importante destacar que los formatos funcionan exclusivamente sobre una capa visual, ya que, como hemos explicado anteriormente, Excel solamente almacena datos NUMERICOS, ASCII o FORMULAS.

	A	B	C	D
1	DATO	FORMATO HORA	FORMATO PORCENTAJE	FORMATO CIENTIFICO
2	0,9123232	21:53:45	91%	9,12E-01
3				

El caso más destacable, tal vez, es el de las fechas y las horas. Para las fechas, Excel considera internamente una fecha como cantidad de días que pasaron desde el 1ro de enero de 1900. Para las horas, Excel divide un día considerando al valor 0 como las 00.00 hs, al valor 0,5 como las 12:00 hs y al valor 1 como las 00.00 hs del día siguiente.

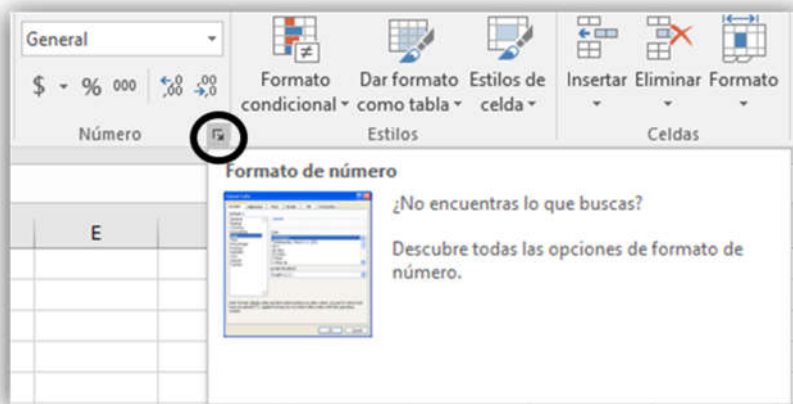
	A	B
1	DATO	FECHA
2	1	domingo, 1 de enero de 1900
3	2	lunes, 2 de enero de 1900
4	42340	miércoles, 2 de diciembre de 2015
5		
6	DATO	HORA
7	0,0001	00:00:09
8	0,5	12:00:00
9	0,99999	23:59:59
10		

La opción de **FORMATO DE NÚMERO** permite un gran nivel de personalización en la “máscara visual” que se utilizará para mostrar los datos contenidos en una celda.



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)

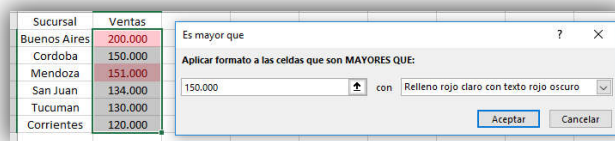
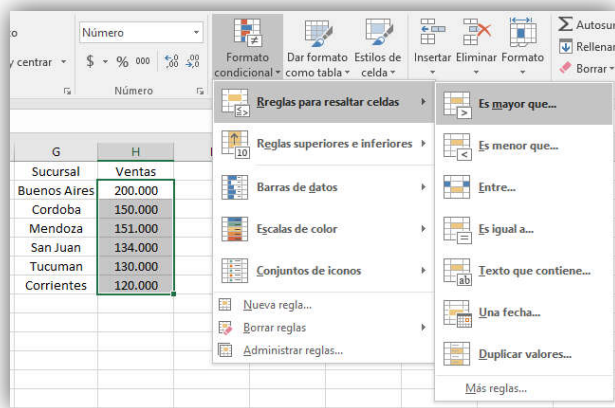


1.4. Formatos condicionales

La herramienta **FORMATO CONDICIONAL** permite aplicar distintos tipos de formatos en las celdas según cumplan éstas ciertas condiciones específicas, como valores duplicados, valores que cumplen criterios específicos (como mayor que 100 o es igual a "Ingresos" con Resaltar reglas de celdas y Reglas superiores e inferiores).

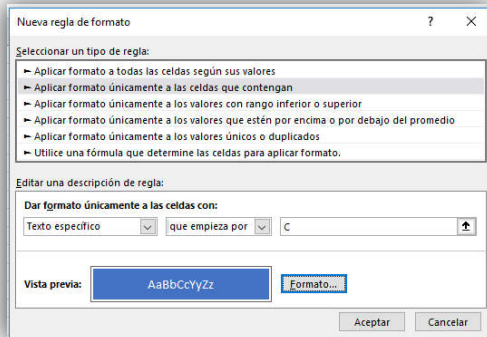
El formato condicional es dinámico y, por lo tanto, a medida que cambien los valores, el formato se ajustará automáticamente.

En el ejemplo tenemos una lista de sucursales y a cuanto ascendieron sus ventas anuales. Queremos resaltar aquellas sucursales cuya venta fue mayor a \$150.000. Para ello utilizaremos la opción de formato condicional.





Podemos, además, dar formato a celdas que respondan a una determinada característica no numérica, por ejemplo, las sucursales que empiecen con la letra C. Para ello vamos a la opción **más reglas**.



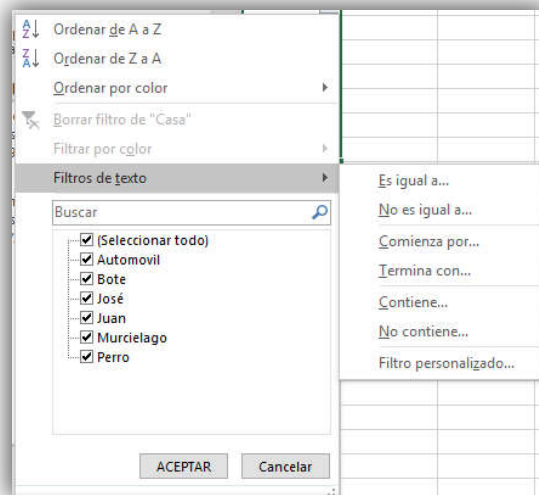
Sucursal	Ventas
Buenos Aires	200.000
Cordoba	150.000
Mendoza	151.000
San Juan	134.000
Tucuman	130.000
Corrientes	120.000

1.5. Filtros

Cuando se tiene una gran cantidad de información en una planilla, es importante el disponer de herramientas que permitan filtrar determinados datos. Excel permite aplicar distintos tipos de filtros en función al tipo de datos con los que se estén trabajando.

1.5.1. Filtros de texto

Los filtros de texto permiten trabajar con las cadenas de caracteres ASCII que componen los datos en las celdas. Podremos, por ejemplo, filtrar todas las palabras que NO contengan el carácter "J", así como también las que comiencen o terminen con una determinada cadena de caracteres, entre otras opciones.





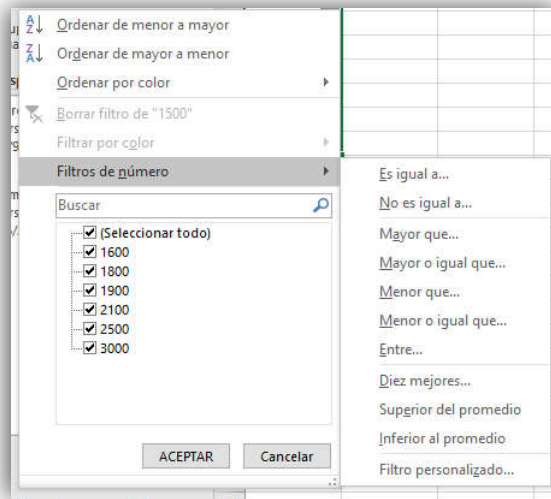
TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



1.5.2. Filtros de Números

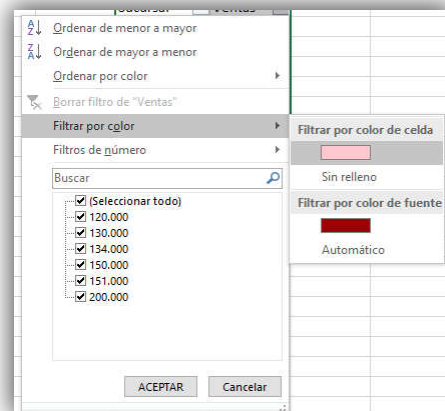
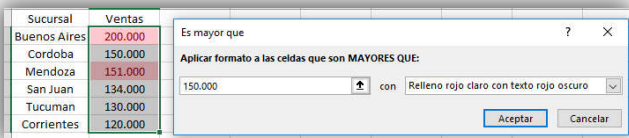
Los filtros de números permiten trabajar con los valores numéricos contenidos en las distintas celdas de un rango o de una tabla. Excel ofrece una gran cantidad de filtros para trabajar con nuestros datos numéricos.



1.5.3. Filtros por color

Excel también permite aplicarles filtros a nuestras celdas dependiendo si están rellenas con un determinado color en particular o si la fuente tiene una determinada característica de color.

La opción de filtros por color es una excelente herramienta para combinar con la opción formatos condicionales. En el ejemplo, sobre el listado de sucursales y su monto anual de ventas, aplicamos un formato condicional para que nos muestre los valores mayores a 150.000, para luego aplicarle un filtro por color de celdas. De esa manera, Excel mostrará solamente las celdas que cumplan con dicha característica.





TÓPICOS DE EXCEL

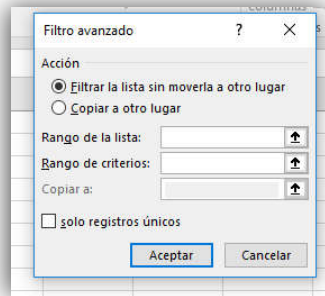
(Material teórico gratuito 2019)



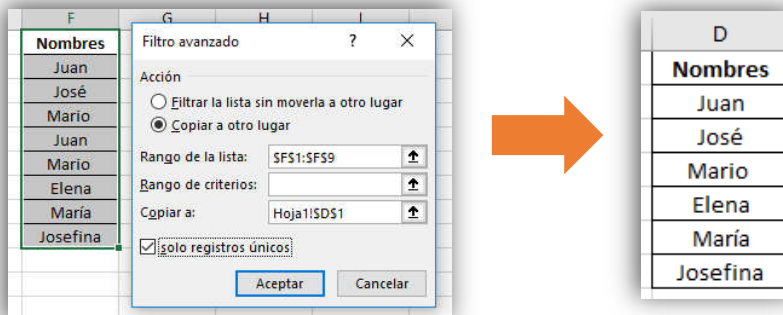
Al combinar las herramientas de formato condicional con la de filtro por color, podremos reducir la cantidad de datos mostrados por la planilla, enfocando nuestra atención en la información que realmente nos interesa.

1.6. Filtros Avanzados

Entre las herramientas de Excel, dentro de la pestaña datos, se encuentra la opción filtros avanzados. Con filtros avanzados se pueden realizar interesantes manipulaciones de información en las planillas de cálculo.



Entre las aplicaciones de Filtros avanzados es posible, por ejemplo, copiar una serie de datos sin valores duplicados a otro lugar de la planilla (utilizando la opción Sólo registros únicos)



El funcionamiento de la opción **Rangos de criterios** es bastante sencillo, si se entiende su lógica.

B	C	D	E	F	G	H
Sucursal	Local	Ventas		Sucursal	Local	Ventas
Buenos Aires	Grande	200.000		Buenos Aires		
Mendoza	Grande	170.000			Mediano	
Córdoba	Mediano	120.000				<95000
Corrientes	Mediano	110.000				
Entre Ríos	Chico	90.000				
Tucumán	Chico	95.000				
San Juan	Mediano	125.000				
San Luis	Grande	190.000				
Río Negro	Chico	95.000				
Chubut	Mediano	100.000				

Diagram illustrating the 'Rangos de criterios' option. It shows a table with columns B (Sucursal), C (Local), and D (Ventas). A separate table shows the result of applying the 'Rangos de criterios' option, with columns F (Sucursal), G (Local), and H (Ventas). The result table shows rows for Buenos Aires (Local), Mediano (Local), and <95000 (Ventas). The diagram labels the result as 'Condición "Y"' and 'Condición "O"'.



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



En este caso se pide a Excel que realice un filtro a la lista utilizando la siguiente instrucción lógica; a saber: Filtrar la lista Mostrando los valores (dentro de la columna Sucursal) que sean iguales a “Buenos Aires” o los que (dentro de la columna Local) sean igual a “Mediano” o los que (dentro de la columna Ventas) sean inferior a 95000.

Sucursal	Local	Ventas
Buenos Aires	Grande	200.000
Córdoba	Mediano	120.000
Corrientes	Mediano	110.000
Entre Ríos	Chico	90.000
San Juan	Mediano	125.000
Chubut	Mediano	100.000

De una forma similar se aplica otro criterio de filtro. En este caso se precisa la información de las sucursales en donde se hayan vendidos montos de más de \$100.000 y menos de \$200.000. El rango de criterios quedaría de la siguiente manera.

B	C	D	E	F	G
Sucursal	Local	Ventas		Ventas	Ventas
Buenos Aires	Grande	200.000		>100000	<200000
Mendoza	Grande	170.000			
Córdoba	Mediano	120.000			
Corrientes	Mediano	110.000			
Entre Ríos	Chico	90.000			
Tucumán	Chico	95.000			
San Juan	Mediano	125.000			
San Luis	Grande	190.000			
Río Negro	Chico	95.000			
Chubut	Mediano	100.000			

Se observa que se mantiene la lógica de que las condiciones “Y” se agregan en forma horizontal, en este caso se agregó una nueva columna Ventas.

Sucursal	Local	Ventas
Mendoza	Grande	170.000
Córdoba	Mediano	120.000
Corrientes	Mediano	110.000
San Juan	Mediano	125.000
San Luis	Grande	190.000



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)

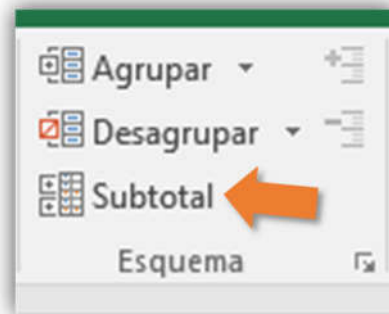


La opción de Filtro avanzado resulta extremadamente útil en procesos de automatización de tareas mediante el empleo de Macros.

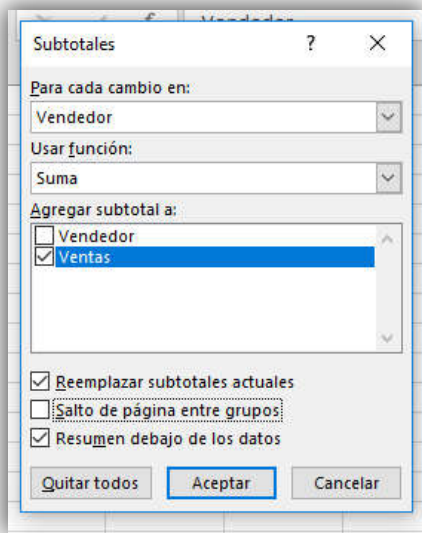
1.7. Subtotales

Cuando se tienen listas en donde la información de ciertas celdas se refiere al mismo ítem, es posible calcular subtotales de manera sencilla con la opción **SUBTOTAL**, dentro de la pestaña **DATOS**.

	A	B
1	Vendedor	Ventas
2	Sanchez	200.000
3	Sanchez	190.000
4	Sanchez	180.000
5	Gomez	210.000
6	Gomez	195.000
7	Gomez	185.000
8	Alvarez	230.000
9	Alvarez	240.000
10	Alvarez	190.000



La opción **SUBTOTAL**, permite insertar subtotales y totales por cada cambio de ítem, para ello es importante que la columna que se utilizará como parámetro esté ordenada. La ventana de trabajo para esta herramienta es la siguiente:



1	2	3	A	B
			Vendedor	Ventas
			Sanchez	200.000
			Sanchez	190.000
			Sanchez	180.000
			Total Sanchez	570.000
			Gomez	210.000
			Gomez	195.000
			Gomez	185.000
			Total Gomez	590.000
			Alvarez	230.000
			Alvarez	240.000
			Alvarez	190.000
			Total Alvarez	660.000
			Total general	1.820.000

Las guías deslizables del costado izquierdo son útiles para poder agrupar la información, activándolas, se mostrarán los totales correspondientes a cada ítem.



2. ¿Qué son las funciones en Excel?

Las funciones de Excel son procedimientos (algoritmos) predefinidos, que permiten realizar diferentes tipos de cálculo.

Las funciones se alimentan de valores específicos, denominados parámetros o argumentos, que el usuario debe introducir en un orden particular.

2.1. ¿Cómo invocar a una función?

Antes de responder a dicha pregunta, es necesario aclarar que las últimas versiones de Excel traen incorporadas más de 400 funciones, con lo cual resulta bastante dificultoso para la mayoría de los usuarios el memorizar los nombres y la ubicación de los distintos argumentos de éstas en su construcción.

Excel, en su interfaz, posee diversos métodos para invocar a las funciones; algunos de éstos son:

- 1) Escribir el nombre de la función (luego de los símbolos “=”, “+” y “@”).
Esta metodología es considerada la más avanzada, ya que supone que el usuario conoce el orden de los argumentos de la función utilizada.

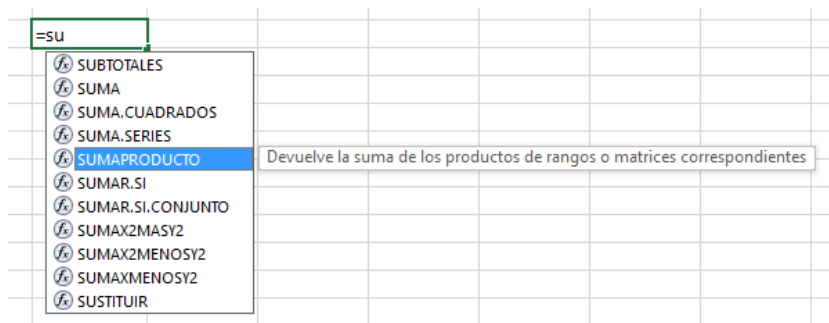


Figura 2.1. Al escribir, luego del símbolo “=”, las primeras letras del nombre de una función, Excel desplegará un cuadro para seleccionar la función requerida. Al recorrer dichos nombres se mostrará un pequeño resumen de la aplicación de la función correspondiente.



Figura 2.2. El usuario debe introducir los argumentos correspondientes a la función elegida. El separador de argumentos utilizado se mostrará en la ayuda que aparece debajo de la función. (En este caso, el separador de argumentos es el punto y coma “;”)



2) Utilizando el ícono “Insertar función” junto a la barra de fórmulas.

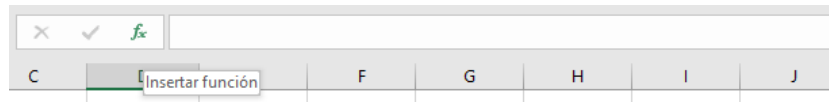


Figura 2.3. El ícono “Insertar Función” facilita el trabajo con funciones en Excel

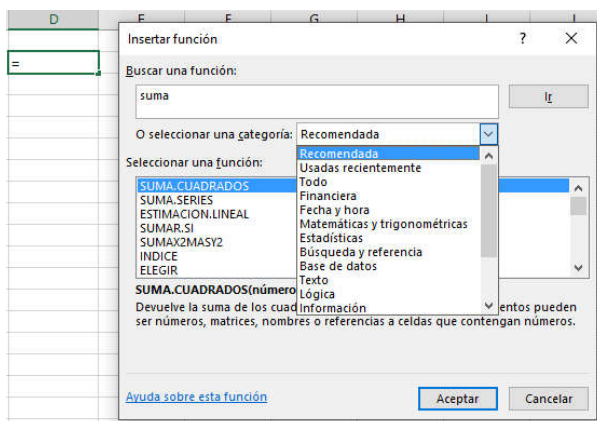


Figura 2.4. La ventana “Insertar Función” nos permitirá buscarla por nombre o categoría, también recordará las que utilizamos frecuentemente. En esta ventana, además podemos acceder al panel de ayuda de cada función.

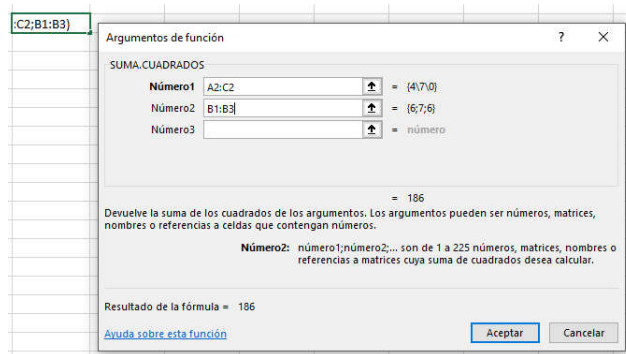


Figura 2.5. La ventana “Argumentos de función” nos brindará una interfaz clara para el ingreso de los argumentos de la función elegida. También nos mostrará el resultado que arrojará la función. Esta ventana resulta de mucha utilidad cuando trabajamos con funciones con gran cantidad de parámetros.



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



3) Desde la “Biblioteca de Funciones”.

En la pestaña Fórmulas, tendremos a las funciones agrupadas en categorías, este método es similar al anterior, en tanto y en cuanto nos mostrará una interfaz para el ingreso de los argumentos.

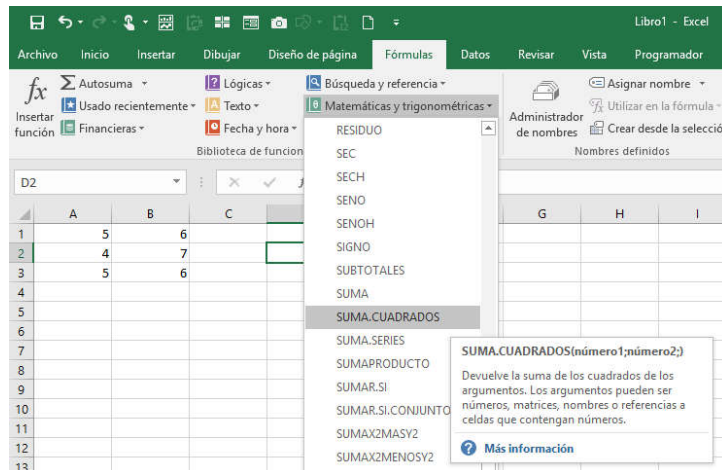


Figura 2.6. La opción “Biblioteca de funciones” agrupa a las funciones dependiendo su categoría. También permite acceder a la ayuda de Excel respectiva a cada función.

2.2. Tipos de funciones

Microsoft Excel 2016 cuenta con más de 400 funciones incorporadas que se utilizan para resolver diversos tipos de operaciones, a partir de los datos ingresados en la planilla de cálculo.

Los datos utilizados para trabajar con funciones pueden ser tipo numérico, texto, fecha/hora, así como también otras funciones encadenadas en fórmulas. Las funciones permiten realizar desde cálculos sencillos hasta cálculos complejos como ecuaciones científicas, estadísticas y financieras.

Las funciones más utilizadas en Excel se clasifican de la siguiente manera:

2.2.1. Funciones de Búsqueda y Referencia

Las funciones de búsqueda y referencia en Excel nos permiten encontrar valores dentro de nuestra planilla de cálculo de acuerdo con los criterios establecidos en la búsqueda. También nos ayudan a obtener información de referencia de determinadas celdas.

Una de las mayores utilidades de este tipo de funciones es la de realizar comparaciones entre pares de datos y devolver un resultado correspondiente.



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



FUNCIÓN	INGLÉS	DESCRIPCIÓN
AREAS	AREAS	Devuelve el número de áreas de una referencia. Un área es un rango de celdas contiguas o una única celda.
BUSCAR	LOOKUP	Busca valores de un rango de una columna o una fila o desde una matriz.
BUSCARH	HLOOKUP	Busca en la primera fila de una tabla o matriz de valores y devuelve el valor en la misma columna desde una fila especificada.
BUSCARV	VLOOKUP	Busca un valor en la primera columna de la izquierda de una tabla y luego devuelve un valor en la misma fila desde una columna especificada. De forma predeterminada, la tabla se ordena de forma ascendente.
COINCIDIR	MATCH	Devuelve la posición relativa de un elemento en una matriz, que coincide con un valor dado en un orden especificado.
COLUMNA	COLUMN	Devuelve el número de columna de una referencia.
COLUMNAS	COLUMNS	Devuelve el número de columnas en una matriz o referencia.
DESREF	OFFSET	Devuelve una referencia a un rango que es un número especificado de filas y columnas de una referencia dada.
DIRECCION	ADDRESS	Crea una referencia de celda en forma de texto una vez especificados los números de fila y columna.
ELEGIR	CHOOSE	Elige un valor o una acción de una lista de valores a partir de un número de índice.
FILA	ROW	Devuelve el número de fila de una referencia.
FILAS	ROWS	Devuelve el número de filas de una referencia o matriz.
HIPERVINCULO	HYPERLINK	Crea un acceso directo o salto que abre un documento guardado en el disco duro, en un servidor de red o en Internet.
IMPORTARDATOSDINAMICOS	GETPIVOTDATA	Extrae datos almacenados en una tabla dinámica.
INDICE	INDEX	Devuelve un valor o referencia de la celda en la intersección de una fila y columna en particular, en un rango especificado.
INDIRECTO	INDIRECT	Devuelve una referencia especificada por un valor de texto.
RDTR	RTD	Recupera datos en tiempo real de un programa compatible con automatizaciones COM.
TRANSPONER	TRANSPOSE	Devuelve un rango vertical de celdas como un rango horizontal, o viceversa.

2.2.2. Funciones de Fecha y Hora

Las funciones de fecha y hora de Microsoft Excel son de gran utilidad para los siguientes casos: buscar fechas específicas, conocer la hora actual, encontrar la diferencia en días laborales entre dos fechas además de facilitar una gran cantidad de acciones que involucren el trabajo con este tipo de datos.

Las funciones de fecha y hora deberían de las más conocidas por los profesionales en Ciencias Económicas, debido a su potencial en el trabajo con datos que involucren el paso del tiempo.



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



FUNCIÓN	INGLÉS	DESCRIPCIÓN
AHORA	NOW	Devuelve la fecha y hora actuales con formato de fecha y hora.
AÑO	YEAR	Devuelve el año, un número entero en el rango 1900-9999.
DIA	DAY	Devuelve el día del mes (un número de 1 a 31).
DIA.LAB	WORKDAY	Devuelve el número de serie de la fecha antes o después de un número especificado de días laborables.
DIA.LAB.INTL	WORKDAY.INTL	Devuelve el número de serie de la fecha anterior o posterior a un número especificado de días laborables con parámetros de fin de semana personalizados.
DIAS.LAB	NETWORKDAYS	Devuelve el número total de días laborables entre dos fechas.
DIAS.LAB.INTL	NETWORKDAYS.INTL	Devuelve el número de días laborables completos entre dos fechas con parámetros de fin de semana personalizados.
DIAS360	DAYS360	Calcula el número de días entre dos fechas basándose en un año de 360 días (doce meses de 30 días).
DIASEM	WEEKDAY	Devuelve un número de 1 a 7 que identifica el día de la semana.
FECHA	DATE	Devuelve el número que representa la fecha en código de fecha y hora de Microsoft Excel.
FECHA.MES	EDATE	Devuelve el número de serie de la fecha que es el número indicado de meses antes o después de la fecha inicial.
FECHANUMERO	DATEVALUE	Convierte una fecha en forma de texto en un número que representa la fecha en código de fecha y hora de Microsoft Excel.
FIN.MES	EOMONTH	Devuelve el número de serie del último día del mes antes o después del número especificado de meses.
FRAC.AÑO	YEARFRAC	Devuelve la fracción del año que representa el número de días completos entre la fecha_inicial y la fecha_fin.
HORA	HOUR	Devuelve la hora como un número de 0 (12:00 a.m.) a 23 (11:00 p.m.).
HORANUMERO	TIMEVALUE	Convierte una hora de texto en un número de serie de Excel para una hora, un número de 0 (12:00:00 a.m.) a 0.999988426 (11:59:59 p.m.). Da formato al número con un formato de hora después de introducir la fórmula.
HOY	TODAY	Devuelve la fecha actual con formato de fecha.
MES	MONTH	Devuelve el mes, un número entero de 1 (enero) a 12 (diciembre).
MINUTO	MINUTE	Devuelve el minuto, un número de 0 a 59.
NSHORA	TIME	Convierte horas, minutos y segundos dados como números en un número de serie de Excel, con formato de hora.
NUM.DE.SEMANA	WEEKNUM	Devuelve el número de semanas en el año.
SEGUNDO	SECOND	Devuelve el segundo, un número de 0 a 59.



TÓPICOS DE EXCEL (Material teórico gratuito 2019)



2.2.3. Funciones de Texto

Las funciones de texto en Excel permiten concatenar cadenas de caracteres, remover los espacios en blanco, contar el número de caracteres de una cadena de texto, así como también reemplazar ciertos caracteres por otros. Este tipo de funciones es ideal para organizar y formatear correctamente nuestra información no numérica contenida en nuestra planilla de cálculo.

FUNCIÓN	INGLÉS	DESCRIPCIÓN
CARACTER	CHAR	Devuelve el carácter especificado por el número de código a partir del juego de caracteres establecido en su PC.
CODIGO	CODE	Devuelve el número de código del primer carácter del texto del juego de caracteres usados por su PC.
CONCATENAR	CONCATENATE	Une varios elementos de texto en uno solo.
DECIMAL	FIXED	Redondea un número al número especificado de decimales y devuelve el resultado como texto con o sin comas.
DERECHA	RIGHT	Devuelve el número especificado de caracteres del final de una cadena de texto.
ENCONTRAR	FIND	Devuelve la posición inicial de una cadena de texto dentro de otra cadena de texto.
ESPACIOS	TRIM	Quita todos los espacios del texto excepto los espacios individuales entre palabras.
EXTRAE	MID	Devuelve los caracteres del centro de una cadena de texto, dada una posición y longitud iniciales.
HALLAR	SEARCH	Devuelve el número de caracteres en el cual se encuentra un carácter en particular o cadena de texto, leyendo de izquierda a derecha.
IGUAL	EXACT	Comprueba si dos cadenas de texto son exactamente iguales y devuelve VERDADERO o FALSO.
IZQUIERDA	LEFT	Devuelve el número especificado de caracteres del principio de una cadena de texto.
LARGO	LEN	Devuelve el número de caracteres de una cadena de texto.
LIMPIAR	CLEAN	Quita todos los caracteres no imprimibles del texto.
MAYUSC	UPPER	Convierte una cadena de texto en letras mayúsculas.
MINUSC	LOWER	Convierte todas las letras de una cadena de texto en minúsculas.
MONEDA	DOLLAR	Convierte un número en texto usando formato de moneda.
NOMPROPIO	PROPER	Convierte una cadena de texto en mayúsculas o minúsculas, según corresponda; la primera letra de cada palabra en mayúscula y las demás letras en minúscula.
REEMPLAZAR	REPLACE	Reemplaza parte de una cadena de texto por otra.
REPETIR	REPT	Repite el texto un número determinado de veces.
SUSTITUIR	SUBSTITUTE	Reemplaza el texto existente con texto nuevo en una cadena.
T	T	Comprueba si un valor es texto y devuelve el texto si lo es, o comillas dobles si no lo es.



TEXTO	<i>TEXT</i>	Convierte un valor en texto, con un formato de número específico.
TEXTobaht	<i>BAHTTEXT</i>	Convierte un número en texto (baht).
VALOR	<i>VALUE</i>	Convierte un argumento de texto que representa un número en un número.

2.2.4. Funciones Estadísticas

Este tipo de funciones nos permite disponer de una gran cantidad de herramientas aplicables al campo de la Estadística Descriptiva para resumir datos, dependiendo de la característica deseada. Además, nos facilita un conjunto de herramientas de Estadística Inferencial, así como las principales distribuciones teóricas de probabilidad, para la resolución de los problemas directos e inversos.

FUNCIÓN	INGLÉS	DESCRIPCIÓN
COEF.DE.CORREL	<i>CORREL</i>	Devuelve el coeficiente de correlación de dos conjuntos de datos.
COEFICIENTE.ASIMETRIA	<i>SKEW</i>	Devuelve el sesgo de una distribución: una caracterización del grado de asimetría de una distribución alrededor de su media.
COEFICIENTE.R2	<i>RSQ</i>	Devuelve el cuadrado del coeficiente del momento de correlación del producto Pearson de los puntos dados.
CONTAR	<i>COUNT</i>	Cuenta el número de celdas de un rango que contienen números.
CONTAR.BLANCO	<i>COUNTBLANK</i>	Cuenta el número de celdas en blanco dentro de un rango especificado.
CONTAR.SI	<i>COUNTIF</i>	Cuenta las celdas en el rango que coinciden con la condición dada.
CONTAR.SI.CONJUNTO	<i>COUNTIFS</i>	Cuenta el número de celdas que cumplen un determinado conjunto de condiciones o criterios.
CONTARA	<i>COUNTA</i>	Cuenta el número de celdas no vacías de un rango.
COVARIANCE.P	<i>COVARIANCE.P</i>	Devuelve la covarianza de población, el promedio de los productos de las desviaciones para cada pareja de puntos de datos en dos conjuntos de datos.
COVARIANZA.M	<i>COVARIANCE.S</i>	Devuelve la covarianza, el promedio de los productos de las desviaciones para cada pareja de puntos de datos en dos conjuntos de datos.
CRECIMIENTO	<i>GROWTH</i>	Devuelve números en una tendencia de crecimiento exponencial coincidente con puntos de datos conocidos.
CUARTIL.EXC	<i>QUARTILE.EXC</i>	Devuelve el cuartil de un conjunto de datos en función de los valores del percentil de 0..1, exclusivo.
CUARTIL.INC	<i>QUARTILE.INC</i>	Devuelve el cuartil de un conjunto de datos en función de los valores del percentil de 0..1, inclusive.
CURTOSIS	<i>KURT</i>	Devuelve la curtosis de un conjunto de datos.



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



DESVEST.M	<i>STDEV.S</i>	Calcula la desviación estándar en función de una muestra (omite los valores lógicos y el texto).
DESVEST.P	<i>STDEV.P</i>	Calcula la desviación estándar en función de la población total proporcionada como argumentos (omite los valores lógicos y el texto).
DESVESTA	<i>STDEVA</i>	Calcula la desviación estándar de una muestra, incluyendo valores lógicos y texto. Los valores lógicos y el texto con valor FALSO tienen valor asignado 0, los que presentan valor VERDADERO tienen valor 1.
DESVESTPA	<i>STDEVPA</i>	Calcula la desviación estándar de la población total, incluyendo valores lógicos y el texto. Los valores lógicos y el texto con valor FALSO tienen valor asignado 0, los que presentan un valor VERDADERO tienen valor 1.
DESVIA2	<i>DEVSQ</i>	Devuelve la suma de los cuadrados de las desviaciones de los puntos de datos con respecto al promedio de la muestra.
DESVPROM	<i>AVEDEV</i>	Devuelve el promedio de las desviaciones absolutas de la media de los puntos de datos.
DISTR.BETA.N	<i>BETA.DIST</i>	Devuelve la función de distribución de probabilidad beta.
DISTR.BINOM.N	<i>BINOM.DIST</i>	Devuelve la probabilidad de una variable aleatoria discreta siguiendo una distribución binomial.
DISTR.CHICUAD	<i>CHISQ.DIST</i>	Devuelve la probabilidad de cola izquierda de la distribución chi cuadrado.
DISTR.CHICUAD.CD	<i>CHISQ.DIST.RT</i>	Devuelve la probabilidad de cola derecha de la distribución chi cuadrado.
DISTR.EXP.N	<i>EXPON.DIST</i>	Devuelve la distribución exponencial.
DISTR.F.CD	<i>F.DIST.RT</i>	Devuelve la distribución (de cola derecha) de probabilidad F (grado de diversidad) para dos conjuntos de datos.
DISTR.F.N	<i>F.DIST</i>	Devuelve la distribución (de cola izquierda) de probabilidad F (grado de diversidad) para dos conjuntos de datos.
DISTR.GAMMA.N	<i>GAMMA.DIST</i>	Devuelve la distribución gamma.
DISTR.HIPERGEOM.N	<i>HYPGEOM.DIST</i>	Devuelve la distribución hipergeométrica.
DISTR.LOGNORM	<i>LOGNORM.DIST</i>	Devuelve la distribución logarítmico-normal de x, donde ln(x) se distribuye normalmente con los parámetros de media y desv_ estándar.
DISTR.NORM.ESTAND.N	<i>NORM.S.DIST</i>	Devuelve la distribución normal estándar (tiene una medida de cero y una desviación estándar de uno).
DISTR.NORM.N	<i>NORM.DIST</i>	Devuelve la distribución normal para la media y la desviación estándar especificadas.
DISTR.T.2C	<i>T.DIST.2T</i>	Devuelve la distribución t de Student de dos colas.
DISTR.T.CD	<i>T.DIST.RT</i>	Devuelve la distribución t de Student de cola derecha.



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



DISTR.T.N	<i>T.DIST</i>	Devuelve la distribución t de Student de cola izquierda.
DISTR.WEIBULL	<i>WEIBULL.DIST</i>	Devuelve la probabilidad de una variable aleatoria siguiendo una distribución de Weibull.
ERROR.TIPICO.XY	<i>STEYX</i>	Devuelve el error típico del valor de Y previsto para cada X de la regresión.
ESTIMACION.LINEAL	<i>LINEST</i>	Devuelve estadísticas que describen una tendencia lineal que coincide con puntos de datos conocidos, mediante una línea recta usando el método de los mínimos cuadrados.
ESTIMACION.LOGARITMICA	<i>LOGEST</i>	Devuelve estadísticas que describen una curva exponencial, coincidente con puntos de datos conocidos.
FISHER	<i>FISHER</i>	Devuelve la transformación Fisher o coeficiente Z.
FRECUENCIA	<i>FREQUENCY</i>	Calcula la frecuencia con la que ocurre un valor dentro de un rango de valores y devuelve una matriz vertical de números.
GAMMA.LN	<i>GAMMALN</i>	Devuelve el logaritmo natural de la función gamma, G(x).
GAMMA.LN.EXACTO	<i>GAMMALN.PRECISE</i>	Devuelve el logaritmo natural de la función gamma, G(x).
INTERSECCION.EJE	<i>INTERCEPT</i>	Calcula el punto en el cual una línea intersectará el eje Y usando una línea de regresión optimizada trazada a través de los valores conocidos de X e Y.
INTERVALO.CONFIANZA.NORM	<i>CONFIDENCE.NORM</i>	Devuelve el intervalo de confianza para una media de población con una distribución normal.
INTERVALO.CONFIANZA.T	<i>CONFIDENCE.T</i>	Devuelve el intervalo de confianza para una media de población con una distribución de T de Student.
INV.BETA.N	<i>BETA.INV</i>	Devuelve el inverso de la función de densidad de probabilidad beta acumulativa (DISTR.BETA.N).
INV.BINOM	<i>BINOM.INV</i>	Devuelve el menor valor cuya distribución binomial acumulativa es mayor o igual que un valor de criterio.
INV.CHICUAD	<i>CHISQ.INV</i>	Devuelve el inverso de la probabilidad de cola izquierda de la distribución chi cuadrado.
INV.CHICUAD.CD	<i>CHISQ.INV.RT</i>	Devuelve el inverso de la probabilidad de cola derecha de la distribución chi cuadrado.
INV.F	<i>F.INV</i>	Devuelve el inverso de la distribución de probabilidad F (de cola izquierda): si $p = \text{DISTR.F}(x, \dots)$, entonces $\text{INV.F}(p, \dots) = x$.
INV.F.CD	<i>F.INV.RT</i>	Devuelve el inverso de la distribución de probabilidad F (cola derecha): si $p = \text{DISTR.F.CD}(x, \dots)$, entonces $\text{INV.F.CD}(p, \dots) = x$.
INV.GAMMA	<i>GAMMA.INV</i>	Devuelve el inverso de la distribución gamma acumulativa: si $p = \text{DISTR.GAMMA.N}(x, \dots)$, entonces $\text{INV.GAMMA}(p, \dots) = x$.
INV.LOGNORM	<i>LOGNORM.INV</i>	Devuelve el inverso de la distribución logarítmico-normal de x, donde $\ln(x)$ se distribuye de forma normal con los parámetros Media y <i>desv_estándar</i> .
INV.NORM	<i>NORM.INV</i>	Devuelve el inverso de la distribución acumulativa normal para la media y desviación estándar especificadas.



TÓPICOS DE EXCEL (Material teórico gratuito 2019)



INV.NORM.ESTAND	<i>NORM.S.INV</i>	Devuelve el inverso de la distribución normal estándar acumulativa. Tiene una media de cero y una desviación estándar de uno.
INV.T	<i>T.INV</i>	Devuelve el inverso de cola izquierda de la distribución t de Student.
INV.T.2C	<i>T.INV.2T</i>	Devuelve el inverso de dos colas de la distribución t de Student.
JERARQUIA.EQV	<i>RANK.EQ</i>	Devuelve la jerarquía de un número dentro de una lista de números: su tamaño en relación con otros valores de la lista.
JERARQUIA.MEDIA	<i>RANK.AVG</i>	Devuelve la jerarquía de un número dentro de una lista de números: su tamaño en relación con otros valores de la lista; si más de un valor tiene la misma jerarquía, se devuelve el promedio de la jerarquía.
K.ESIMO.MAYOR	<i>LARGE</i>	Devuelve el valor k-ésimo mayor de un conjunto de datos. Por ejemplo, el trigésimo número más grande.
K.ESIMO.MENOR	<i>SMALL</i>	Devuelve el valor k-ésimo menor de un conjunto de datos. Por ejemplo, el trigésimo número menor.
MAX	<i>MAX</i>	Devuelve el valor máximo de una lista de valores. Omite los valores lógicos y texto.
MAXA	<i>MAXA</i>	Devuelve el valor máximo de un conjunto de valores. Incluye valores lógicos y texto.
MEDIA.ACOTADA	<i>TRIMMEAN</i>	Devuelve la media de la porción interior de un conjunto de valores de datos.
MEDIA.ARMO	<i>HARMEAN</i>	Devuelve la media armónica de un conjunto de números positivos: el recíproco de la media aritmética de los recíprocos.
MEDIA.GEOM	<i>GEOMEAN</i>	Devuelve la media geométrica de una matriz o rango de datos numéricos positivos.
MEDIANA	<i>MEDIAN</i>	Devuelve la mediana o el número central de un conjunto de números.
MIN	<i>MIN</i>	Devuelve el valor mínimo de una lista de valores. Omite los valores lógicos y texto.
MINA	<i>MINA</i>	Devuelve el valor mínimo de una lista de valores. Incluye valores lógicos y texto.
MODA.UNO	<i>MODE.SNGL</i>	Devuelve el valor más frecuente o repetitivo de una matriz o rango de datos.
MODA.VARIOS	<i>MODE.MULT</i>	Devuelve una matriz vertical de los valores más frecuentes o repetitivos de una matriz o rango de datos. Para una matriz horizontal, use =TRANSPONER(MODA.VARIOS(número1,número2,...)).
NEGBINOM.DIST	<i>NEGBINOM.DIST</i>	Devuelve la distribución binomial negativa, la probabilidad de encontrar <i>núm_fracasos</i> antes que <i>núm_éxito</i> , con probabilidad <i>probabilidad_s</i> de éxito.
NORMALIZACION	<i>STANDARDIZE</i>	Devuelve un valor normalizado de una distribución caracterizada por una media y desviación estándar.



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



PEARSON	<i>PEARSON</i>	Devuelve el coeficiente de correlación producto o momento r de Pearson, r.
PENDIENTE	<i>SLOPE</i>	Devuelve la pendiente de una línea de regresión lineal de los puntos dados.
PERCENTIL.EXC	<i>PERCENTILE.EXC</i>	Devuelve el percentil k-ésimo de los valores de un rango, donde k está en el rango 0..1, exclusivo.
PERCENTIL.INC	<i>PERCENTILE.INC</i>	Devuelve el percentil k-ésimo de los valores de un rango, donde k está en el rango 0,1, inclusive.
PERMUTACIONES	<i>PERMUT</i>	Devuelve el número de permutaciones para un número determinado de objetos que pueden ser seleccionados de los objetos totales.
POISSON.DIST	<i>POISSON.DIST</i>	Devuelve la distribución de Poisson.
PROBABILIDAD	<i>PROB</i>	Devuelve la probabilidad de que los valores de un rango se encuentren entre dos límites o sean iguales a un límite inferior.
PROMEDIO	<i>AVERAGE</i>	Devuelve el promedio (media aritmética) de los argumentos, los cuales pueden ser números, nombres, matrices o referencias que contengan números.
PROMEDIO.SI	<i>AVERAGEIF</i>	Busca el promedio (media aritmética) de las celdas que cumplen un determinado criterio o condición.
PROMEDIO.SI.CONJUNTO	<i>AVERAGEIFS</i>	Busca el promedio (media aritmética) de las celdas que cumplen un determinado conjunto de condiciones o criterios.
PROMEDIOA	<i>AVERAGEA</i>	Devuelve el promedio (media aritmética) de los argumentos; 0 evalúa el texto como FALSO; 1 como VERDADERO. Los argumentos pueden ser números, nombres, matrices o referencias.
PRONOSTICO	<i>FORECAST</i>	Calcula o predice un valor futuro en una tendencia lineal usando valores existentes.
PRUEBA.CHICUAD	<i>CHISQ.TEST</i>	Devuelve la prueba de independencia: el valor de distribución chi cuadrado para la estadística y los grados adecuados de libertad.
PRUEBA.F.N	<i>F.TEST</i>	Devuelve el resultado de una prueba F, la probabilidad de dos colas de que las varianzas en Matriz1 y Matriz2 no sean significativamente diferentes.
PRUEBA.FISHER.INV	<i>FISHERINV</i>	Devuelve la función inversa de la transformación Fisher o coeficiente Z: si $y = \text{FISHER}(x)$, entonces $\text{PRUEBA.FISHER.INV}(y) = x$.
PRUEBA.T.N	<i>T.TEST</i>	Devuelve la probabilidad asociada con la prueba t de Student.
PRUEBA.Z.N	<i>Z.TEST</i>	Devuelve el valor P de una cola de una prueba z.
RANGO.PERCENTIL.EXC	<i>PERCENTRANK.EXC</i>	Devuelve la jerarquía de un valor en un conjunto de datos como un porcentaje (0,1, exclusivo) del conjunto de datos.
RANGO.PERCENTIL.INC	<i>PERCENTRANK.INC</i>	Devuelve la jerarquía de un valor en un conjunto de datos como un porcentaje (0,1, inclusive) del conjunto de datos.
TENDENCIA	<i>TREND</i>	Devuelve números en una tendencia lineal que coincide con puntos de datos conocidos, usando el método de los mínimos cuadrados.



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



VAR.P	VAR.P	Calcula la varianza en función de la población total (omite los valores lógicos y el texto).
VAR.S	VAR.S	Calcula la varianza en función de una muestra (omite los valores lógicos y el texto).
VARA	VARA	Calcula la varianza de una muestra, incluyendo valores lógicos y texto. Los valores lógicos y el texto con valor FALSO tienen valor asignado 0, los de valor lógico VERDADERO tienen valor 1.
VARPA	VARPA	Calcula la varianza de la población total, incluyendo valores lógicos y texto. Los valores lógicos y el texto con valor FALSO tienen valor asignado 0, los de valor lógico VERDADERO tienen valor 1.

2.2.5. Funciones Financieras

Las funciones financieras facilitan la tarea del cálculo de casos relacionados con la administración financiera y los problemas referentes al valor tiempo del dinero.

Con este tipo de funciones podremos valorar activos, trasladando su valor a través de una línea del tiempo imaginaria, en operaciones financieras que involucren una o más prestaciones y contraprestaciones.

FUNCIÓN	INGLÉS	DESCRIPCIÓN
AMORTIZ.LIN	AMORLINC	Devuelve la amortización de cada uno de los períodos contables.
AMORTIZ.PROGRE	AMORDEGRC	Devuelve la amortización de cada período contable mediante el uso de un coeficiente de amortización.
CANTIDAD.RECIBIDA	RECEIVED	Devuelve la cantidad recibida al vencimiento para un valor bursátil completamente invertido.
CUPON.DIAS	COUPDAYS	Devuelve el número de días en el período nominal que contiene la fecha de liquidación.
CUPON.DIAS.L1	COUPDAYBS	Devuelve el número de días del inicio del período nominal hasta la fecha de liquidación.
CUPON.DIAS.L2	COUPDAYSNCR	Devuelve el número de días de la fecha de liquidación hasta la siguiente fecha nominal.
CUPON.FECHA.L1	COUPPCD	Devuelve la fecha de cupón anterior antes de la fecha de liquidación.
CUPON.FECHA.L2	COUPNCD	Devuelve la próxima fecha nominal después de la fecha de liquidación.
CUPON.NUM	COUPNUM	Devuelve el número de cupones pagables entre la fecha de liquidación y la fecha de vencimiento.
DB	DB	Devuelve la depreciación de un activo durante un período específico usando el método de depreciación de saldo fijo.
DDB	DDB	Devuelve la depreciación de un activo en un período específico mediante el método de depreciación por doble disminución de saldo u otro método que se especifique.
DURACION	DURATION	Devuelve la duración anual de un valor bursátil con pagos de interés periódicos.
DURACION.MODIF	MDURATION	Devuelve la duración modificada de Macauley para un valor bursátil con un valor nominal asumido de 100\$.
DVS	VDB	Devuelve la depreciación de un activo para cualquier período especificado, incluyendo períodos parciales, usando el método de depreciación por doble disminución del saldo u otro método que especifique.
INT.ACUM	ACCRINT	Devuelve el interés devengado de un valor bursátil que paga intereses periódicos.



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



INT.ACUM.V	<i>ACCRINTM</i>	Devuelve el interés devengado para un valor bursátil que paga intereses al vencimiento.
INT.EFECTIVO	<i>EFFECT</i>	Devuelve la tasa de interés anual efectiva.
INT.PAGO.DIR	<i>ISPMT</i>	Devuelve el interés de un préstamo de pagos directos.
LETRA.DE.TES.PRECIO	<i>TBILLPRICE</i>	Devuelve el precio de un valor nominal de 100\$ para una letra de tesorería.
LETRA.DE.TES.RENDTO	<i>TBILLYIELD</i>	Devuelve el rendimiento de una letra de tesorería.
LETRA.DE.TEST.EQV.A.BONO	<i>TBILLEQ</i>	Devuelve el rendimiento para un bono equivalente a una letra de tesorería.
MONEDA.DEC	<i>DOLLARDE</i>	Convierte un precio en dólar, expresado como fracción, en un precio en dólares, expresado como número decimal.
MONEDA.FRAC	<i>DOLLARFR</i>	Convierte un precio en dólar, expresado como número decimal, en un precio en dólares, expresado como una fracción.
NPER	<i>NPER</i>	Devuelve el número de pagos de una inversión, basado en pagos constantes y periódicos y una tasa de interés constante.
PAGO	<i>PMT</i>	Calcula el pago de un préstamo basado en pagos y tasa de interés constantes.
PAGO.INT.ENTRE	<i>CUMIPMT</i>	Devuelve el pago de intereses acumulativo entre dos períodos.
PAGO.PRINC.ENTRE	<i>CUMPRINC</i>	Devuelve el pago principal acumulativo de un préstamo entre dos períodos.
PAGOINT	<i>IPMT</i>	Devuelve el interés pagado por una inversión durante un período determinado, basado en pagos periódicos y constantes y una tasa de interés constante.
PAGOPRIN	<i>PPMT</i>	Devuelve el pago del capital de una inversión determinada, basado en pagos constantes y periódicos, y una tasa de interés constante.
PRECIO	<i>PRICE</i>	Devuelve el precio por 100\$ de valor nominal de un valor bursátil que paga una tasa de interés periódica.
PRECIO.DESCUENTO	<i>PRICEDISC</i>	Devuelve el precio por 100\$ de un valor nominal de un valor bursátil con descuento.
PRECIO.PER.IRREGULAR.1	<i>ODDFPRICE</i>	Devuelve el precio de un valor nominal de 100\$ de un valor bursátil con un período inicial impar.
PRECIO.PER.IRREGULAR.2	<i>ODDLPRICE</i>	Devuelve el precio de un valor nominal de 100\$ de un valor bursátil con un período final impar.
PRECIO.VENCIMIENTO	<i>PRICEMAT</i>	Devuelve el precio por 100\$ de un valor nominal que genera intereses al vencimiento.
RENDTO	<i>YIELD</i>	Devuelve el rendimiento de un valor bursátil que obtiene intereses periódicos.
RENDTO.DESC	<i>YIELDDISC</i>	Devuelve el rendimiento anual para el valor bursátil con descuento. Por ejemplo, una letra de la tesorería.
RENDTO.PER.IRREGULAR.1	<i>ODDFYIELD</i>	Devuelve el rendimiento de un valor bursátil con un primer período impar.
RENDTO.PER.IRREGULAR.2	<i>ODDLYIELD</i>	Devuelve la amortización de un valor bursátil con un período final impar.
RENDTO.VENCTO	<i>YIELDMAT</i>	Devuelve el interés anual de un valor que genera intereses al vencimiento.
SLN	<i>SLN</i>	Devuelve la depreciación por método directo de un activo en un período dado.
SYD	<i>SYD</i>	Devuelve la depreciación por método de anualidades de un activo durante un período específico.
TASA	<i>RATE</i>	Devuelve la tasa de interés por período de un préstamo o una inversión. Por ejemplo, use 6%/4 para pagos trimestrales al 6% TPA.
TASA.DESC	<i>DISC</i>	Devuelve la tasa de descuento del valor bursátil.
TASA.INT	<i>INTRATE</i>	Devuelve la tasa de interés para la inversión total en un valor bursátil.
TASA.NOMINAL	<i>NOMINAL</i>	Devuelve la tasa de interés nominal anual.
TIR	<i>IRR</i>	Devuelve la tasa interna de retorno de una inversión para una serie de valores en efectivo.
TIR.NO.PER	<i>XIRR</i>	Devuelve la tasa interna de retorno para un flujo de caja que no es necesariamente periódico.
TIRM	<i>MIRR</i>	Devuelve la tasa interna de retorno para una serie de flujos de efectivo periódicos, considerando costo de la inversión e interés al volver a invertir el efectivo.
VA	<i>PV</i>	Devuelve el valor presente de una inversión: la suma total del valor actual de una serie de pagos futuros.



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



VF	<i>FV</i>	Devuelve el valor futuro de una inversión basado en pagos periódicos y constantes, y una tasa de interés también constante.
VF.PLAN	<i>FVSCHEDULE</i>	Devuelve el valor futuro de una inversión inicial después de aplicar una serie de tasas de interés compuesto.
VNA	<i>NPV</i>	Devuelve el valor neto actual de una inversión a partir de una tasa de descuento y una serie de pagos futuros (valores negativos) y entradas (valores positivos).
VNA.NO.PER	<i>XNPV</i>	Devuelve el valor neto actual para un flujo de caja que no es necesariamente periódico.

2.2.6. Funciones Lógicas

Las funciones lógicas en Excel se utilizan en la toma de decisiones. En base al resultado de una función decidiremos si ejecutar o no cierta acción requerida.

Este tipo de funciones realiza una consulta respecto al valor de una celda, comparándola con una condición determinada, devolviendo el valor correspondiente al resultado de dicha comparación.

Utilizadas correctamente, se pueden confeccionar planillas muy dinámicas que nos ayudarán a tomar decisiones en función a los valores que se encuentren en éstas.

Sin duda la función lógica más utilizada es la función SI. La estructura de esta función es la siguiente:

=SI(PRUEBA LÓGICA;VALOR SI VERDADERO;VALOR SI FALSO)

Un sencillo ejemplo es el siguiente:

Si la celda A1 tiene un valor menor a 18 entonces se debe mostrar MENOR DE EDAD, en caso contrario (es decir, si la celda A1 tiene un valor igual o mayor a 18, entonces se debe mostrar MAYOR DE EDAD.

	A	B	C	D	E
1					
2		=SI(A1<18;"MENOR DE EDAD";"MAYOR DE EDAD")			
3					

	A	B
1	15	
2		MENOR DE EDAD

	A	B
1	19	
2		MAYOR DE EDAD

Veremos a continuación un listado con las funciones lógicas más conocidas en Excel.

FUNCIÓN	INGLÉS	DESCRIPCIÓN
FALSO	<i>FALSE</i>	Devuelve el valor lógico FALSO.
NO	<i>NOT</i>	Cambia FALSO por VERDADERO y VERDADERO por FALSO.
O	<i>OR</i>	Comprueba si alguno de los argumentos es VERDADERO y devuelve VERDADERO o FALSO. Devuelve FALSO si todos los argumentos son FALSO.
SI	<i>IF</i>	Comprueba si se cumple una condición y devuelve un valor si se evalúa como VERDADERO y otro valor si se evalúa como FALSO.
SI.ERROR	<i>IFERROR</i>	Devuelve un valor si la expresión es un error y otro valor si no lo es.



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



VERDADERO	TRUE	Devuelve el valor lógico VERDADERO.
Y	AND	Comprueba si todos los argumentos son VERDADEROS y devuelve VERDADERO o FALSO. Devuelve FALSO si alguno de los argumentos es FALSO.

2.2.7. Funciones Matemáticas y Trigonométricas

Son las funciones más útiles y utilizadas en Excel. Las Funciones Matemáticas y Trigonométricas incluyen gran cantidad de operaciones aritméticas, así como de redondeo de cifras.

FUNCIÓN	INGLÉS	DESCRIPCIÓN
ABS	ABS	Devuelve el valor absoluto de un número, es decir, un número sin signo.
ACOS	ACOS	Devuelve el arcoseno de un número, en radianes, dentro del intervalo de 0 a Pi.
ACOSH	ACOSH	Devuelve el coseno hiperbólico inverso de un número.
AGREGAR	AGGREGATE	Devuelve un agregado de una lista o base de datos.
ALEATORIO	RAND	Devuelve un número aleatorio mayor o igual que 0 y menor que 1.
ALEATORIO.ENTRE	RANDBETWEEN	Devuelve un número aleatorio entre los números que especifique.
ASENO	ASIN	Devuelve el arcoseno de un número en radianes, dentro del intervalo -Pi/2 a Pi/2
ASENOH	ASINH	Devuelve el seno hiperbólico inverso de un número.
ATAN	ATAN	Devuelve el arco tangente de un número en radianes, dentro del intervalo -Pi/2 a Pi/2.
ATAN2	ATAN2	Devuelve el arco tangente de las coordenadas X e Y especificadas, en un valor en radianes comprendido entre -Pi y Pi, excluyendo -Pi.
ATANH	ATANH	Devuelve la tangente hiperbólica inversa de un número.
COCIENTE	QUOTIENT	Devuelve la parte entera de una división.
COMBINAT	COMBIN	Devuelve el número de combinaciones para un número determinado de elementos.
COS	COS	Devuelve el coseno de un ángulo.
COSH	COSH	Devuelve el coseno hiperbólico de un número.
ENTERO	INT	Redondea un número hasta el entero inferior más próximo.
EXP	EXP	Devuelve e elevado a la potencia de un número determinado.
FACT	FACT	Devuelve el factorial de un número.
FACT.DOUBLE	FACTDOUBLE	Devuelve el factorial doble de un número.
GRADOS	DEGREES	Convierte radianes en grados.
LN	LN	Devuelve el logaritmo natural de un número.
LOG	LOG	Devuelve el logaritmo de un número en la base especificada.
LOG10	LOG10	Devuelve el logaritmo en base 10 de un número.
M.C.D	GCD	Devuelve el máximo común divisor.
M.C.M	LCM	Devuelve el mínimo común múltiplo.
MDETERM	MDETERM	Devuelve el determinante matricial de una matriz.
MINVERSA	MINVERSE	Devuelve la matriz inversa de una matriz dentro de una matriz.
MMULT	MMULT	Devuelve el producto matricial de dos matrices, una matriz con el mismo número de filas que Matriz1 y columnas que Matriz2.
MULTINOMIAL	MULTINOMIAL	Devuelve el polinomio de un conjunto de números.
MULTIPLO.INFERIOR	FLOOR	Redondea un número hacia abajo, hasta el múltiplo significativo más cercano.
MULTIPLO.INFERIOR.EXACTO	FLOOR.PRECISE	Redondea un número hacia abajo, hasta el entero o múltiplo significativo más cercano.



TÓPICOS DE EXCEL (Material teórico gratuito 2019)



MULTIPLO.SUPERIOR	<i>CEILING</i>	Redondea un número hacia arriba, hasta el múltiplo significativo más cercano.
MULTIPLO.SUPERIOR.EXACTO	<i>CEILING.PRECISE</i>	Redondea un número hacia arriba, al entero o múltiplo significativo más cercano.
NUMERO.ROMANO	<i>ROMAN</i>	Convierte un número arábigo en romano, en formato de texto.
PI	<i>PI</i>	Devuelve el valor Pi con precisión de 15 dígitos.
POTENCIA	<i>POWER</i>	Devuelve el resultado de elevar el número a una potencia.
PRODUCTO	<i>PRODUCT</i>	Multiplica todos los números especificados como argumentos.
RADIANES	<i>RADIANS</i>	Convierte grados en radianes.
RAIZ	<i>SQRT</i>	Devuelve la raíz cuadrada de un número.
RAIZ2PI	<i>SQRTPI</i>	Devuelve la raíz cuadrada de (número * Pi).
REDOND.MULT	<i>MROUND</i>	Devuelve un número redondeado al múltiplo deseado.
REDONDEA.IMPARG	<i>ODD</i>	Redondea un número positivo hacia arriba y un número negativo hacia abajo hasta el próximo entero impar.
REDONDEA.PARG	<i>EVEN</i>	Redondea un número positivo hacia arriba y un número negativo hacia abajo hasta el próximo entero par.
REDONDEAR	<i>ROUND</i>	Redondea un número al número de decimales especificado.
REDONDEAR.MAS	<i>ROUNDUP</i>	Redondea un número hacia arriba, en dirección contraria a cero.
REDONDEAR.MENOS	<i>ROUNDDOWN</i>	Redondea un número hacia abajo, hacia cero.
RESIDUO	<i>MOD</i>	Proporciona el residuo después de dividir un número por un divisor.
SENO	<i>SIN</i>	Devuelve el seno de un ángulo determinado.
SENOH	<i>SINH</i>	Devuelve el seno hiperbólico de un número.
SIGNO	<i>SIGN</i>	Devuelve el signo de un número: 1, si el número es positivo; cero, si el número es cero y -1, si el número es negativo.
SUBTOTALES	<i>SUBTOTAL</i>	Devuelve un subtotal dentro de una lista o una base de datos.
SUMA	<i>SUM</i>	Suma todos los números en un rango de celdas.
SUMA.CUADRADOS	<i>SUMSQ</i>	Devuelve la suma de los cuadrados de los argumentos.
SUMA.SERIES	<i>SERIESSUM</i>	Devuelve la suma de una serie de potencias.
SUMAPRODUCTO	<i>SUMPRODUCT</i>	Devuelve la suma de los productos de rangos o matrices correspondientes.
SUMAR.SI	<i>SUMIF</i>	Suma las celdas que cumplen determinado criterio o condición.
SUMAR.SI.CONJUNTO	<i>SUMIFS</i>	Suma las celdas que cumplen un determinado conjunto de condiciones o criterios.
SUMAX2MASY2	<i>SUMX2PY2</i>	Devuelve la suma del total de las sumas de cuadrados de números en dos rangos o matrices correspondientes.
SUMAX2MENOSY2	<i>SUMX2MY2</i>	Suma las diferencias entre cuadrados de dos rangos o matrices correspondientes.
SUMAXMENOSY2	<i>SUMXMY2</i>	Suma los cuadrados de las diferencias en dos rangos correspondientes de matrices.
TAN	<i>TAN</i>	Devuelve la tangente de un ángulo.
TANH	<i>TANH</i>	Devuelve la tangente hiperbólica de un número.
TRUNCAR	<i>TRUNC</i>	Convierte un número decimal a uno entero al quitar la parte decimal o de fracción.

Es evidente que el usuario de Excel utilizará más asiduamente algunas funciones, dependiendo de la aplicación que le éste le dé al trabajo con planillas de cálculo; pero resulta interesante el conocer, al menos parcialmente, la utilidad de cada grupo de funciones.



En el próximo apartado estudiaremos como potenciar el uso de funciones a partir de las llamadas fórmulas, que nos permitirán personalizar en gran medida el alcance y la aplicación de dichas funciones.

Las funciones ayudan a reducir en gran medida, la necesidad de la utilización de fórmulas escritas manualmente; pero, a la vez, las fórmulas nos permiten utilizar funciones encadenadas dentro de operaciones matemáticas.

2.3. Diferencia entre funciones y fórmulas

Resulta importante hacer la distinción entre que son las funciones, cuya definición estudiamos anteriormente, y que son las fórmulas en Excel.

A diferencia de las funciones, (que están preestablecidas) las fórmulas en Excel son generadas por el usuario y nos permiten realizar cálculos que incluyan una o más funciones, así como también números y operadores matemáticos.

Podríamos decir, estrictamente hablando que todo lo que se encuentre a la derecha de un símbolo igual “=” será tratado por Excel como si fuese una fórmula. Cuyo resultado será válido si está bien escrita, o bien, será informado como un error si no lo está.

	A	B	C
1	1		
2	2		
3	3	=A1+A2+A3+A4+suma(A6:A8)	
4	4		
5			
6	5		
7	6		
8	7		

Figura 2.7. Una fórmula puede contener funciones, referencias a celdas, números y operadores matemáticos.

2.4. Caracteres de TEXTO dentro de las funciones y fórmulas

Supongamos que queremos unir el contenido de dos celdas, podemos utilizar una fórmula que incluya el operador de unión “&” o la función CONTATENAR.



A	B	C
123		=A1&A2
.000		
		123.000

Figura 2.8. Utilizando el operador de unión "&" para unir el contenido de dos o más celdas.

	A	B	C	D
1	123		123.000	
2	.000			
3			=CONCATENAR(A1;A2)	

Figura 2.9. Utilizando la función CONCATENAR (CONCAT a partir de Excel 2016) para unir el contenido de dos o más celdas.

Si además quisiéramos agregarle una cadena de texto a esta unión (por ejemplo la palabra PESOS), deberíamos escribirla entre comillas (" PESOS", nótese que se escribió un espacio entre la primer doble comilla y la letra P), ya que de otra manera Excel no interpretaría dicha palabra como una cadena de texto.

	A	B	C
1	123		=A1&A2&" PESOS"
2	.000		
3			=CONCATENAR(A1;A2; " PESOS")
4			
5			123.000 PESOS

Figura 2.10. Nótese que, al agregar una cadena de texto, ésta debe ser escrita entre comillas.

2.5. Errores en las Funciones y Fórmulas en Excel

Al ingresar incorrectamente una fórmula, o una función, Excel mostrará distintos tipos de errores, resulta muy recomendable que el usuario de Excel se familiarice con el motivo de aparición de los mismos, para poder solucionarlos con mayor celeridad.



Los principales errores mostrados por Excel son:

Error:

Puede ocurrir por dos motivos.

El primero es debido a que el número tiene muchas cifras y no cabe dentro de la columna, al tener esta un ancho menor al adecuado.

Solución: Ajustar el ancho de la columna hasta que desaparezca en el error.

El segundo motivo de aparición es porque se le dio formato de fecha a un número negativo.

Solución: Quitar en signo negativo al número al que se le quiere dar formato fecha.

Error: #¡DIV/0!

El error se produce cuando se quiere dividir un valor entre cero.

Solución: Remover la división entre cero.

Error: #N/A

Este error significa que Excel no encuentra el valor al que se está refiriendo. Este tipo de error es muy común al utilizar la función BUSCARV.

Solución: Se puede utilizar la función SI.ERROR (Ver Figura 11)

The screenshot shows an Excel spreadsheet with the following data:

	A	B	C	D	E
7		1			
8		2	=SI.ERROR((BUSCARV(4;A7:A9;1;0));"UPPS")		
9		3			
10			UPPS		

The formula bar at the top shows: `=SI.ERROR((BUSCARV(4;A7:A9;1;0));"UPPS")`

Figura 2.11. Al no encontrar el valor 4, dentro de la matriz de búsqueda, la fórmula devolverá **UPPS** como resultado. Si no se hubiera usado la función SI.ERROR, el resultado sería **#N/A**

Error: #¿NOMBRE?

La razón más común por la que se genera este error es el haber escrito mal el nombre de una función. También se produce por escribir mal un rango nombrado. Puede darse con bastante asiduidad cuando un usuario está acostumbrado a usar Excel en inglés y al pasar Excel en castellano intenta escribir las funciones con su nombre original. (y viceversa)



Solución: Asegurarse de escribir el nombre de la función correctamente, así como todos los rangos nombrados que se utilicen en las funciones.

Error: #¡NULO!

Este error se da cuando se quiere sumar la intersección de dos rangos disjuntos.

Solución: ¡No se puede sumar algo que no existe!

	A	B	C	D
7	1	3		2
8	2	2		3
9	3	3		4
10				
11	=SUMA(A7:B9 D7:D9)			
12				

Figura 2.12. Al no existir la intersección entre los rangos sumados, el resultado será el error #¡NULO!. Nótese que, al separar los rangos con un espacio, la función SUMA tomará la intersección de los mismos para sumar sus valores.

	A	B	C	D
7	1	3		
8	2	2	3	
9	3	3		
10				
11	=SUMA(A7:B9 A8:C8)			
12		4		

Figura 2.13. En este caso el resultado de la suma será el valor 4, ya que es la suma de los valores que componen la intersección de ambos rangos.

Error: #¡NUM!

La capacidad de procesamiento, en lo que respecta a números extremadamente grandes, tiene un límite en Excel. Si como resultado de una fórmula o función se produce un resultado mayor al que Excel puede computar, se producirá este error.

Ejemplo: =FACT(200) intentará calcular el resultado del número factorial correspondiente a 200, es decir $(1 \times 2 \times 3 \times 4 \times 5 \dots \times 198 \times 199 \times 200)$. Como el resultado es un número más grande al mayor que Excel puede procesar, éste devolverá el error #¡NUM!

Solución: No hay solución a este error, ya que se trata de una limitación de cálculo de Excel.



Error #¡REF!

Este error aparece cuando la celda (o el conjunto de ellas) que sirve como referencia al argumento de una función ha desaparecido porque se la ha eliminado. Este error es, tal vez, el más común de todos, y se suele producir al trabajar con grandes bloques de datos que se eliminan.

Solución: Ajustar en la función los argumentos que se han eliminado.

Error #¡VALOR!

Este error es generado por utilizar argumentos de tipo texto cuando en realidad la función espera valores numéricos. Es la forma que tiene Excel de decir "Hay algo incorrecto en la escritura de la fórmula. O puede que haya algo incorrecto en las celdas a las que hace referencia".

Ejemplo: =SUMA("A";"B"). Como se esperan argumentos numéricos, pero estos son cadenas de texto, el resultado de la función será: #¡VALOR!

Solución: Verificar que los tipos de datos de los argumentos sean los adecuados, en particular aquellos argumentos en los que la función espera un valor numérico.

2.6. Funciones recursivas y dinámicas

2.6.1. Funciones recursivas

En ocasiones puede ocurrir que al intentar ingresar una función o fórmula, Excel nos muestra esta ventana:

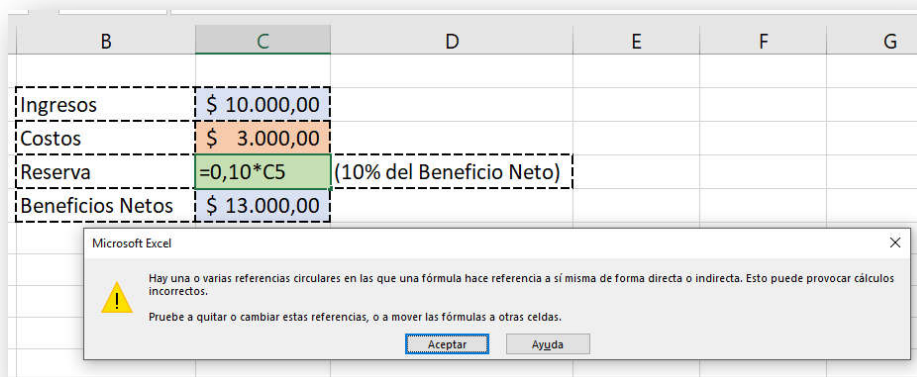


Figura 2.14. Ventana que informa la existencia de una referencia circular en la fórmula (es decir que el resultado de la misma alimenta a valores que están dentro de la propia fórmula).

En el ejemplo de la figura 14, la empresa tiene determinados ingresos, ciertos costos y una reserva que asciende al 10% de los beneficios netos (que a la vez se obtienen de restarle a los ingresos, los costos y también dicha reserva).



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



Excel entenderá que existe una referencia circular, ya que, por defecto, no permite el cálculo iterativo circular (es decir... “probar valores en una celda hasta que el resultado sea el buscado y se cumplan todas las condiciones en la celda correspondiente”)

Por lo general, esto nos ayuda a mantener una adecuada integridad en los cálculos que realiza Excel, ya que no es muy común que el resultado de una fórmula alimente a valores que está dentro de ésta.

En ocasiones, es posible que deseemos usar referencias circulares, ya que permiten que las funciones se iteren, es decir, se repitan hasta que se cumpla una condición numérica determinada. Esto puede hacer más lento el equipo, por lo que los cálculos iterativos normalmente están desactivados en Excel.

Si queremos permitir las referencias circulares, podemos habilitar los cálculos iterativos, aunque deberemos determinar cuántas veces se volverá a calcular la fórmula.

Si activamos los cálculos iterativos sin cambiar los valores preestablecidos, Excel detendrá los cálculos después de 100 iteraciones o después de que todos los valores de la referencia circular cambien por menos de 0,001 entre iteraciones (lo que suceda primero).

De todos modos, Excel nos permite controlar la cantidad máxima de iteraciones, así como también el valor de cambio mínimo de la referencia circular entre iteraciones.

A todas estas opciones se accede desde **INICIO – OPCIONES – FORMULAS** (en donde aparecerá una ventana similar a la de la figura 15).

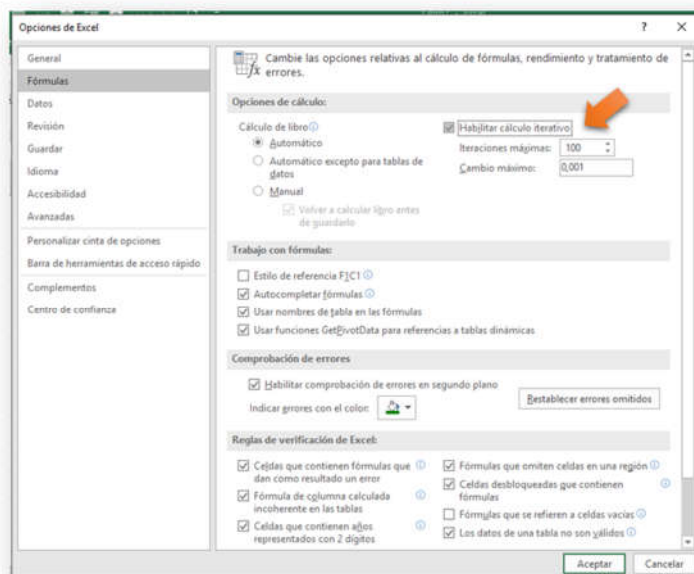


Figura 2.15. Habilitando el cálculo iterativo podremos utilizar funciones recursivas, para poder resolver casos similares a los del ejemplo planteado. También podremos ajustar la cantidad de iteraciones máximas y los valores de cambio mínimo entre iteraciones.

Una vez activada la función, Excel calculará sin problema alguno el valor buscado en el ejemplo, como lo muestra la figura 16.



	A	B	C	D	E	F	G
1							
2		Ingresos	\$ 10.000,00				
3		Costos	\$ 3.000,00				
4		Reserva	\$ 1.444,44	(10% del Beneficio Neto)			
5		Beneficios Netos	\$ 14.444,44				
6							

Figura 2.16. Activando el cálculo iterativo, Excel podrá mostrarnos el monto correspondiente a la reserva requerida.

Hablando en términos matemáticos, existen tres casos posibles como resultados de las iteraciones, estos son:

La solución converge: Se logra un resultado final estable. Esta es la condición deseable.

La solución diverge: En las iteraciones, aumenta la diferencia entre el resultado actual y el anterior.

La solución oscila: Ejemplo, después de la primera iteración el resultado es 1, después de la siguiente iteración el resultado es -1, después de la siguiente iteración el resultado es 1 y así sucesivamente.

2.6.2. Funciones dinámicas

Poco conocidas, existen en Excel dos funciones que llamaremos “dinámicas” ya que debido a su naturaleza y construcción permiten mostrar distintos resultados en función a los datos mostrados en pantalla y a los argumentos que se utilicen en dichas funciones.

Una de ellas es la función **SUBTOTAL** y la otra es la función **AGREGAR**. Esta segunda función (disponible a partir de Excel 2010) supera en mucho a la segunda y es la que explicaremos en esta ocasión.

La función **AGREGAR** (La traducción correcta de “AGREGATE” sería “TOTALIZAR”) realiza distintos tipos de cálculos en función al tipo de opción que se elija; además de tener en cuenta o no, distintos tipos de datos (como los datos ocultos), en cuanto el usuario así lo decida.

La metodología de uso de esta función es la siguiente:

=AGREGAR(**núm_función ; **opciones** ; **matriz** ; [k])**

Argumento: **núm_función** (Obligatorio)

Al ingresar este argumento, el cual debe ser un valor numérico, la función informará cual es la opción deseada, a saber:



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



Núm_ función	Función
1	PROMEDIO
2	CONTAR
3	CONTARA
4	MAX
5	MIN
6	PRODUCTO
7	DESVEST.M
8	DESVEST.P
9	SUMA
10	VAR.M

Núm_ función	Función
14	K.ESIMO.MAYOR
15	K.ESIMO.MENOR
16	PERCENTIL.INC
17	CUARTIL.INC
18	PERCENTIL.EXC
19	QUARTILCUARTIL.EXC

En este argumento se puede hacer referencia a una celda para que, al cambiar los valores de la misma, la función totalice el resultado con las distintas funciones. De forma similar, se puede utilizar la opción de “validación de datos” para generar una lista con los nombres de las funciones y a través de una función **BUSCARV**, que busque en una lista auxiliar, devolver el valor numérico de dicha opción.

Argumento: Opciones (Obligatorio)

Este argumento resulta ser un valor numérico que determina qué valores del rango de evaluación de la función se omitirán.

Opción	Comportamiento
0 u omitido	Omitir funciones AGREGAR y SUBTOTALES anidadas
1	Omitir filas ocultas y funciones AGREGAR y SUBTOTALES anidadas
2	Omitir valores de error y funciones AGREGAR y SUBTOTALES anidadas
3	Omitir filas ocultas, valores de error y funciones AGREGAR y SUBTOTALES anidadas
4	No omitir nada
5	Omitir filas ocultas
6	Omitir valores de error
7	Omitir filas ocultas y valores de error



Argumento: matriz (Obligatorio)

Este argumento representa la matriz o rango de datos que evaluará la función, puede ser un rango nombrado. Resulta importante destacar que la función AGREGAR está diseñada para columnas de datos o rangos verticales. No está diseñada para filas de datos ni para rangos horizontales.

Argumento: [k] (Obligatorio para algunas funciones)

Este argumento es un valor numérico que debe ser utilizado si se seleccionan determinadas funciones. Las funciones que exigen el uso de este argumento son:

Núm_ función	Función
14	K.ESIMO.MAYOR
15	K.ESIMO.MENOR
16	PERCENTIL.INC
17	CUARTIL.INC
18	PERCENTIL.EXC
19	QUARTILCUARTIL.EXC

2.6.3. Ejemplo del uso de la función AGREGAR

Se tiene un listado de vendedores en donde figura el nombre y el monto total vendido. Se quiere utilizar la función AGREGAR para saber cuál fue el promedio de las ventas. No se quiere omitir ninguna celda.

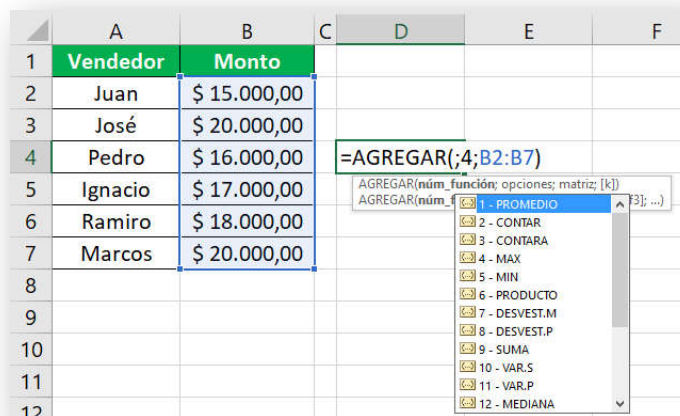


Figura 2.17. Como primer argumento se selecciona el correspondiente al PROMEDIO (lo que equivale a ingresar el valor 1).



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



	A	B	C	D	E	F	G	H
1	Vendedor	Monto						
2	Juan	\$ 15.000,00						
3	José	\$ 20.000,00						
4	Pedro	\$ 16.000,00		=AGREGAR(1;;B2:B7)				
5	Ignacio	\$ 17.000,00						
6	Ramiro	\$ 18.000,00						
7	Marcos	\$ 20.000,00						
8								
9								
10								

AGREGAR(núm_función; opciones; matriz; [k])

AGREGAR(núm_función; opciones; matriz; [k])

- 0 - Omitir las funciones AGREGAR y SUBTOTALES anidadas
- 1 - Omitir filas ocultas y funciones AGREGAR y SUBTOTALES anidadas
- 2 - Omitir valores de error y funciones AGREGAR y SUBTOTALES anidadas
- 3 - Omitir filas ocultas, valores de error y funciones AGREGAR y SUBTOTALES anidadas
- 4 - No omitir nada
- 5 - Omitir filas ocultas
- 6 - Omitir valores de error
- 7 - Omitir filas ocultas y valores de error

Figura 2.18. Como segundo argumento se selecciona el correspondiente la opción No omitir nada (lo que equivale a ingresar el valor 4).

En este caso se utiliza el ingreso manual de los argumentos, ya que es el primer ejemplo, durante el curso veremos formas más elegantes de utilizar esta poderosa función, lo que nos permitirá confeccionar planillas muy versátiles y limpias.

En la figura 19 veremos el resultado final, es decir el promedio calculado por la función AGREGAR:

	A	B	C	D
1	Vendedor	Monto		
2	Juan	\$ 15.000,00		
3	José	\$ 20.000,00		
4	Pedro	\$ 16.000,00		\$ 17.666,67
5	Ignacio	\$ 17.000,00		
6	Ramiro	\$ 18.000,00		
7	Marcos	\$ 20.000,00		
8				

Figura 2.19. Resultado final del cálculo del promedio con la función agregar. Nótese que no hizo falta el uso del parámetro [k], al no requerirlo la opción elegida en el primer parámetro.



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



3. Tablas dinámicas y Herramientas Business Intelligence en Excel

3.1. ¿Qué son las tablas dinámicas en Excel?

Para que los datos se conviertan en información útil para la toma de decisiones es necesario muchas veces agruparlos y presentarlos en forma de tablas. Pero, a menos que dominemos las herramientas de la Estadística Descriptiva, resulta bastante complicado el realizar este proceso, aun teniendo cierto dominio en la utilización de funciones y fórmulas en Excel.

La herramienta de Tablas Dinámicas en Excel nos permite procesar gran cantidad de información y resumirla en cuadros de múltiples entradas. Dicho de otra manera, Excel nos permite realizar de manera muy sencilla (en cuanto a los pasos a seguir) el llamado análisis multidimensional de nuestros datos.

Definiremos el concepto de análisis multidimensional, como la capacidad de contextualizar el comportamiento de una o más variables (denominadas medidas) mediante la utilización de dimensiones. Las medidas generalmente serán numéricas y las dimensiones habitualmente serán alfanuméricas.

Haciendo énfasis en las denominadas dimensiones, podemos definir las como un conjunto de elementos que poseen características comunes que responden a preguntas del tipo “quién” (cliente), “qué” (producto), “cuándo” (temporalidad), “dónde” (ubicación), “cómo” (canal de comercialización). A su vez, las dimensiones se pueden subdividir en jerarquías, niveles y categorías, con una estructura tipo árbol.

Con muy pocos pasos, la herramienta de Tablas Dinámicas en Excel permite realizar ajustes que resuman nuestros datos en función a cómo necesitamos visualizarlos. De igual manera podemos generar Gráficos Dinámicos que se actualizarán junto con las Tablas Dinámicas creadas (Figura 3.1).

Suma de Monto	Concepto					
Trimestres	LIBRERÍA	MANTENIMIENTO	REFRIGERIOS	VIATICOS	Total general	
Trim.1	\$ 10.964,00	\$ 28.257,00	\$ 13.605,00	\$ 11.604,00	\$ 64.430,00	
Trim.2	\$ 11.329,00	\$ 27.598,00	\$ 12.676,00	\$ 11.406,00	\$ 63.009,00	
Trim.3	\$ 13.502,00	\$ 24.484,00	\$ 12.216,00	\$ 10.346,00	\$ 60.548,00	
Trim.4	\$ 10.963,00	\$ 23.243,00	\$ 13.645,00	\$ 12.997,00	\$ 60.848,00	
Total general	\$ 46.758,00	\$ 103.582,00	\$ 52.142,00	\$ 46.353,00	\$ 248.835,00	



Figura 3.1. Sin necesidad de utilizar complicadas fórmulas, la herramienta tablas dinámicas de Excel permite realizar resúmenes de grandes cantidades de datos mediante cuadros y gráficos.



3.2. ¿Cómo activar la Herramienta Tablas Dinámicas?

Para activar la herramienta de Tablas Dinámicas necesitamos una fuente de datos (por ejemplo, una lista situada en una hoja de un libro en Excel) tabulada en columnas que tengan, en su primera fila, títulos. (Figura 3.2).

	A	B	C	D
1	Fecha	Sucursal	Concepto	Monto
2	1/1/2017	OESTE	MANTENIMIENTO	\$ 78,00
3	1/1/2017	OESTE	VIATICOS	\$ 634,00
4	1/1/2017	SUR	REFRIGERIOS	\$ 56,00
5	2/1/2017	SUR	VIATICOS	\$ 159,00
6	3/1/2017	NORTE	REFRIGERIOS	\$ 25,00
7	3/1/2017	OESTE	LIBRERÍA	\$ 91,00
8	3/1/2017	SUR	VIATICOS	\$ 415,00
9	4/1/2017	ESTE	VIATICOS	\$ 93,00
10	4/1/2017	OESTE	VIATICOS	\$ 786,00
11	4/1/2017	OESTE	LIBRERÍA	\$ 428,00
12	5/1/2017	ESTE	MANTENIMIENTO	\$ 379,00
13	5/1/2017	OESTE	MANTENIMIENTO	\$ 80,00
14	5/1/2017	OESTE	MANTENIMIENTO	\$ 21,00
15	6/1/2017	ESTE	REFRIGERIOS	\$ 116,00
16	6/1/2017	ESTE	MANTENIMIENTO	\$ 549,00
17	6/1/2017	NORTE	VIATICOS	\$ 338,00

Figura 3.2. El insumo de las tablas dinámicas son los datos tabulados, más adelante veremos cómo utilizar dicha herramienta para resumirlos.

Resulta muy sencillo el procedimiento para activar una tabla dinámica, el mismo consiste en seleccionar la lista con los datos, y luego la pestaña **INSERTAR** -> sección **TABLAS** -> opción **TABLA DINÁMICA**. (Figura 3.3)

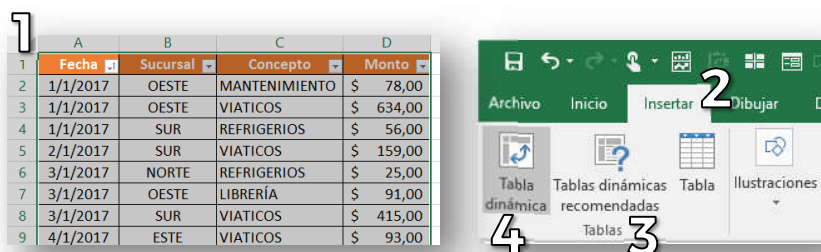


Figura 3.3. Para activar la herramienta tablas dinámicas: 1) se selecciona la lista que contenga los datos a analizar. 2) Pestaña INSERTAR. 3) Sección TABLAS. 4) Opción TABLA DINÁMICA.

Una vez activada la opción **TABLA DINÁMICA**, Excel nos mostrará una ventana de creación con algunas opciones relacionadas a la ubicación de la fuente de los datos y la de la tabla dinámica.

Si bien la mayoría de las veces bastará con dejar las opciones por defecto; resulta recomendable conocer los tres campos de la ventana de creación de la tabla dinámica (figura 3.4):



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)

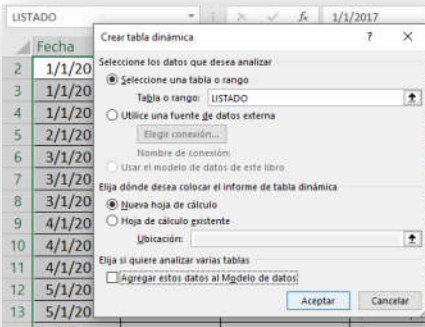


Figura 3.4. La ventana de creación de tablas dinámicas contiene tres campos de configuración referentes a: la fuente de datos, la ubicación de la tabla dinámica y la eventual relación entre dos o más tablas dinámicas.

Seleccione los datos que desea analizar: Se puede elegir entre un rango (o tabla ubicado en el libro) como así también, una fuente externa (por ejemplo, datos ubicados en otro archivo de Excel).

Elija dónde desea colocar el informe de tabla dinámica: Permite seleccionar la ubicación de la tabla dinámica; en una nueva hoja dentro del libro o dentro de la hoja donde están ubicados los datos.

Elija si quiere analizar varias tablas: Si se selecciona esta opción, los datos formarán parte de un eventual conjunto de tablas dinámicas, que podrán ser analizadas en forma conjunta.

Una vez seleccionadas las opciones, Excel mostrará la ventana de construcción de la tabla dinámica, en donde podremos seleccionar qué datos relacionar para generar el informe.

Básicamente la tabla dinámica, se estructura en dos secciones; la primera llamada **LISTA DE CAMPOS DE TABLA** (en donde elegiremos la ubicación de los datos y, eventualmente, aplicaremos los filtros) y la segunda, llamada **INFORME DE TABLA DINÁMICA** (que será el resultado que nos mostrará Excel) (Figura 3.5).

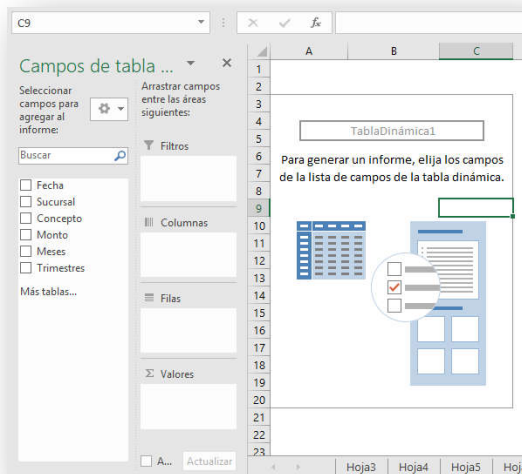


Figura 3.5. La lógica de construcción de la tabla es muy sencilla. Aunque puede parecer intimidante al contemplarla por primera vez, con pocos clics podemos generar informes muy versátiles.

Hay que destacar que no existe una única forma para ubicar los datos en la lista de campos, lo que permite que la herramienta de tablas dinámicas sea tan versátil para trabajar con datos masivos.



3.2.1. Ejemplo de aplicación

La mejor manera de dominar las Tablas Dinámicas en Excel es con la práctica guiada. En este módulo, trabajaremos con un caso de aplicación que iremos resolviendo y, simultáneamente, expondremos algunos conceptos teóricos que serán de gran utilidad en la aplicación de la herramienta a nuestro ejercicio profesional.

En el ejemplo tenemos un listado de 1000 entradas referidas a los egresos diarios de dinero que tuvieron, durante el año 2017, las cuatro sucursales de una empresa. Las sucursales están denominadas como **NORTE**, **SUR**, **ESTE** y **OESTE**; y los egresos están divididos en los siguientes conceptos: **MANTENIMIENTO**, **REFRIGERIOS**, **VIÁTICOS** y **LIBRERÍA** (en la anterior figura 3.2 se muestra un pequeño recorte de cómo está construido dicho listado).

La gerencia ha requerido la presentación de un informe referente a los gastos trimestrales de cada sucursal, desagregados en cada uno de los conceptos. El plazo de tiempo que nos otorgó para realizar este informe es de cinco minutos.

Solución:

- 1) Una vez activada la opción de tabla dinámica y seleccionado como fuente de datos el listado en cuestión. Procedemos a seleccionar los datos a utilizar, desplazando cada uno donde corresponda. El desplazamiento se realiza “arrastrando” la etiqueta mientras se mantiene presionado el botón izquierdo del mouse o del touchpad (ver figuras 3.6 a 3.9)

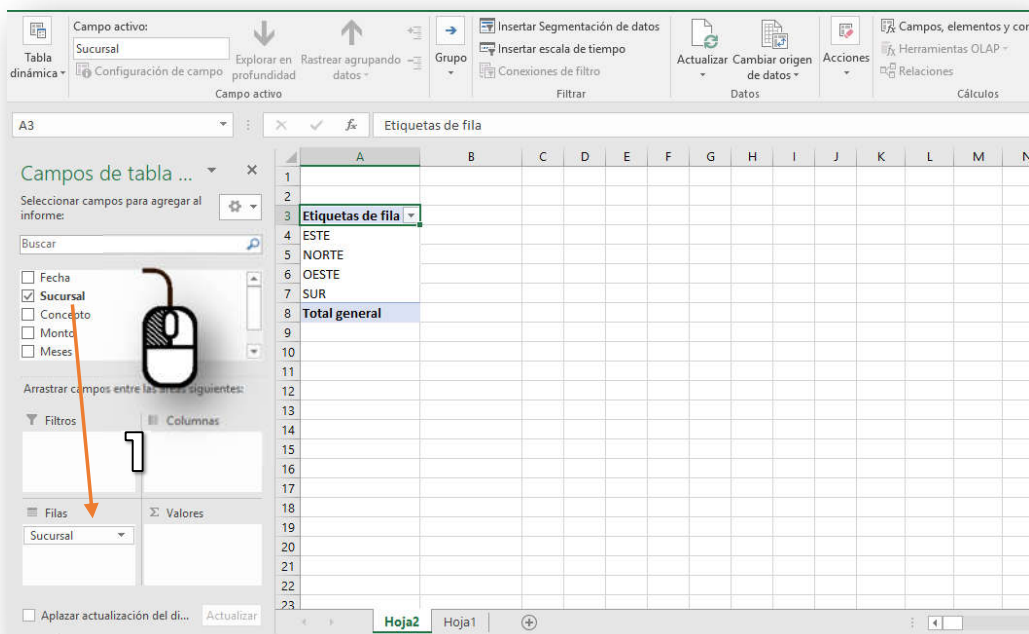


Figura 3.6. Para el ejemplo, desplazaremos la etiqueta SUCURSAL hacia el campo FILAS. Automáticamente los valores correspondientes a SUCURSAL se mostrarán ubicados por FILA en la sección INFORME de la tabla dinámica.



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)

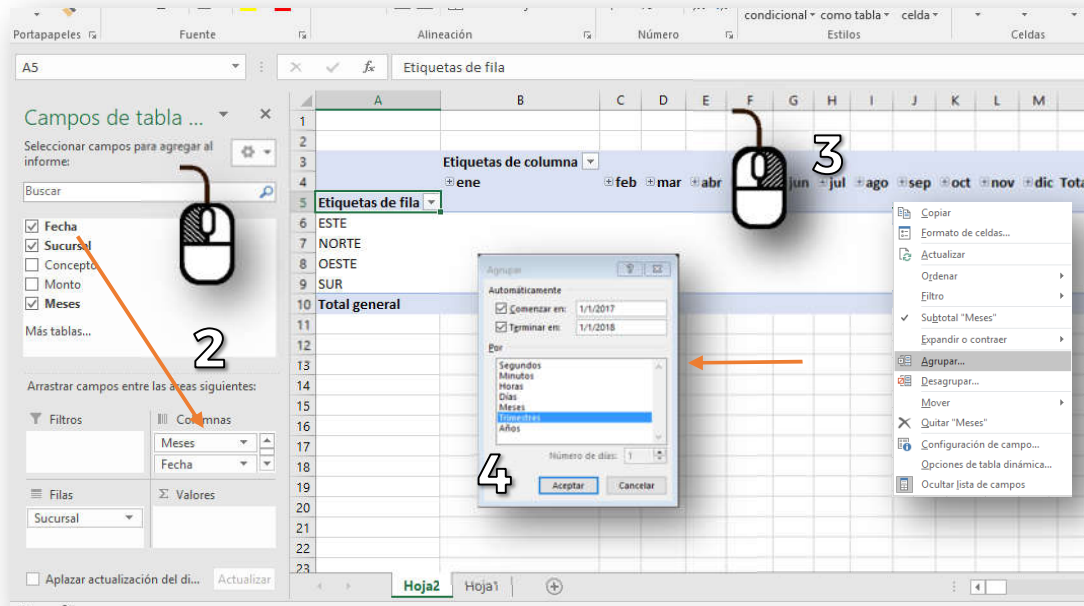


Figura 3.7. En las últimas versiones de Excel, al seleccionar una etiqueta que tenga datos en formato fecha, automáticamente se agruparán; en este caso se agrupan en meses, nosotros podemos agruparlas manualmente en trimestres, por ejemplo.

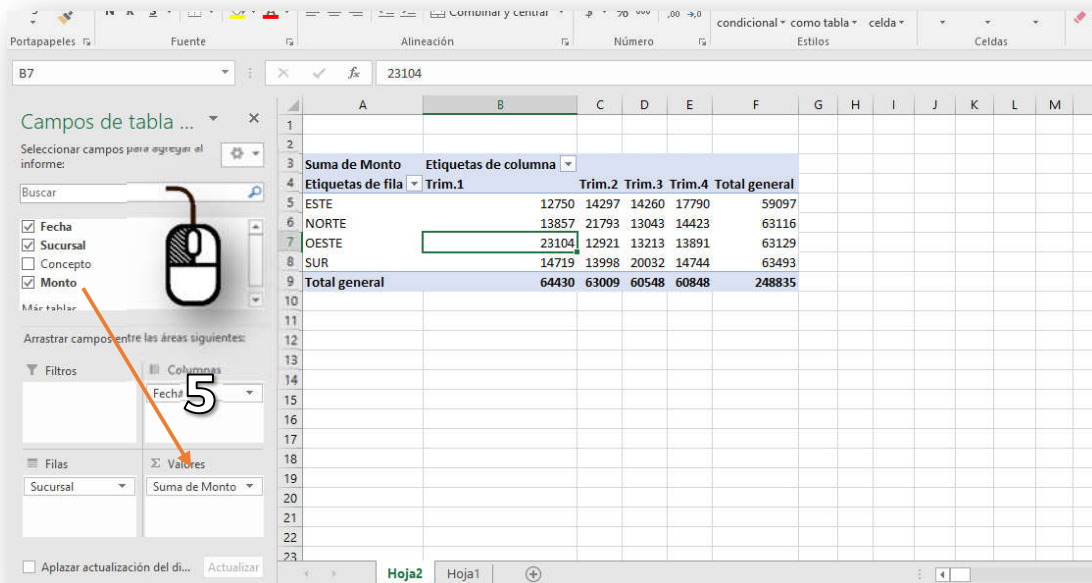


Figura 3.8. Si se agrega al campo VALORES datos numéricos, por defecto Excel procederá a realizar una suma de los mismos. Los valores serán mostrados sin formato definido. Esto se puede modificar manualmente.



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)

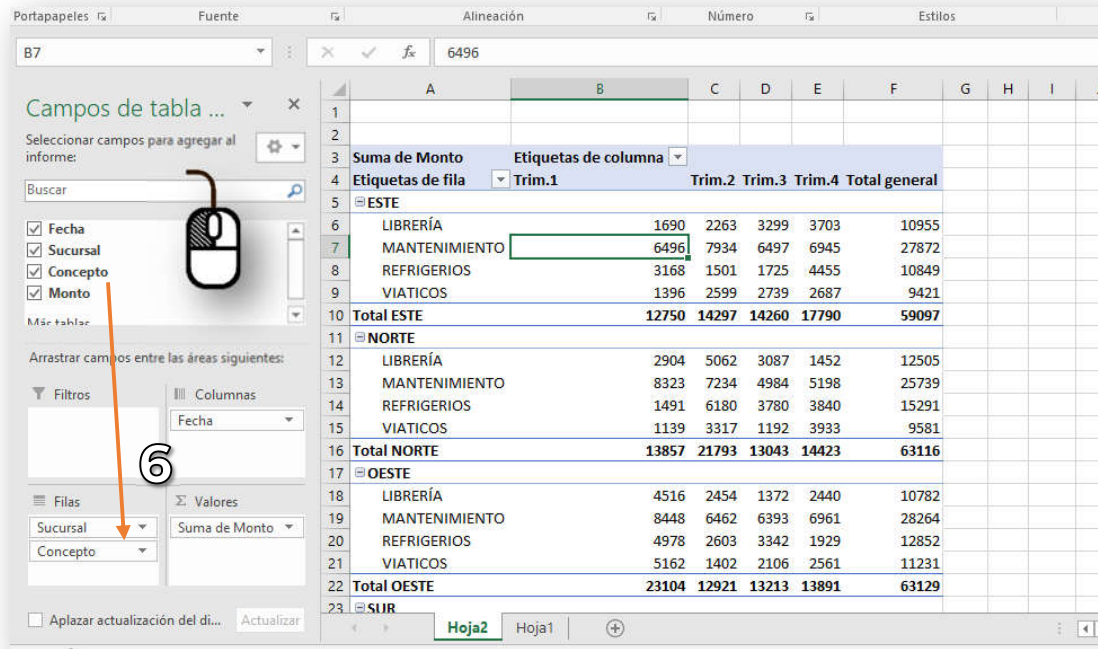


Figura 3.9. Se pueden agregar más etiquetas a las filas o a las columnas, de este modo podremos realizar un análisis multinivel con el nivel de desagregación que necesitemos.

Con el anterior ejemplo hemos visto lo simple que resulta la construcción de las tablas dinámicas, pero solamente hemos arañado la superficie de la inmensa cantidad de configuraciones que nos ofrece Excel respecto a esta herramienta.

Al activar las Tablas Dinámicas, Excel nos mostrará dos nuevas barras de herramientas: **ANALIZAR** y **DISEÑO** (Figuras 3.10 y 3.11)

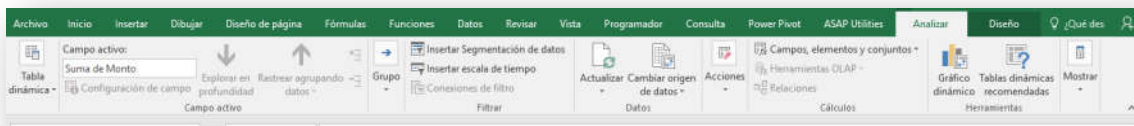


Figura 3.10. La pestaña ANALIZAR nos ofrece una serie de herramientas complementarias como, por ejemplo, la opción de insertar Gráficos Dinámicos o los filtros visuales con la opción Insertar Segmentación de datos.

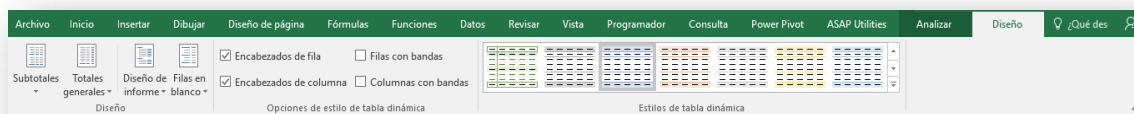


Figura 3.11. La pestaña DISEÑO nos permite personalizar la apariencia del informe generado por la herramienta de tablas dinámicas.



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



3.3. Tipos de Informes Dinámicos

Excel permite mostrar en el área Valores, de los informes generados por las tablas dinámicas, varias opciones de visualización y cálculo.

Los datos del área Valores, por defecto, se muestran calculados dependiendo que tipo de datos tenga como origen. Si los datos de origen son numéricos, los valores aparecerán sumados. Si los datos de origen son del tipo alfanumérico, los valores aparecerán contados.

De manera opcional, también puede crear un cálculo personalizado para que los valores se muestren de una manera alternativa a la visualizada por defecto.

3.4. Personalizando los cálculos

Una vez confeccionado el informe dinámico, seguiremos utilizando el ejemplo anterior, podríamos por ejemplo querer saber cuáles fueron los gastos promedios trimestrales, por concepto, de cada sucursal, para esto deberíamos realizar los siguientes pasos: (figura 3.12 a 3.14)

Cabe aclarar que existen maneras alternativas a la mostrada en la figura 3.12, pero todas llevan a la misma ventana de configuración.

Sucursal	Concepto	Trim.1	Trim.2	Trim.3	Trim.4	Total general
ESTE	LIBRERÍA	1690	2263	3299	3703	10955
	MANTENIMIENTO	6496	7934	6497	6945	27872
	REFRIGERIOS	3168	1501	1725	4455	10849
	VIATICOS	1396	2599	2739	2687	9421
Total ESTE		12750	14297	14260	17790	59097
NORTE	LIBRERÍA	2904	5062	3087	1452	12505
	MANTENIMIENTO	8323	7234	4984	5198	25739
	REFRIGERIOS	1491	6180	3780	3840	15291
	VIATICOS	1139	3317	1192	3933	9581
Total NORTE		13857	21793	13043	14423	63116
OESTE	LIBRERÍA	4516	2454	1372	2440	10782
	MANTENIMIENTO	8448	6462	6393	6961	28264
	REFRIGERIOS	4978	2603	3342	1929	12852
	VIATICOS	5162	1402	2106	2561	11231
Total OESTE		23104	12921	13213	13891	63129
SUR	LIBRERÍA	1854	1550	5744	3368	12516
	MANTENIMIENTO	4990	5968	6610	4139	21707
	REFRIGERIOS	3968	2392	3369	3421	13150
	VIATICOS	3907	4088	4309	3816	16120
Total SUR		14719	13998	20032	14744	63493

Figura 3.12. En el campo Valor, clic izquierdo en la etiqueta y luego seleccionar la opción "Configuración de Campo de Valor".

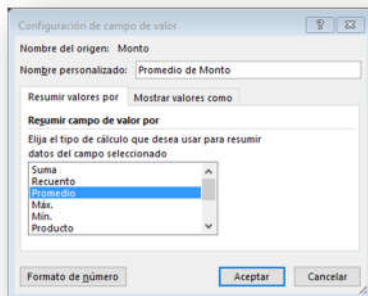


Figura 3.13. En la ventana Configuración de campo Valor, seleccionamos el tipo de cálculo que queremos aplicar. En este caso escogeremos la opción Promedio.



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



Sucursal	Concepto	Trim.1	Trim.2	Trim.3	Trim.4	Total general
ESTE	LIBRERÍA	281,67	226,30	219,93	231,44	233,09
	MANTENIMIENTO	224,00	273,59	232,04	330,71	260,49
	REFRIGERIOS	198,00	150,10	215,63	234,47	204,70
	VIATICOS	155,11	259,90	304,33	223,92	235,53
Total ESTE		212,50	242,32	237,67	261,62	239,26
NORTE	LIBRERÍA	264,00	266,42	171,50	145,20	215,60
	MANTENIMIENTO	297,25	258,36	216,70	236,27	254,84
	REFRIGERIOS	165,67	386,25	315,00	182,86	263,64
	VIATICOS	227,80	301,55	170,29	245,81	245,67
Total NORTE		261,45	294,50	217,38	209,03	246,55
OESTE	LIBRERÍA	265,65	350,57	196,00	221,82	256,71
	MANTENIMIENTO	272,52	190,06	266,38	248,61	241,57
	REFRIGERIOS	292,82	371,86	278,50	241,13	292,09
	VIATICOS	430,17	280,40	162,00	284,56	287,97
Total OESTE		300,05	243,79	235,95	248,05	260,86
SUR	LIBRERÍA	168,55	258,33	287,20	240,57	245,41
	MANTENIMIENTO	207,92	213,14	287,39	258,69	238,54
	REFRIGERIOS	220,44	398,67	210,56	263,15	248,11
	VIATICOS	217,06	371,64	253,47	272,57	268,67
Total SUR		207,31	274,47	263,58	258,67	248,99
Total general		246,86	265,86	240,27	243,39	248,84

Figura 3.14. La tabla dinámica mostrará los valores promediados. Para mostrar el resultado con dos números decimales se le aplicó un formato numérico (pestaña INICIO, sección NÚMERO)

3.5. Tipo de funciones de resumen de datos en tablas dinámicas

Entre los tipos de funciones que Excel permite utilizar en el resumen de los datos, tenemos las que se encuentran en la siguiente lista:

Función	Resumen
Suma	La suma de los valores. Es la función predeterminada de los valores numéricos.
Contar	El número de valores. Es la función predeterminada de valores que no son numéricos.
Promedio	El promedio de los valores.
Máx.	El valor máximo.
Mín.	El valor mínimo.
Producto	El producto de los valores.
Contar números	Número de valores que son números.
DesvEst	Un cálculo de la desviación estándar de una muestra
DesvEstP	La desviación estándar de una población
Var	Un cálculo de la varianza de una muestra
VarP	La varianza de una población.



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



3.5.1. Mostrando valores relativos

Siguiendo con el ejemplo anterior, el gerente podría requerir, que el informe muestre valores porcentuales, tomando como base cada trimestre. Esto resulta también muy sencillo de aplicar desde la ventana de *Configuración de campo de valor* (figura 3.15)

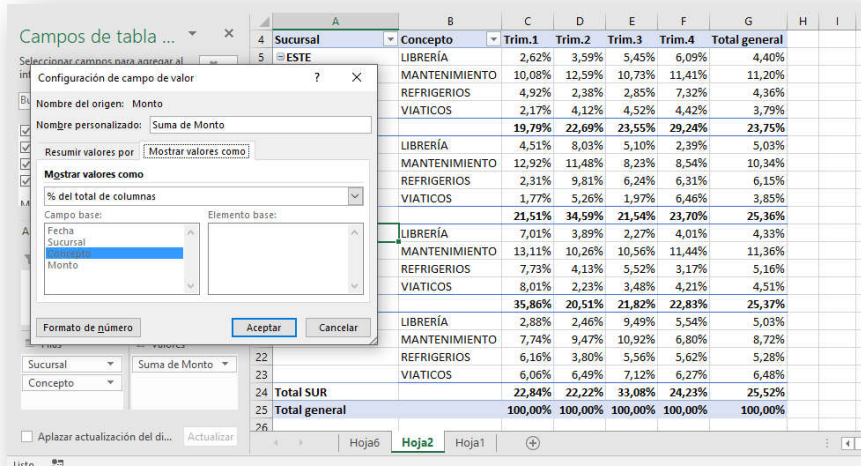


Figura 3.15. En la ventana de Configuración de campo de valor, seleccionando la pestaña % total de columnas, podremos visualizar los valores relativos a cada trimestre.

Entre las opciones que nos ofrece Excel en la pestaña “Mostrar valores como” encontraremos:

Opción de cálculo	Resultado
<i>Sin cálculo</i>	Muestra el valor que se introduce en el campo.
<i>% del total general</i>	Muestra los valores como un porcentaje del total general de todos los valores o puntos de datos en el informe.
<i>% del total de columnas</i>	Muestra todos los valores de cada columna o serie como un porcentaje del total de la columna o serie.
<i>% del total de filas</i>	Muestra el valor de cada fila o categoría como un porcentaje del total de la fila o categoría.
<i>% de</i>	Muestra los valores como un porcentaje del valor del Elemento base en el Campo base.
<i>% del total de filas principales</i>	Calcula los valores de la siguiente forma: (valor del elemento) / (valor del elemento principal en las filas)
<i>% del total de columnas principales</i>	Calcula los valores de la siguiente forma: (valor del elemento) / (valor del elemento principal en las columnas)
<i>% del total principal</i>	Calcula los valores de la siguiente forma: (valor del elemento) / (valor del elemento principal del campo Base seleccionado)



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



Opción de cálculo	Resultado
<i>Diferencia de</i>	Muestra los valores como la diferencia del valor del Elemento base en el Campo base.
<i>% de la diferencia de</i>	Muestra los valores como la diferencia de porcentaje del valor del Elemento base en el Campo base.
<i>Total en</i>	Muestra el valor de elementos sucesivos en el campo Base como un total acumulado.
<i>% del total en</i>	Calcula el valor como un porcentaje de elementos sucesivos en campo Base que se muestran como un total acumulado.
<i>Clasificar de menor a mayor</i>	Muestra el rango de valores seleccionados en un campo específico, enumerando el elemento más pequeño en el campo como 1 y cada valor mayor tendrá un valor de jerarquía más alto.
<i>Clasificar de mayor a menor</i>	Muestra el rango de valores seleccionados en un campo específico, enumerando el elemento más grande en el campo como 1 y cada valor menor tendrá un valor de jerarquía más alto.
<i>Índice</i>	Calcula los valores de la siguiente forma: $((\text{valor en celda}) \times (\text{Total general de Totales generales})) / ((\text{Suma total de filas}) \times (\text{Suma total de columnas}))$

3.6. Personalizando la apariencia de los informes

Excel permite personalizar la apariencia de los informes. Lo anterior se consigue muy fácilmente a través de la pestaña DISEÑO mostrada en la figura 3.16, ésta tiene tres secciones, las cuales son:



Figura 3.16. La pestaña diseño tiene tres secciones. La sección DISEÑO, la sección OPCIONES DE ESTILO DE TABLA DINÁMICA y la sección Estilos de tabla dinámica.

- 1) **Sección DISEÑO:** Aquí podemos personalizar la manera en la cual se va a estructurar la visualización de los datos en el informe dinámico. Tiene cuatro opciones que permite trabajar con la visualización de: subtotales, totales generales, el tipo de diseño del informe y la visualización, o no, de filas en blanco dentro del mismo. (ver figuras 3.17 a 3.21).

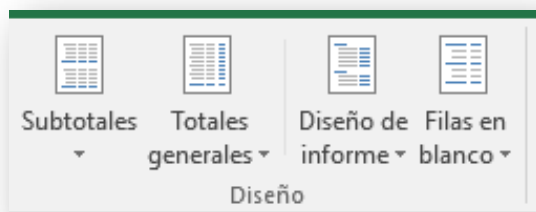


Figura 3.17. La sección DISEÑO está compuesta por cuatro opciones relacionadas con la manera en la que se visualizarán los datos en el informe dinámico.



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)

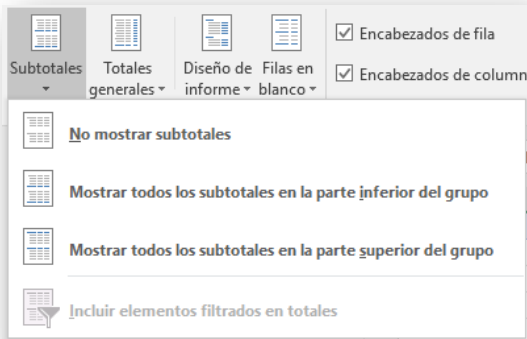


Figura 3.18. La opción Subtotales tiene cuatro opciones (La cuarta se activa cuando hay elementos filtrados en el informe dinámico).

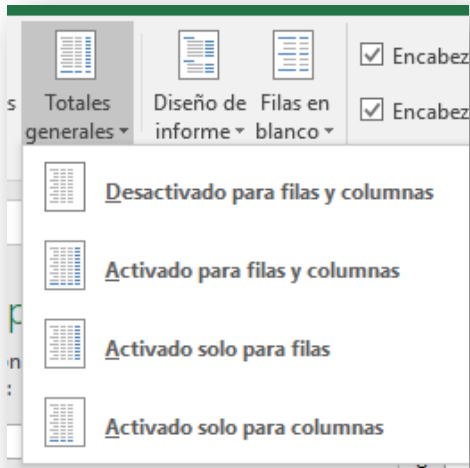


Figura 3.19. La opción Totales Generales permite personalizar la visualización de los totales calculados automáticamente al generar el informe dinámico.

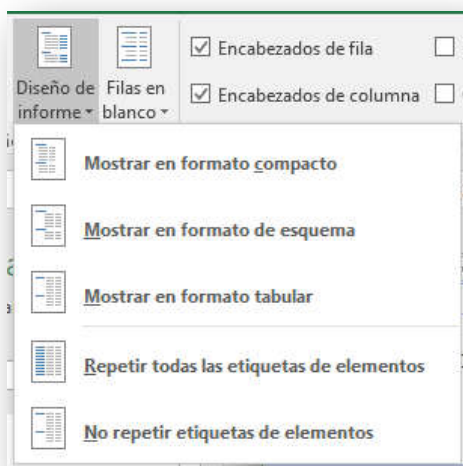


Figura 3.20. La opción Diseño de Informe permite personalizar la manera en que se muestran los encabezados y la repetición, o no de las etiquetas en las filas.



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)

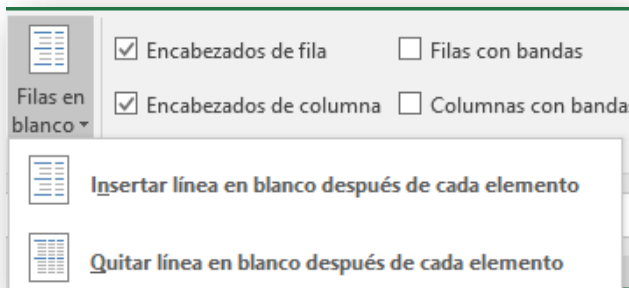


Figura 3.21. La opción Filas en blanco permite insertar, o quitar, espacios en blanco luego de cada elemento.

- 2) **Sección Opciones de estilo de tabla dinámica:** Aquí podremos activar o desactivar la visualización de bandas y encabezados, tanto en filas como en columnas (figura 3.22)

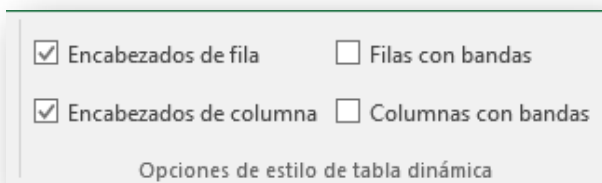


Figura 3.22. La sección Opciones de estilo de tabla dinámica cuenta con dos campos para las filas y dos campos para las columnas.

- 3) **Sección Estilo de tabla dinámica:** La última de las opciones, nos permite asignarles estilos visuales a nuestros informes dinámicos. Cuenta con más de 90 estilos y nos da la posibilidad de personalizar el estilo visual de cada uno de los componentes del informe (figura 3.23)

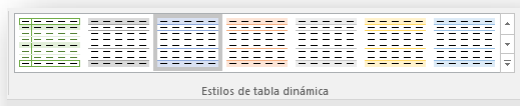
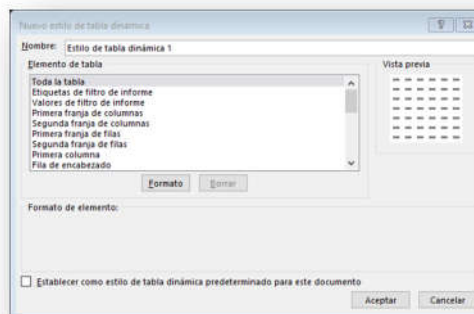


Figura 3.23. La sección Estilos de tabla dinámica permite personalizar el estilo visual de nuestro informe dinámico.



Además de la personalización del estilo visual y del diseño del informe dinámico, Excel nos permite filtrar etiquetas, para así mostrar solamente la información que queremos analizar. Tradicionalmente se utilizaba el campo **FILTRO**; pero a partir de la versión 2010 tenemos otras herramientas que veremos en la sección siguiente de este módulo.



3.7. Gráficos dinámicos y Datos segmentados

La pestaña **ANALIZAR** de la herramienta Tablas Dinámicas, nos aporta una serie de opciones extremadamente versátiles para acompañar al informe dinámico en el análisis de datos masivos para la toma de decisiones. Entre las herramientas más utilizadas, encontraremos *Gráficos dinámicos* y *Segmentación de datos* (Figura 3.24)

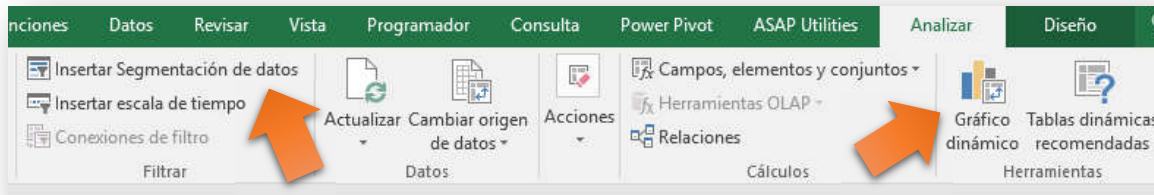


Figura 3.24. Estas opciones permiten, junto al informe generado por la tabla dinámica, la generación de tableros visuales para el análisis rápido de la información.

3.7.1. Gráficos dinámicos

Si bien es cierto que las tablas dinámicas son una herramienta analítica eficiente, debemos ser conscientes que un gráfico es más poderoso para transmitir información, puesto que permite al usuario final obtener conclusiones sobre los datos, rápidamente.

En una tabla dinámica puede resultar muy difícil identificar relaciones, tendencias, diferencias o cualquier conclusión relevante, sin embargo, en un gráfico dinámico de Excel, ambas funcionalidades son complementarias ya que las tablas dinámicas se encargan de manipular y resumir los datos, mientras que los gráficos facilitan el entendimiento de dicho análisis, a través de elementos visuales.

Utilizaremos el ejemplo anterior para generar un gráfico dinámico a partir de un informe en donde se muestren, por concepto, el gasto anual de cada sucursal (ver figura 3.25).

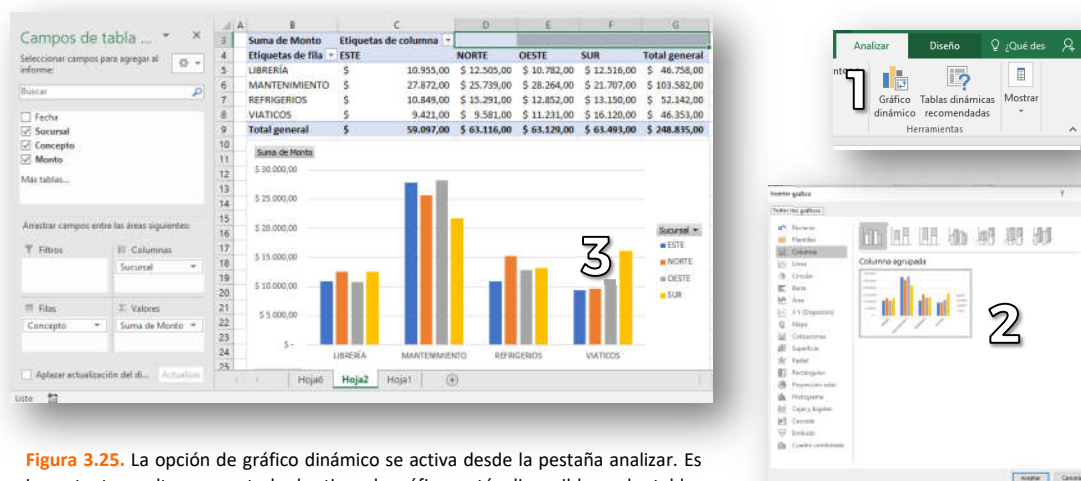


Figura 3.25. La opción de gráfico dinámico se activa desde la pestaña analizar. Es importante resaltar que no todos los tipos de gráficos están disponibles en las tablas dinámicas.



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



Supongamos ahora, que queremos cambiar el informe y que nos muestre cuánto dinero egresó por trimestre. Al modificar la posición de las etiquetas en el panel Campos de tabla dinámica, también se modificará el gráfico dinámico (figura 3.26).

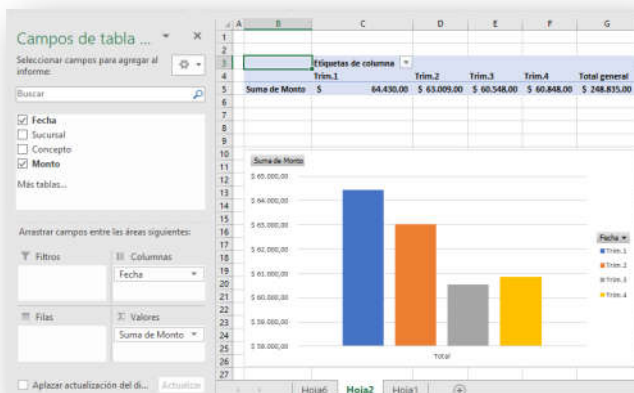


Figura 3.26. Al cambiar las etiquetas, y modificar el informe dinámico, también cambiará el gráfico dinámico.

3.7.2. Segmentación de datos

La herramienta segmentación de datos proporciona botones en los que puede hacer clic para filtrar datos de una tabla dinámica. Esta funcionalidad está disponible a partir de Excel 2010.

Además de permitir el filtrado rápido, las segmentaciones de datos también no indican el estado de filtrado actual, haciendo fácil la tarea de entender qué es lo que se está mostrando, y lo que no, en una tabla dinámica filtrada.

Volveremos a utilizar el ejemplo del módulo. Esta vez, vamos a analizar un informe en donde se muestren, por concepto, el gasto anual de cada sucursal, pero aplicaremos filtros por sucursal y por concepto. (Figura 3.27 y 3.28).

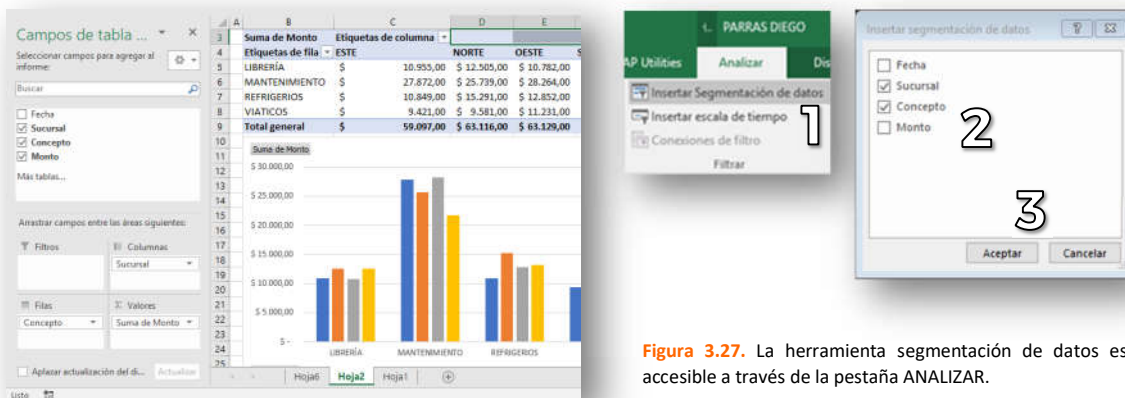


Figura 3.27. La herramienta segmentación de datos es accesible a través de la pestaña ANALIZAR.



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)

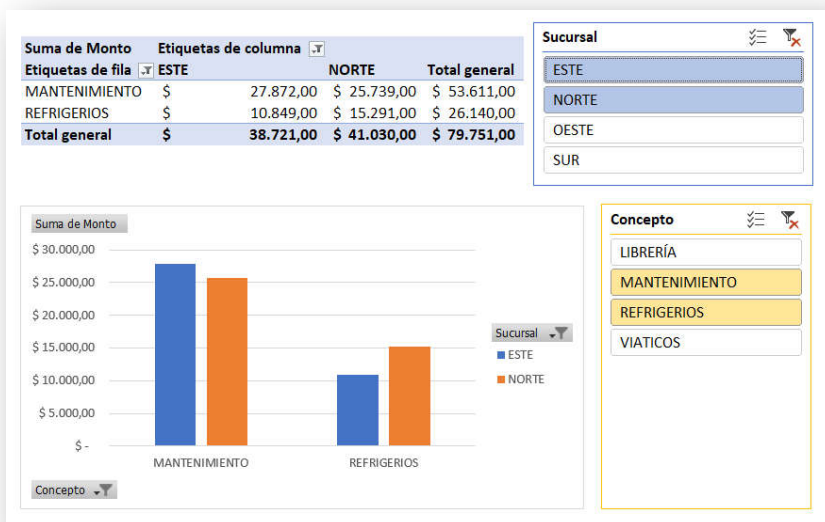


Figura 3.28. Con la herramienta Tablas dinámicas, segmentación de datos y gráficos dinámicos, se pueden crear tableros de gestión con mucha facilidad.

En este módulo introductorio al estudio de la herramienta Tablas Dinámicas en Excel hemos podido observar la simplicidad y versatilidad de esta funcionalidad.

Si bien puede parecer intimidante en un principio, las tablas dinámicas son una de las herramientas más sencillas de utilizar, de las disponibles en Excel. Resta comenzar a aplicarlas en nuestro ejercicio profesional para incrementar exponencialmente nuestra productividad.

Nos despedimos de los señores lectores hasta el próximo módulo, en donde abordaremos los fundamentos de la programación en Excel a partir de Macros y las herramientas VBA.

3.8. Modelos de Datos

Un modelo de datos es, básicamente, un método para integrar datos de varias tablas, para luego generar de una manera relativamente sencilla, una base de datos relacional en un libro de Excel. En éste los modelos de datos se usan de forma transparente para el usuario, resultando además uno de los pilares para las herramientas POWER de Excel.

Las herramientas POWER son un conjunto de potencian la capacidad de Excel en lo que respecta al manejo de grandes cantidades de datos. Como están relacionadas directamente con la construcción de bases de datos relacionales, es de vital importancia que el usuario de estas herramientas comprenda ciertos conceptos propios de los sistemas de datos.

Si bien existen dentro de la teoría diferentes tipos de modelos, el que utiliza Excel para alimentar las herramientas POWER se denomina modelo de datos relacional.

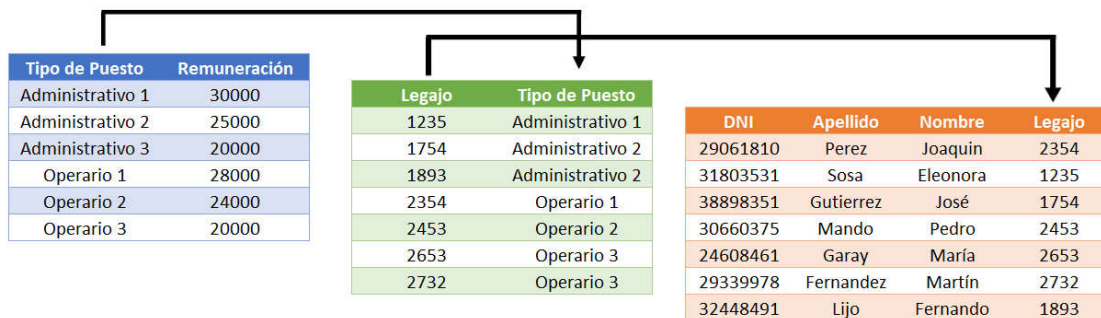


El objetivo principal de los modelos de datos es proporcionar información sobre la metodología de almacenamiento de datos. Un modelo de datos se puede asimilar como el esquema que aporta información sobre la organización de los datos.

3.8.1. Modelos de datos relacionales

De forma gráfica se puede presentar al modelo de datos relacional como un conjunto de tablas que están vinculadas entre sí por una columna en común (es habitual referirse a las columnas con el término campo, en el lenguaje de las bases de datos).

En la siguiente figura, se muestran tres tablas que describen diferentes conceptos, y que comparten columnas entre sí.



Los primeros conceptos de bases de datos que se abordarán serán los de tablas y relaciones; siendo una tabla un conjunto de filas y columnas, mientras que las relaciones son las columnas que comparten dos tablas entre sí.

El poder de las bases de datos relacionales reside en la posibilidad de cruzar los datos entre dos o más tablas, a partir de las relaciones existentes entre éstas. Esto se conoce con el término **CONSULTA**.

Las consultas pueden realizar diversas funciones en una base de datos. La función más común es recuperar datos específicos de las tablas. Los datos que quiere ver generalmente están distribuidos en varias tablas y las consultas le permiten verlos en una única hoja de datos.

Si se quisiera saber, por ejemplo, cual es la remuneración de cada empleado, se realiza una consulta utilizando como nexos a la columna que cada par de tabla tenga en común.





TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



DNI	Remuneración
29061810	28000
31803531	30000
38898351	25000
30660375	28000
24608461	20000
29339978	20000
32448491	25000

Las ventajas de los modelos de datos relacionales, en donde cada tabla contiene una parte de la totalidad de la información disponible son varias, entre ellas se pueden destacar las siguientes:

Ingreso de nuevos datos: Si se ingresan nuevas entradas en alguna de las tablas, éstas podrán conectarse fácilmente con las otras a través de las consultas.

Modificación de datos: Si se modifican los datos en alguna de las tablas, éstos estarán disponibles para las otras a través de las consultas.

Seguridad en el tratamiento de datos: Las tablas pueden ser administradas por distintas personas, lo cual redundará en la posibilidad de limitar cierta información a determinados usuarios.

Resulta importante el resaltar que es altamente recomendable que la información contenida en las tablas se encuentre optimizada utilizando en éstas las llamadas formas normales de bases de datos.

Si bien existe una vasta cantidad de bibliografía que se especializa en el tema de normalización de bases de datos, se puede establecer como un primer resumen, los siguientes pasos para optimizar la creación de tablas.

- 1) Los valores de cada entrada en la tabla deben ser atómicos: es decir que cada entrada debe contener solamente un valor. (Ej. Si un trabajador tiene dos teléfonos de contacto, estos deben aparecer cada uno en un campo separado)
- 2) Debe existir un campo identificador por cada tabla, del cual dependen todos los demás campos. (Ej. En la tabla empleados el campo DNI es el que identifica unívocamente a cada empleado). A este campo se lo denomina "llave primaria".
- 3) Si una tabla contiene valores que se repiten, es recomendable construir otra tabla, asignándole un código a esos valores. (Ej. Varios empleados pueden tener los mismos ingresos, pero por distintos puestos de trabajo)

3.9. Herramientas "Power" en Excel

Con la capacidad de procesamiento de los equipos modernos y el advenimiento del paradigma de almacenamiento online de datos, las organizaciones tienen a utilizar cada vez más información en sus procesos de negocio.



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



Si bien en su versión 2007 Excel aumentó su capacidad de 65.536 filas y 256 columnas hasta 1.048.576 filas y 16.384 columnas, el motor tradicional de gestión de datos en Excel no resulta adecuado para el manejo de gran cantidad de información.



Tradicionalmente, para trabajar con datos de múltiples fuentes en Excel, había que utilizar la herramienta de obtención de datos externos, para luego de importados los datos, utilizar funciones de búsqueda y referencia como las vistas en la clase 1.

Si bien el método tradicional, debido a la conocida flexibilidad en las herramientas de Excel, resulta adecuado en el tratamiento no masivo de datos (relativamente pocos datos, que provienen de fuentes tradicionales, por ejemplo: un archivo de texto, una base de Access, u otro libro de Excel); su versatilidad, velocidad y estabilidad presentan serias falencias cuando se trata de gestionar grandes cantidades de datos.

Como se ha argumentado, la velocidad y estabilidad de procesamiento en Excel disminuye en gran medida, al aumentar la cantidad de datos a procesar; tornándose en ciertas oportunidades poco adecuado ante experiencias en donde se necesita realizar procesos de simulación de escenarios, por ejemplo.

Es por ello que, al desarrollarse en sus inicios el paradigma de gestión y procesamiento masivo de datos (BIG DATA), los usuarios de Excel vieron que las capacidades de éste resultaban insuficientes. Si bien Excel no era adecuado para la nueva metodología de gestión de datos, la implementación cuasi universal de este software en las organizaciones, sumado a la dificultad de uso y costo de los programas específicos para el tratamiento BIG DATA, provocaron que las organizaciones que no disponían de grandes recursos postergasen la oportunidad de la aplicación de este paradigma.

Desde la versión 2010, incipientemente, Excel ha venido sumando cada vez más complementos que permitieron adicionarle renovadas y mejoradas capacidades de manejo de datos masivos.

Las herramientas Excel Power Business Intelligence, (en adelante Herramientas Power) permiten la gestión rápida y eficiente de grandes cantidades de datos, desde su importación y transformación desde múltiples fuentes (Power Query), su modelación relacional y tratamiento (Power Pivot) y su visualización en dashboards (Power View y Power Maps).

La idea subyacente en las Herramientas Power es la de permitir la creación de DASHBOARDS o tableros de comando para evaluar, entre otras opciones, distintos escenarios de trabajo, luego de la importación y posterior modelización de ingentes cantidades de datos.

Resultaría bastante pretencioso, además de poco acertado, realizar en pocas páginas una descripción completa de estas herramientas; pero a modo de introducción se presentarán las características principales de cada una de éstas.



3.9.1. Power Pivot

Power Pivot es una herramienta de Excel que fue introducida como un complemento en Excel 2010 y se convirtió en una opción nativa en el código de Excel 2016. Debido a esto la velocidad y estabilidad de esta herramienta se ha mejorado considerablemente respecto a versiones anteriores.

PowerPivot permite:

- 1) Importar millones de filas de data desde múltiples fuentes de datos a un de Excel.
- 2) Crear relaciones entre tablas de datos.
- 3) Generar columnas calculadas utilizando fórmulas especiales.
- 4) Construir Tablas y Gráficos Dinámicos relacionados.
- 5) Analizar a profundidad los datos para generar insumos para modelos predictivos aplicados.

PowerPivot dispone de su propio lenguaje de construcción de fórmulas (llamadas medidas), denominado DAX (Data Analysis Expressions). DAX es un conjunto de funciones, operadores y constantes que se pueden usar en una fórmula, o expresión, para calcular y regresar uno o más valores.

DAX permite al usuario generar nueva información desde los datos que se encuentren en el modelo, el aprendizaje de DAX no es difícil si se conoce la metodología de construcción de fórmulas a partir de funciones de Excel.

3.9.1.1. Utilizando PowerPivot

En el ejemplo siguiente se generará un modelo de datos a partir de algunas hojas de un mismo libro de Excel. En Hoja1 se aloja la tabla **TipoPuesto** en donde se encuentran las columnas *Tipo de Puesto* y *Remuneración*; en Hoja2 se aloja la tabla **Legajo** en donde se encuentran las columnas *Legajo* y *Tipo de Puesto*; en Hoja3 se aloja la tabla **Empleados** en donde se encuentran las columnas *DNI*, *Apellido*, *Nombre* y *Legajo*.

Tipo de Puesto	Remuneración
Administrativo 1	30000
Administrativo 2	25000
Administrativo 3	20000
Operario 1	28000
Operario 2	24000
Operario 3	20000

Legajo	Tipo de Puesto
1235	Administrativo 1
1754	Administrativo 2
1893	Administrativo 2
2354	Operario 1
2453	Operario 2
2653	Operario 3
2732	Operario 3

DNI	Apellido	Nombre	Legajo
29061810	Perez	Joaquin	2354
31803531	Sosa	Eleonora	1235
38898351	Gutierrez	José	1754
30660375	Mando	Pedro	2453
24608461	Garay	María	2653
29339978	Fernandez	Martín	2732
32448491	Lijo	Fernando	1893

El objetivo es generar una tabla dinámica que contenga las tres tablas. Tradicionalmente, se puede realizar esta tarea utilizando la función **BUSCARV**, relacionando las tablas de a pares. La desventaja de este método consiste en que, a medida que aumenta la cantidad de datos, las búsquedas se vuelven más lentas. Es importante recordar que, aún aceptando la pérdida de velocidad y de estabilidad a medida que aumenta la cantidad de datos, existe la limitación de no poder trabajar con más de 1.048.576 filas.



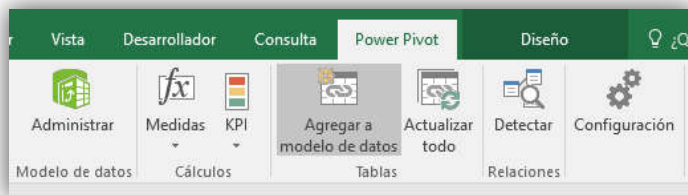
TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



Primer paso: Agregar las tablas al modelo de datos. Para ello se seleccionan las tablas desde el cuadro de nombres y se las agrega al modelo de datos desde la pestaña PowerPivot.

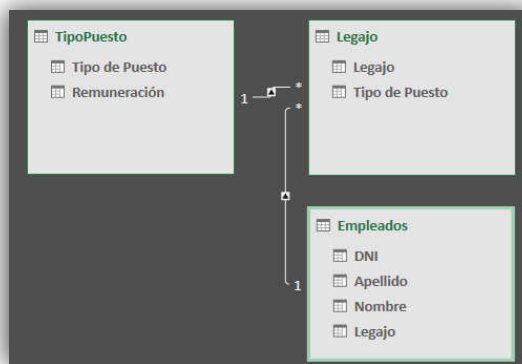
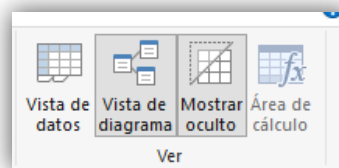
	TipoPuesto	Remuneración
3	Administrativo 2	25000
4	Administrativo 3	20000
5	Operario 1	28000
6	Operario 2	24000
7	Operario 3	20000



Segundo paso: Activar la vista de Diagrama y Establecer las relaciones. Desde la pestaña **Inicio** de la ventana de PowerPivot, se activará la vista de Diagrama y desde la pestaña **Diseñar** se utilizará la opción **Crear relación**.

Remuneración	Tipo de Puesto
30000	Administrativo 1
25000	Administrativo 2
20000	Administrativo 3
28000	Operario 1
24000	Operario 2

Legajo	Tipo de Puesto
1235	Administrativo 1
1754	Administrativo 2
1893	Administrativo 2
2354	Operario 1
2453	Operario 2

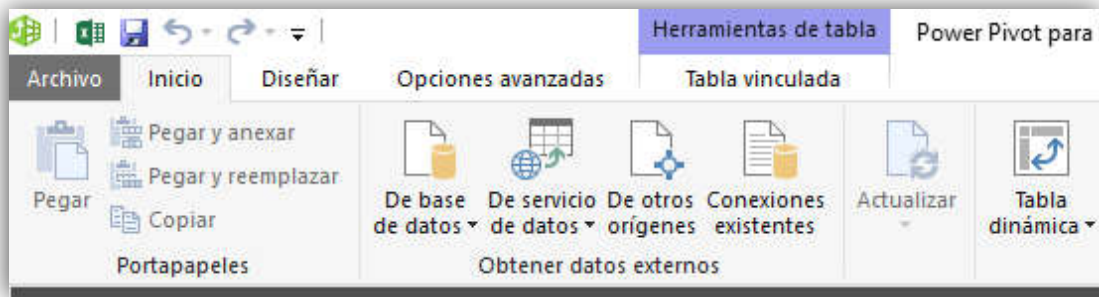


Tercer paso: Generar la tabla dinámica. Desde la pestaña **Inicio** de la ventana de PowerPivot, se activa la opción **Tabla dinámica**

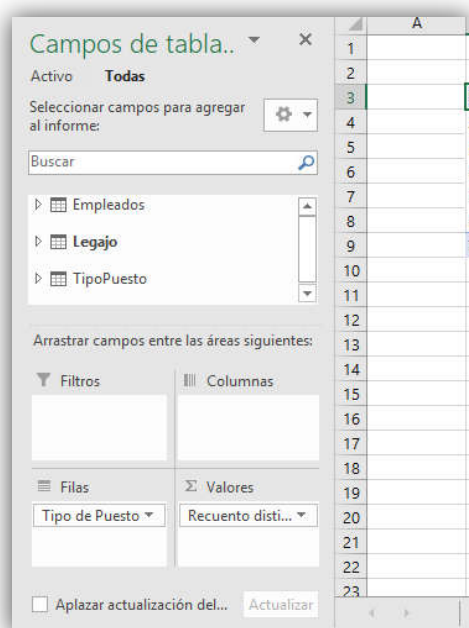


TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



La tabla dinámica resultante ahora permite relacionar los datos de las distintas tablas originales.



Resulta obvio que el potencial de la herramienta PowerPivot no se agota en este sencillo ejemplo. Aunque la gran mayoría de los usuarios encontrarán esta herramienta algo críptica, es realmente sencillo utilizarla para crear reportes de tablas dinámicas que incluyan varias fuentes de información.

Anteriormente se ha comentado la capacidad que tiene PowerPivot para trabajar con medidas mediante la utilización del lenguaje DAX (Data Analysis Expressions), el cual está basado en fórmulas que permiten definir cálculos personalizados.

DAX está compuesto por una biblioteca de funciones y operadores que se pueden combinar para generar fórmulas y expresiones. Se caracteriza por su simplicidad en la sintaxis y su similitud a las fórmulas de Excel, incluso ha heredado muchas funciones con el mismo nombre y misma funcionalidad (excepto pequeñas variaciones en contadas excepciones)



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)

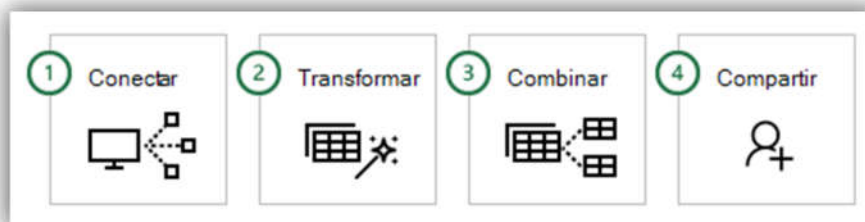


	D.	Apellido	Nombre	Le...	Columna calculada 1	Agregar columna
1	29061...	Perez	Joaquín	2354	Perez Joaquín	
2	31803...	Sosa	Eleonora	1235	Sosa Eleonora	
3	38898...	Gutierrez	José	1754	Gutierrez José	
4	30660...	Mando	Pedro	2453	Mando Pedro	
5	24608...	Garay	María	2653	Garay María	
6	29339...	Fernandez	Martín	2732	Fernandez Martín	
7	32448...	Lijo	Fernando	1893	Lijo Fernando	

3.10. Power Query (Obtener y Transformar desde Excel 2016)

Obtener y transformar en Excel 2016 (Power Query), puede buscar datos desde distintas fuentes, realizar conexiones y también realizar tareas de edición de dichos datos (por ejemplo, quitar una columna, cambiar un tipo de datos o combinar tablas). Estos datos pueden ser compartidos a otros usuarios, o modelizarlos para crear informes dinámicos a través de PowerPivot.

La metodología de Power Query se puede entender con la siguiente imagen:



Conectar: Se refiere a vincular los datos existentes, en una fuente externa o local, a un libro de Excel.

Transformar: Se refiere a la modificación de la estructura de datos en el libro de Excel, sin modificar dicha estructura en la fuente original de datos.

Combinar: Se refiere a la posibilidad de utilizar distintas fuentes de datos para alimentar la base vinculada en el libro de Excel.

Compartir: Se refiere a la utilización de la consulta una vez finalizada, ésta se puede guardar, compartir o utilizarla para generar un informe dinámico.

3.10.1. Utilizando Power Query

En el siguiente ejemplo se procederá a vincular datos existentes en tablas ubicadas en distintas hojas de un libro a través de la herramienta Power Query.

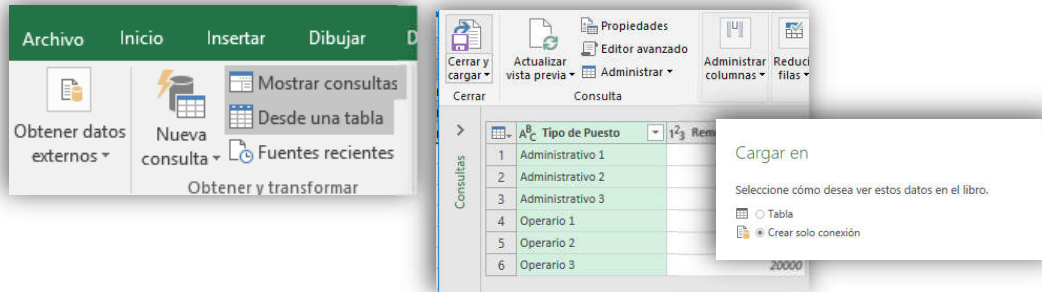


TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



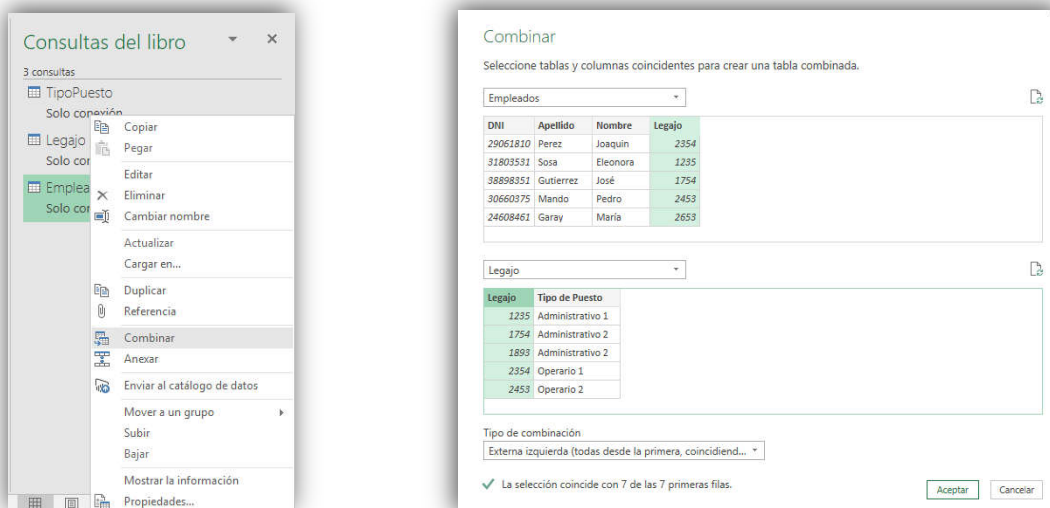
Primer paso: Realizar las consultas. Activando la Opción *Desde una tabla*, se procederá a cargar las consultas (una consulta por tabla)



Al seleccionar la opción *Cerrar y cargar*, Excel almacenará la consulta, mostrando todas las realizadas en un panel ubicado, por defecto, en el sector derecho de la planilla de cálculo. Además, generará una hoja por cada consulta. Todo lo que se modifique en esta hoja creada no afectará a la fuente original de los datos.

Si se seleccionara la opción *Cerrar y Cargar en...* → *Sólo conexión* (es la que efectivamente se utilizará en el ejemplo), no se crearán nuevas hojas para cada consulta, lo cual resulta recomendable si se tratan de datos locales.

Segundo Paso: Elegir las tablas a vincular. En este paso elegiremos desde el panel *Consultas del libro*, la tabla destino para combinarla con la tabla que contiene los datos que se buscan relacionar. En el ejemplo, se elegirá la tabla *Empleados*, combinándola con la tabla *Legajo*, ya que se quiere saber qué puesto tiene cada persona.

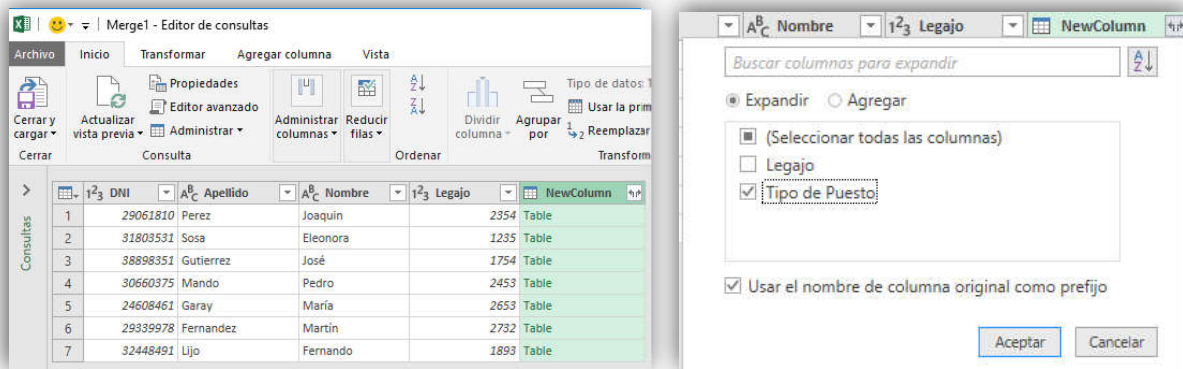


Tercer Paso: Seleccionar la columna adicional de la nueva consulta. Luego de aceptar, se generará una nueva consulta denominada *Merge1* y automáticamente se agregará una nueva columna denominada *NewColumn*. Al hacer clic en el botón al costado del nombre, podremos seleccionar que columna de la tabla elegida se mostrará en esta nueva consulta. En este caso se selecciona *Tipo de Puesto*.




TÓPICOS DE EXCEL

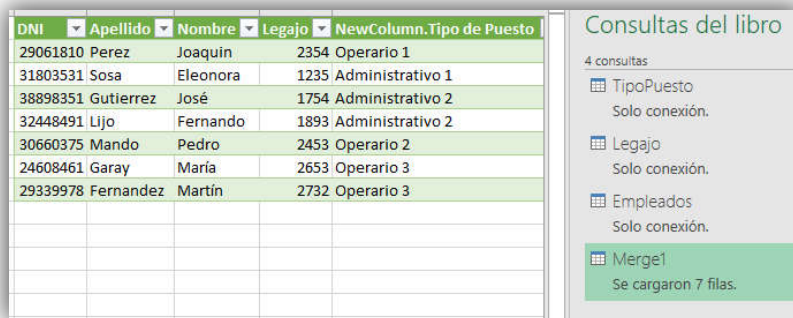
(Material teórico gratuito 2019)



Se puede apreciar que la consulta Merge1 sumó una nueva columna, con el tipo de cargo de cada empleado. Algo que no aparecía en las tablas originales.

	DNI	Apellido	Nombre	Legajo	NewColumn.Tipo de Puesto
1	29061810	Perez	Joaquin	2354	Operario 1
2	31803531	Sosa	Eleonora	1235	Administrativo 1
3	38898351	Gutierrez	José	1754	Administrativo 2
4	30660375	Mando	Pedro	2453	Operario 2
5	24608461	Garay	María	2653	Operario 3
6	29339978	Fernandez	Martín	2732	Operario 3
7	32448491	Lijo	Fernando	1893	Administrativo 2

Cuarto Paso: Guardar la consulta. Haciendo clic en la opción  se agregará la consulta al panel Consultas del libro además de crearse una nueva hoja con la tabla.



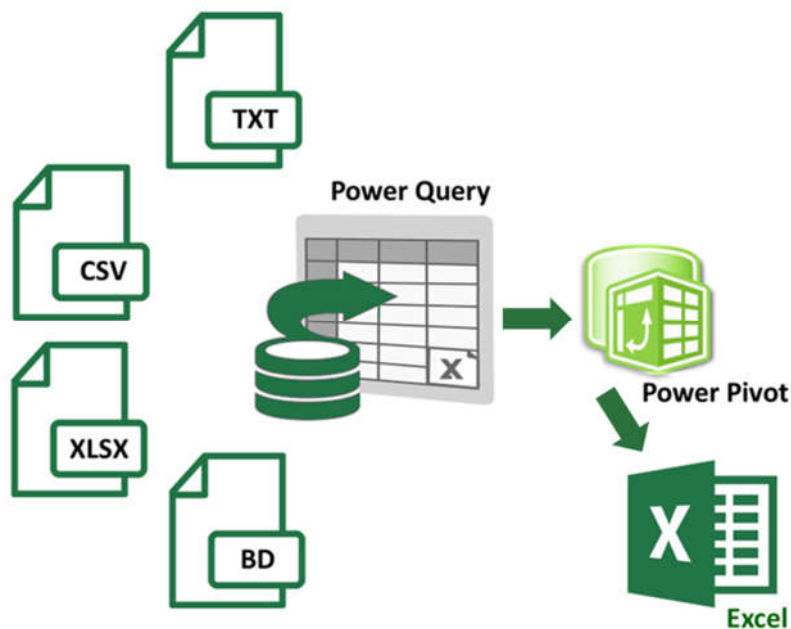
Una consulta de Power Query puede ser utilizada en el modelo de datos que alimenta a Power Pivot. Muchos usuarios de Power Pivot desconocen la versatilidad de Power Query, principalmente porque la segunda es una herramienta que se lanzó con posterioridad a la primera.

Para entender la relación ambas herramientas, es importante saber que ambas comparten varias características en común, ya que fueron las que en un principio se incorporaron a Power Pivot y que se perfeccionaron al lanzarse Power Query.



Power Query, al estar diseñado como una herramienta **ETL** (Extract, Transform and Load; en castellano Extraer, Transformar y Cargar) permite capturar datos desde un conjunto mucho más amplio de fuentes que Power Pivot.

A continuación, se presenta un esquema que grafica la utilización de ambas herramientas para la generación de informes dinámicos a partir de datos masivos, desde diferentes fuentes de datos.



3.11. Power Maps (Mapas 3D desde Excel 2016)

Entre las herramientas Business Intelligence de Excel se incorporó la opción de generar gráficos geolocalizados, con segmentación de fechas y horas a partir de un modelo de datos. La opción en cuestión se denomina Power Maps (Mapas 3D desde Excel 2016).

Power Maps puede ser utilizado para explorar datos mediante Mapas de Microsoft BING MAPS, o con cualquier mapa personalizado a partir de una imagen. También se puede utilizar para evaluar la evolución temporal de los datos a partir de los llamados “paseos”.

Al ver los datos en un espacio geográfico y lo largo del tiempo, se puede extraer nuevas conclusiones a partir de los datos o visualizar la información de forma más evidente.

3.11.1. Generar un Power Map

El primer paso necesario para generar un Power Map es el contar con una tabla de datos ubicables geográficamente. Esta ubicación puede tener distintos niveles de desagregación, desde una ubicación puntual por coordenadas hasta continental.



TÓPICOS DE EXCEL

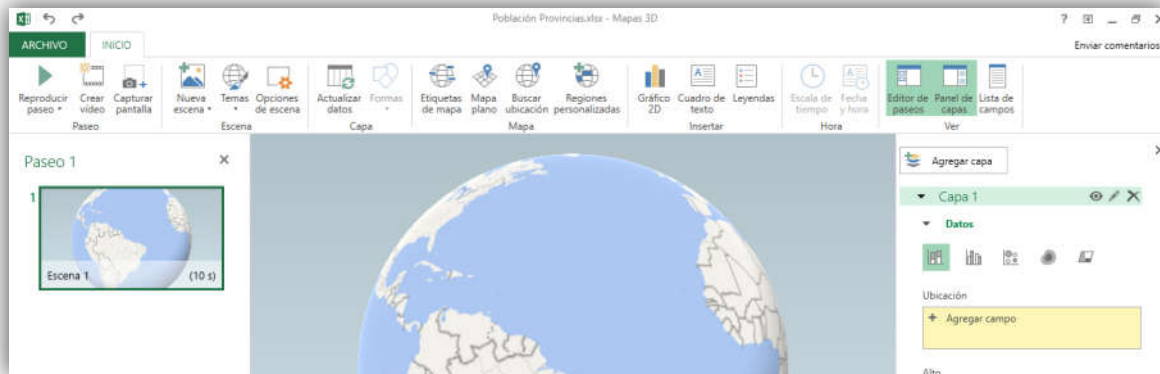
(Material teórico gratuito 2019)



En el ejemplo se dispone de una lista con la cantidad de habitantes por provincia, incluyendo a la Ciudad Autónoma de Buenos Aires.

	A	B
1	ZONA	POBLACION
2	Ciudad de Buenos Aires	2.890.151
3	Buenos Aires	15.625.084
4	Catamarca	367.828
5	Chaco	1.055.259
6	Chubut	509.108
7	Córdoba	3.308.876
8	Corrientes	992.595
9	Entre Ríos	1.235.994
10	Formosa	530.162
11	Jujuy	673.307

Para generar un Power Map, se utiliza la opción **INSERTAR → MAPA 3D**, con lo cual se activará la ventana de esta herramienta.



La ventana de Power Maps, se compone de varias opciones y paneles, pudiendo presentarse como algo intimidante al principio. Sin embargo, resulta muy sencillo generar los gráficos geolocalizados una vez entendida la metodología de utilización de tablas dinámicas en Excel.

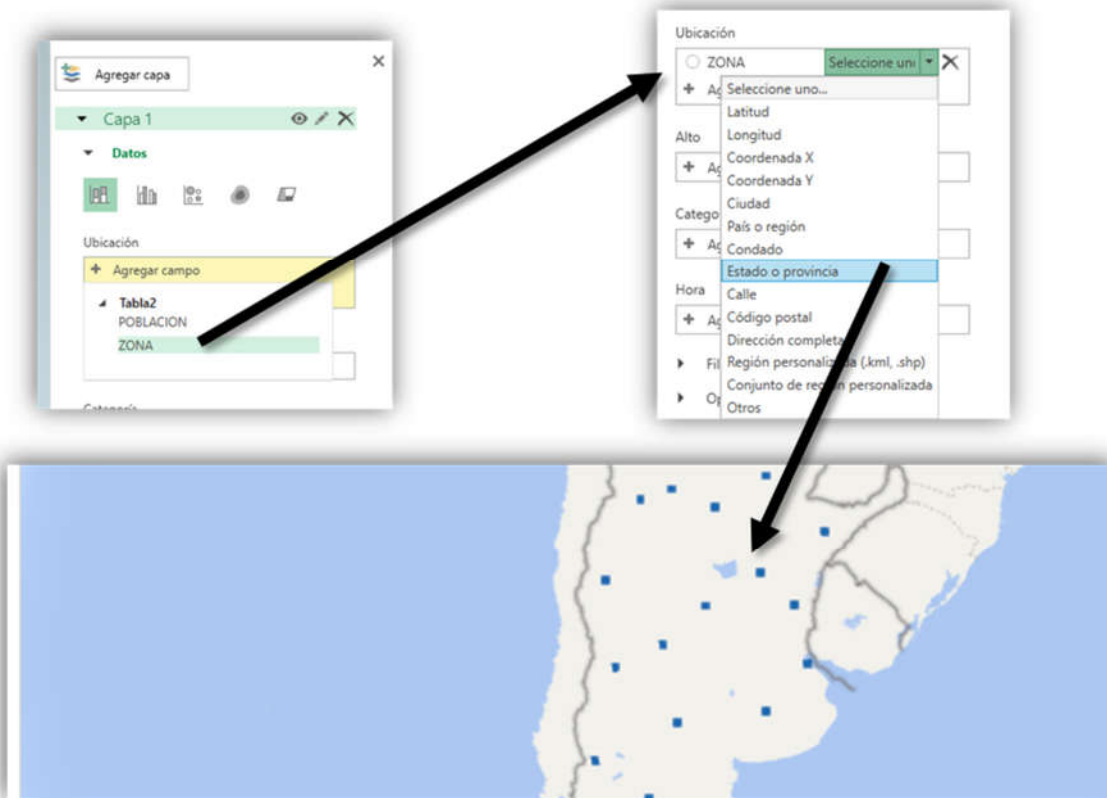
Primer Paso: Una vez abierta la ventana de Power Maps el primer paso es agregar el campo de ubicación. En el ejemplo se utilizará el campo correspondiente a la columna ZONA.

Segundo Paso: Seleccionar tipo de dato. De esta manera, se le indica a Excel que tipo de localización buscar en el mapa. De inmediato, Excel intentará encontrar la correspondencia entre la entrada del campo y su correspondiente ubicación en el mapa.

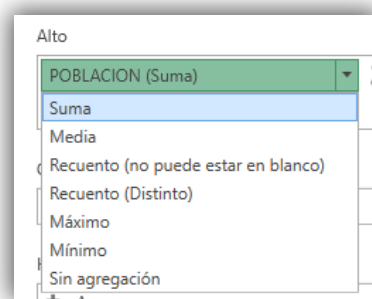
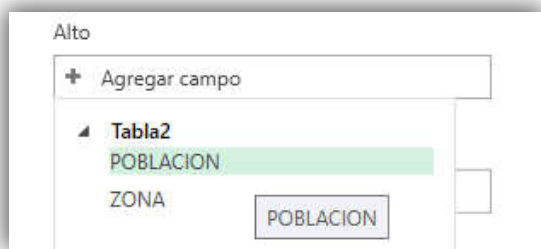


TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



Tercer paso: Seleccionar los valores correspondientes a cada zona. En el ejemplo, se tiene cantidad de habitantes por provincia, con lo cual se utilizará en campo POBLACIÓN, eligiendo la opción SUMA entre los distintos tipos de funciones que nos ofrece Excel.

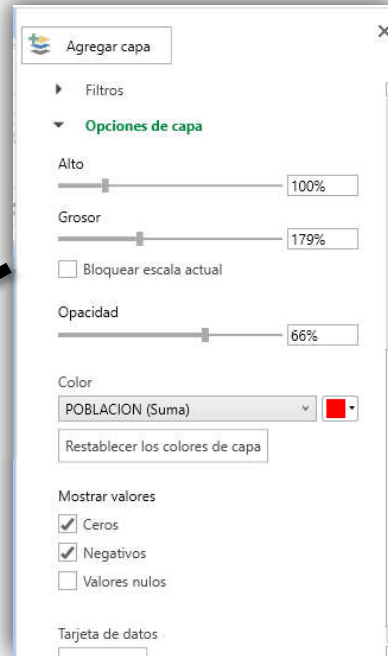
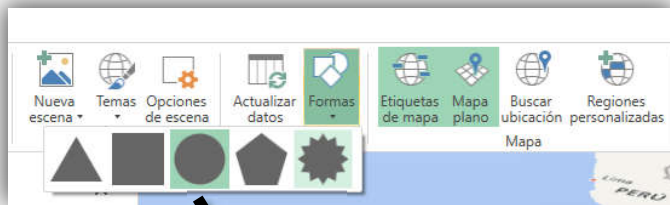


Cuarto Paso: Personalizar el mapa obtenido mediante las opciones de la herramienta. Power Maps permite un alto nivel de personalización, mediante la barra de herramientas y a través de las opciones de capa.

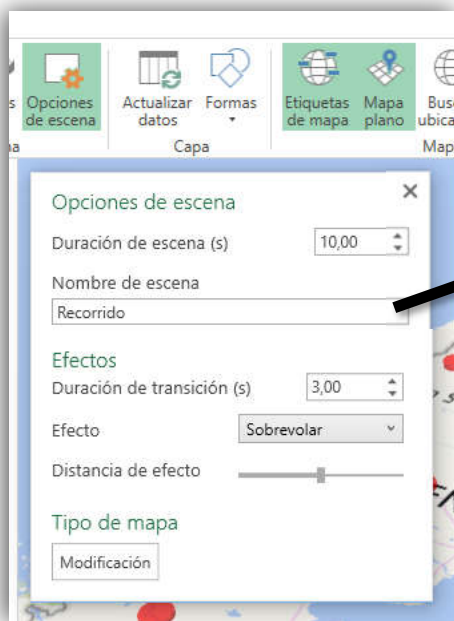


TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



Quinto paso (opcional): Agregar información complementaria al mapa. Power maps permite crear recorridos del mapa obtenido, a partir de las Opciones de Escena. Estas opciones permiten un impacto visual muy interesante en las presentaciones de los datos con los cuales se estén trabajando.

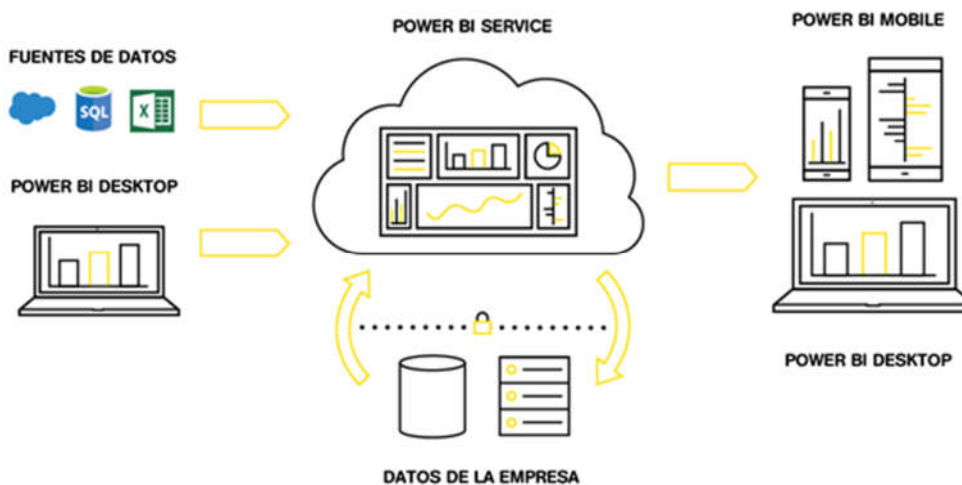




3.12. Power BI. Más allá de Excel

Power BI es un conjunto de aplicaciones que permite analizar datos y compartir información a partir de herramientas online y de aplicativos tanto para computadoras personales (Power BI Desktop), como para dispositivos móviles (Power BI Mobile).

El servicio Power BI puede unificar múltiples fuentes de datos, ya sea estén alojados en servicios cloud computing o localmente.



Power BI Desktop permite al usuario la creación de consultas, conexión de datos desde distintas fuentes y la confección de informes que se pueden compartir de manera sencilla con distintos usuarios.

Es un programa independiente a Excel que integra la gran mayoría de las herramientas Business Intelligence vistas en los apartados anteriores. Con este software el usuario puede conectarse a datos de distintas fuentes, crear modelos de datos con posteriores consultas y finalmente generar informes dinámicos de una manera extremadamente sencilla.

Esta herramienta transforma los datos en objetos enriquecidos visualmente que puede recopilar y organizar la información orientada al análisis BIG DATA.

La diferencia entre Power BI y Excel es que el primero fue pensado y diseñado para la utilización en inteligencia de negocios a partir de una cuenta online que facilita la utilización de las más recientes herramientas para el análisis de datos; mientras que el segundo dispone de dichas herramientas como un agregado posterior.

Es importante, para poder comprender cabalmente los conceptos asociados al paradigma Power BI que éste está pensado como un servicio multiplataforma basado en la actualización de datos en tiempo real a través de un conjunto de herramientas que actúan en conjunto para brindar una solución completa en el análisis de datos masivos.



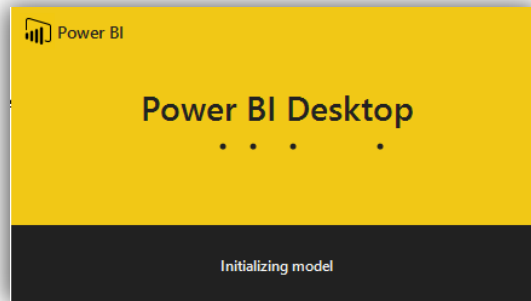
TÓPICOS DE EXCEL

(Material teórico gratuito 2019)

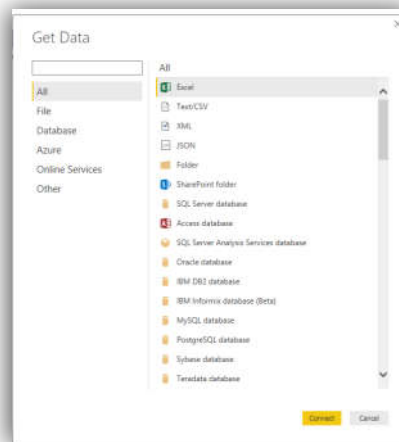
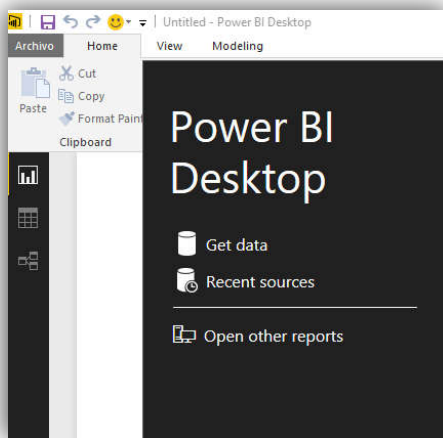


3.12.1. Utilizando Power BI Desktop

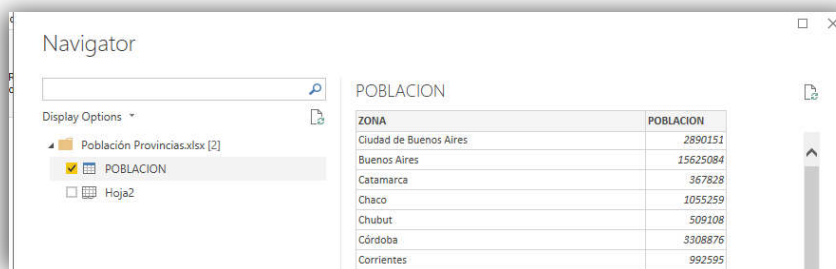
Supuesta la anterior instalación de Power BI Desktop (hay que recordar que es un programa independiente a Excel), se procederá a realizar un sencillo informe gráfico tomando como base el ejemplo anterior de cantidad de habitantes por provincias de Argentina.



Primer paso: Obtener datos o abrir informes anteriores. Con la opción Get data, se procederá a importar los datos provenientes, en este caso, del archivo de Excel correspondiente.



Segundo paso: Seleccionar los campos a importar. En el ejemplo se seleccionarán los campos correspondientes a la hoja Población.



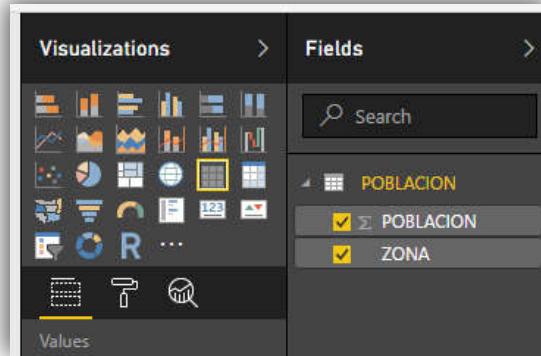


TÓPICOS DE EXCEL

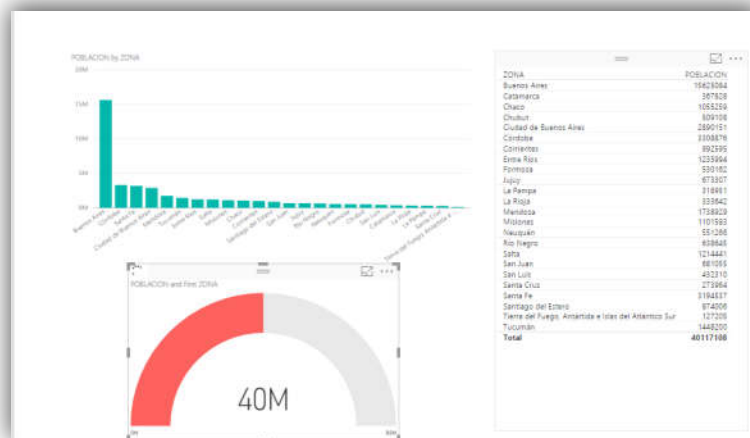
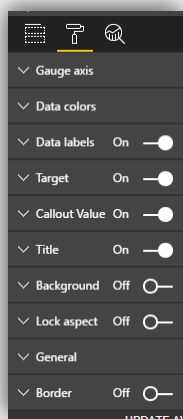
(Material teórico gratuito 2019)



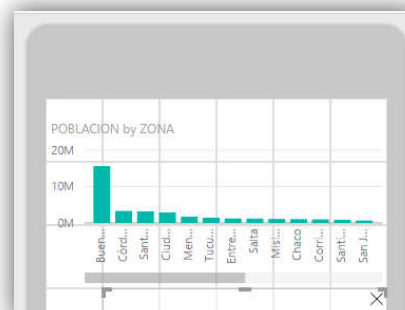
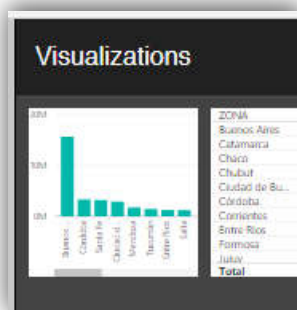
Tercer paso: Elegir, entre las opciones de gráficos e informes, cual utilizar. Cada gráfico o tabla se insertará como un objeto independiente de los demás, pero con dependencia de la fuente de datos.



Cuarto paso: Dar el formato final deseado y ubicar, dentro del lienzo, los objetos para una mejor visualización. La versatilidad de Power BI Desktop con su sistema de objetos, permite un muy alto nivel de personalización.



Quinto paso (Opcional): Configurar el lienzo para la visualización en dispositivos móviles. Es posible configurar la visualización en dispositivos móviles de los datos resumidos por Power BI Desktop.



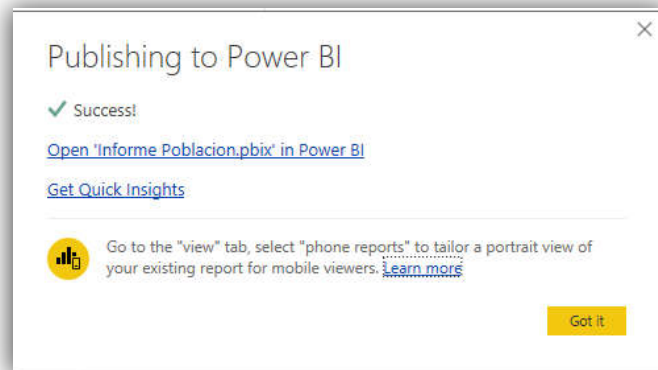
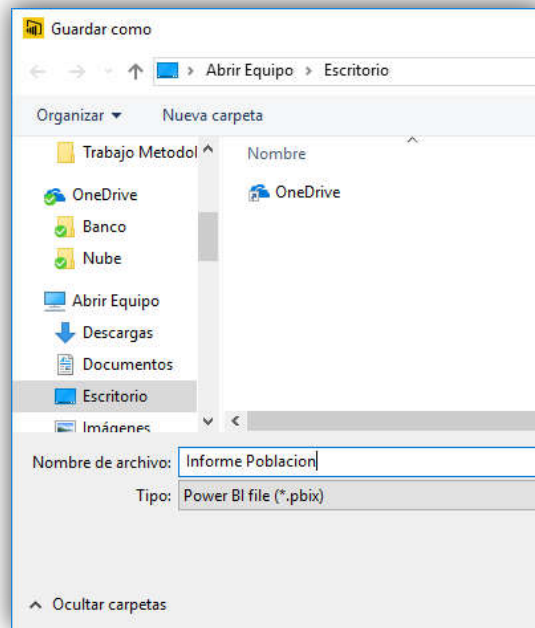
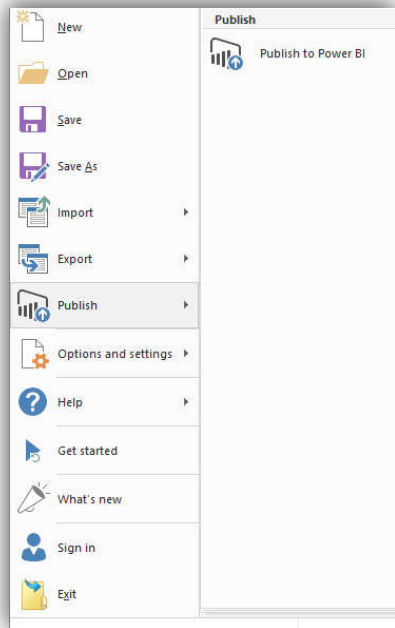


TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



Sexto paso: Exportar el informe realizado. Con la opción publicar, es posible compartir el informe con otros dispositivos y usuarios. Los informes, se guardan localmente con extensión .pbix





TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



4. Grabación de Macros y Programación VBA

4.1. Introducción

Suele suceder que, en el ejercicio de la profesión aplicando Excel, nos enfrentemos a tareas repetitivas que debamos realizar manualmente.

Por otro lado, puede ocurrir que Excel no tenga integrada determinada función o característica que necesitemos para cumplir con cierta tarea.

Es en estos casos en donde debemos plantearnos el aprender a utilizar la grabación de macros y a programar nuestras funciones personalizadas con el lenguaje integrado de Excel, **Visual Basic for Applications (VBA)**. Dominando estas dos herramientas, podemos expandir casi infinitamente las capacidades propias de Excel.

Al dominar el lenguaje **VBA**, además, podremos escribir código que le diga a Excel que realice tareas que no puedan ser grabadas. Por ejemplo, podremos escribir procedimientos que muestren cuadros de diálogo personalizados, interactuar con otros programas en nuestra computadora, o programar nuestras propias aplicaciones que funcionen en el entorno Excel.

El objetivo de este módulo es introducirnos en la programación de macros y funciones sencillas, como así también, el ayudarnos a entender la estructura principal de un código VBA de nivel intermedio.

4.2. Grabación de macros en Excel con VBA

En este punto presentaremos la **grabación de macros de Visual Basic para Aplicaciones (VBA)**: componente clave para los usuarios que desean personalizar y automatizar Excel. También abordaremos algunos aspectos de la optimización del código generado.

En computación, una **macro** (diminutivo de macro instrucción), es una serie de instrucciones que se almacenan para que éstas se puedan ejecutar de manera secuencial mediante una sola llamada u orden de ejecución.

Realizando una analogía entre Excel y un teléfono, **nosotros podemos marcar el número correspondiente todas las veces que queremos llamar** (realizar manualmente las tareas repetitivas en Excel) **o almacenar dicho número en la memoria del teléfono de modo que solamente debemos presionar un botón para realizar la llamada** (utilizar una macro para automatizar dichas tareas).

En Excel, las macros **nos permiten trabajar de manera más eficiente y con menos errores**, sobre todo en las tareas repetitivas que requieren diversos pasos que puedan ser identificables.

Podemos crear una macro, por ejemplo, para formatear e imprimir el informe de ventas de fin de mes.

Con el uso de macros, podremos aumentar de forma impensada nuestra productividad en Excel.

No es necesario ser un experto en Excel para crear y usar macros simples de VBA. Una vez que se entienden algunos principios básicos, resulta muy sencillo utilizar la herramienta grabación de macros de VBA. Cuando se ejecute la macro grabada, Excel realizará las acciones de nuevo.



4.2.1. Traduciendo nuestras acciones en código VBA: la grabadora de macros

En esta sección, describiremos los pasos básicos que deberemos seguir para realizar la grabación de una macro de VBA. **La herramienta que nos ofrece Excel para llevar a cabo la memorización de las secuencias y comandos se denomina Grabadora de macros.**

Esta herramienta **traducirá automáticamente nuestras acciones al código VBA**, que Excel puede leer e interpretar, para replicar nuestras acciones cuando activemos la macro.

En la mayoría de los casos, **basta simplemente con grabar la macro**, para que Excel memorice nuestras secuencias de acciones y comandos. Cuando deseemos repetirlas, será suficiente con invocar la macro correspondiente, **sin importarnos cuál es el código VBA que Excel creó.**

Resulta recomendable que podamos, al menos, entender qué dice el código. De esta manera podremos optimizar y acelerar las tareas memorizadas por Excel.

4.2.2. Grabando nuestra primera macro

Como dijimos anteriormente, la grabadora de macros de Excel traduce nuestras acciones y comandos en código VBA.

A) Podemos iniciar la grabación haciendo clic en el icono Grabar macro en el extremo inferior izquierdo de la barra de estado). Figura 4.1

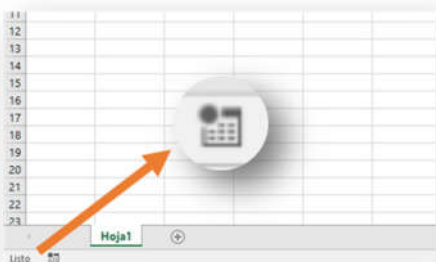


Figura 4.1. La grabadora de macros se puede activar desde el icono ubicado en el sector inferior derecho de la ventana de Excel.

Al activar esta opción, Excel mostrará la **ventana Grabar Macro**. (Figura 4.2)

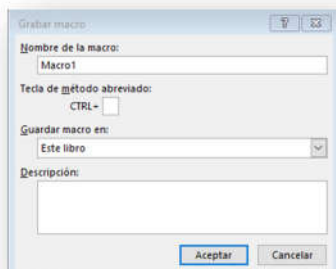


Figura 4.2. La ventana Grabar macro nos permite personalizar el nombre, el atajo del teclado, la ubicación de la macro y agregarle una descripción.

El cuadro de diálogo **Grabar macro** presenta cuatro campos:



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



Nombre de la macro: Excel propone nombres genéricos, como Macro1, Macro2. Podemos cambiarlo, pero deben empezar con una letra y no pueden contener espacios, ni signos de puntuación (se acepta el guion bajo “_”).

Tecla de método abreviado: Podemos asignar un atajo del teclado para invocar a la macro.

El atajo del teclado siempre usa la tecla **Ctrl**. Opcionalmente, también podremos presionar la tecla **MAYUSC** (Shift) al ingresar una letra. Por ejemplo, si presionamos **MAYUSC** (Shift) mientras ingresamos la letra **h**, el atajo del teclado para llamar a nuestra macro será **Ctrl + Shift + h**

Recordar: Las teclas de acceso directo asignadas a las macros tienen prioridad sobre las teclas de método abreviado incorporadas. Por ejemplo, si asignamos **Ctrl + t** a una macro, no podremos usar dicha combinación de teclas para crear una tabla.


Guardar Macro en: Aquí podremos elegir donde se guardará la macro. Tendremos tres opciones:

Este libro: Guarda la macro en el libro en donde estemos trabajando. La macro será accesible solamente en el archivo que generemos (con la extensión .xslm)

Libro de macros personal: Guarda la macro en un archivo oculto que se inicia junto con Excel. La macro será accesible en todos los libros de Excel que utilicemos en nuestra computadora.

Libro nuevo: Excel creará un libro nuevo y almacenará en código de la macro en éste.

Descripción: Podremos ingresar una descripción de la macro (no es obligatorio que lo hagamos).

B) Al presionar  Excel comenzará a registrar nuestras acciones, convirtiéndolas a código VBA. Para detener la grabación de la macro, podemos hacer clic en el icono **STOP**, ubicado en el mismo lugar que el icono de la grabadora de macros. (Figura 4.3)

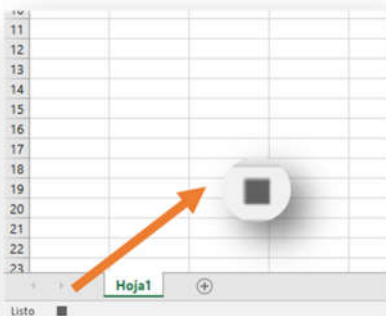


Figura 4.3. Cuando nos encontremos grabando una macro, el icono correspondiente cambiará a un símbolo de **STOP**. Haciendo clic en éste, podemos detener la grabación de macros.

Si bien la anterior manera es la más rápida, no nos permite una opción muy importante que veremos a continuación. Para poder activar dicha opción necesitaremos algún método alternativo.

Una forma alternativa de invocar a la grabadora de macros es a partir de la pestaña **VISTA -> Macros -> Grabar Macros**. (Figura 4.4)

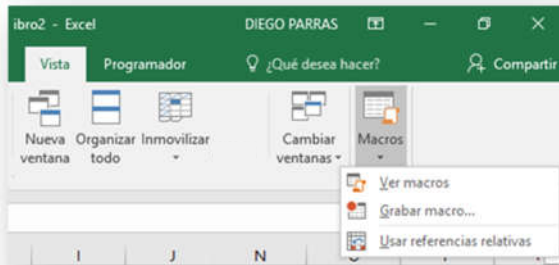


Figura 4.4. Desde la pestaña VISTA podremos acceder a varias opciones de las macros.

Es importante, antes de grabar nuestra primera macro, el mencionar la opción **“Usar referencias relativas”**. (Figura 4.5)

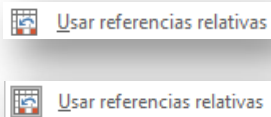


Figura 4.5. Si esta opción está desactivada, la macro se ejecutará **EXACTAMENTE en las celdas en donde las grabamos**. Si la opción está activada, la macro se ejecutará **utilizando a la celda activa como la posición INICIAL RELATIVA**.

4.2.3. Ejemplo (Referencias Absolutas)

Vamos a grabar nuestra primera macro, **la misma consiste en darle un formato de celda color naranja a las celdas A1, B1 y C1**. En este caso NO utilizaremos referencias relativas, con lo cual debemos desactivar dicha opción, seguiremos los siguientes pasos:

- 1) En la pestaña **Vista -> Macros**, verificamos que la opción **Referencias Relativas** esté desactivada.
- 2) En la pestaña **Vista -> Macros**, seleccionaremos la opción **Grabar Macro** (también podemos invocar a la grabadora de macros desde el icono mencionado anteriormente).
- 3) En la ventana, completaremos los campos con la siguiente información. (figura 4.6):

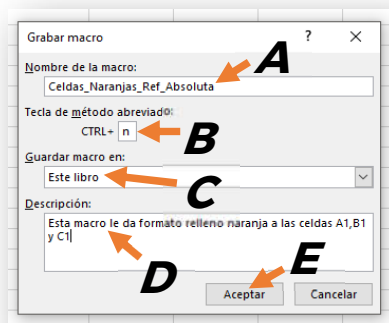


Figura 4.6. **Campo A:** Celdas_Naranjas_Ref_Absoluta. **Campo B:** Ctrl + n, **Campo C:** Este libro, **Campo D:** Esta macro le da formato relleno naranja a las celdas A1, B1 y C1

- 4) Luego de presionar aceptar, Excel entrará en modo de grabación de macros, y todo lo que hagamos será traducido a código VBA. Es importante aclarar que el tiempo que tardemos en grabar la macro no será tenido



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



en cuenta en la creación del código, pero si la cantidad de pulsaciones del mouse o acciones que realicemos (**¡Debemos optimizar la cantidad de pasos a seguir!**).

- 5) **En este paso realizaremos las acciones que serán grabadas en el código VBA**, es decir: seleccionaremos las celdas A1, B1 y C1 y le daremos un formato de relleno color naranja, una vez hecho esto, hacemos clic en el botón STOP descrito anteriormente.
- 6) Cada vez que presionemos la combinación **CTRL+n**, Excel le dará formato relleno naranja a las celdas A1, B1, y C1 de la hoja que tengamos activa.

Hemos grabado nuestra primera macro. **¿Exactamente qué fue lo que memorizó Excel?**

4.2.4. Examinando el Código de una macro

Ya sabemos que, al grabar macros en Excel, nuestras acciones, secuencias y comandos realizados, se traducirán a código VBA que luego, al activar la macro grabada, Excel repetirá siguiendo las directivas de este código memorizado.

Nosotros podemos ingresar al código generado para analizarlo y, eventualmente, modificarlo para poder optimizarlo. Las razones para hacer esto son varias.

La razón principal se da porque, al transcribir en el código VBA las acciones y comandos realizados, Excel utiliza bloques de código preestablecidos que muchas veces resultan redundantes, y agregan complejidad innecesaria a la interpretación y ejecución del código.

Una de las maneras más sencillas de explorar el código generado por una Macro es a partir de la pestaña **Vista** -> **Macros** -> **Ver Macros**. Una vez abierta la ventana Macros, hacemos clic en Modificar. (Figura 4.7)

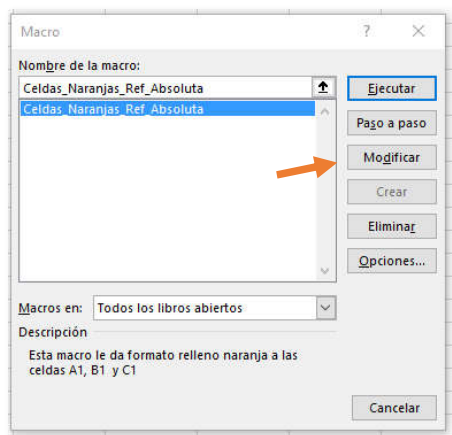


Figura 4.7. La ventana de macros se puede activar también con la combinación de teclas **Alt + F8**.



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



Se mostrará una ventana independiente a Excel denominada **Editor de Visual Basic (VBE)**. El **VBE** es el entorno para programar en **Visual Basic para Aplicaciones (VBA)**. Su interfaz está compuesta por varias partes (figura 4.8), que describiremos en clases.

En este apartado nos enfocaremos en la ventana más grande, denominada Área de trabajo.

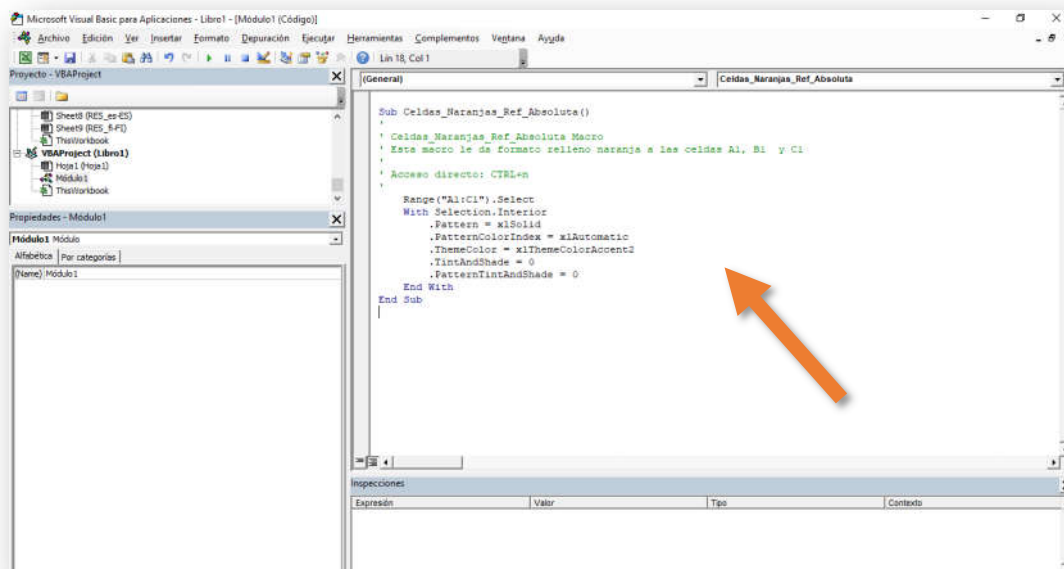


Figura 4.8. En esta ventana se volcará todo el código memorizado por la macro. Es importante destacar que nosotros podríamos hacer el ingreso manual del mismo y éste arrojaría el mismo resultado que la macro.

El código resultante para nuestra macro anterior es el siguiente:

```
Sub Celdas_Naranjas_Ref_Absoluta()  
'  
' Celdas_Naranjas_Ref_Absoluta Macro  
' Esta macro le da formato relleno naranja a las celdas A1, B1 y C1  
'  
' Acceso directo: CTRL+n  
'  
    Range("A1:C1").Select  
    With Selection.Interior  
        .Pattern = xlSolid  
        .PatternColorIndex = xlAutomatic  
        .ThemeColor = xlThemeColorAccent2  
        .TintAndShade = 0  
        .PatternTintAndShade = 0  
    End With  
End Sub
```



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



El lenguaje Visual Basic for Applications (**VBA**) es un lenguaje orientado a objetos al igual que su hermano mayor Visual Basic (**VB**).

Si bien ambos lenguajes son casi idénticos en lo que respecta a la construcción del código; **VB** permite generar programas independientes que funcionan en el entorno Windows, mientras que **VBA** necesita de un programa anfitrión y los “programas” que genere su código funcionarán dentro de ese programa principal.

Vamos a realizar este primer análisis de una manera bastante informal, separando el código en bloques. Para quien entienda el idioma inglés, resulta bastante simple la interpretación de las expresiones del código.

Bloque 1: Generar una Subrutina. Como un primer paso **VBA** necesita saber qué queremos hacer. Cuando queremos una secuencia de instrucciones que se sucedan a partir de un código **VBA**, necesitamos crear una Subrutina.

Las subrutinas son tipos de procedimientos que contienen determinadas instrucciones (más adelante lo veremos con mayor profundidad). La subrutina tendrá el nombre de la macro.

Excel Graba lo siguiente.

```
Sub Celdas_Naranjas_Ref_Absoluta()
```

```
End sub
```

Bloque 2: Comentarios. El segundo paso es (optativamente) grabar todos los comentarios que queramos hacer en el código (los comentarios figurarán en verde y serán antecidos por un apostrofe). Este texto auxiliar no será tenido en cuenta por Excel a la hora de ejecutar el código.

La opción de agregar apostrofes delante de una línea es muy utilizada por los programadores de **VBA** para testear y optimizar código sin necesidad de eliminar dicha línea.

```
Sub Celdas_Naranjas_Ref_Absoluta()
```

```
'
```

```
' Celdas_Naranjas_Ref_Absoluta Macro
```

```
' Esta macro le da formato relleno naranja a las celdas A1, B1 y C1
```

```
'
```

```
' Acceso directo: CTRL+n
```

```
'
```

```
End sub
```



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



Bloque 3: Primera instrucción. En nuestra macro, la primera instrucción era seleccionar las celdas A1, B1 y C1, es exactamente lo que se traduce en el código.

```
Sub Celdas_Naranjas_Ref_Absoluta()  
'  
' Celdas_Naranjas_Ref_Absoluta Macro  
' Esta macro le da formato relleno naranja a las celdas A1, B1 y C1  
'  
' Acceso directo: CTRL+n  
'  
    Range("A1:C1").Select  
  
End sub
```

Notemos que, estamos completando el código generado por la macro, paso a paso a fines de poder interpretar la metodología interna de Excel en lo que respecta a la grabación de una macro.

Bloque 4: Lote de instrucciones. La siguiente instrucción de nuestra macro, consiste en darle un formato color naranja. En esta parte veremos una estructura, habitualmente utilizada por Excel al grabar macros, que permite generar un lote de instrucciones relacionadas.

La estructura **WITH** nos permite el no tener que repetir instrucciones distintas (pero que estén relacionadas por el mismo objeto).

```
Sub Celdas_Naranjas_Ref_Absoluta()  
'  
' Celdas_Naranjas_Ref_Absoluta Macro  
' Esta macro le da formato relleno naranja a las celdas A1, B1 y C1  
'  
' Acceso directo: CTRL+n  
'  
    Range("A1:C1").Select  
    With Selection.Interior  
        .Pattern = xlSolid  
        .PatternColorIndex = xlAutomatic  
        .ThemeColor = xlThemeColorAccent2  
        .TintAndShade = 0  
        .PatternTintAndShade = 0  
    End With  
End Sub
```



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



La estructura en negrita es equivalente a la siguiente (El objeto repetido es *Selection.Interior*):

```
Selection.Interior.Pattern = xlSolid  
Selection.Interior.PatternColorIndex = xlAutomatic  
Selection.Interior.ThemeColor = xlThemeColorAccent2  
Selection.Interior.TintAndShade = 0  
Selection.Interior.PatternTintAndShade = 0
```

En términos coloquiales, Excel está memorizando:

```
Selection.Interior.Pattern = xlSolid
```

(Démosle una trama de relleno sólido al interior de las celdas seleccionadas, ver figura 4.9)



Figura 4.9. El código VBA está replicando la opción *Estilo de Trama* de la pestaña *Relleno* de la ventana *Formato de Celdas*

```
Selection.Interior.PatternColorIndex = xlAutomatic
```

(Démosle el color preestablecido de la paleta de colores usada en Excel a la trama, ver figura 4.10)

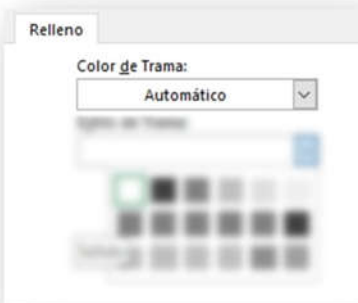


Figura 4.10. El código VBA está replicando la opción *Color de Trama* de la pestaña *Relleno* de la ventana *Formato de Celdas*

```
Selection.Interior.ThemeColor = xlThemeColorAccent2
```

(Démosle al interior de la celda, el color "Énfasis 2", ver figura 4.11)



Figura 4.11. El código VBA está replicando la opción correspondiente a la ventana Colores del tema.



```
Selection.Interior.TintAndShade = 0
```

(Dejemos el color como está, no lo oscurezcamos min = -1, ni lo aclaremos máx = 1)

```
Selection.Interior.PatternTintAndShade = 0
```

(Dejemos el color de la trama del relleno como está, sin oscurecerla ni aclararla, ídem anterior)

Como dijimos, con la estructura **WITH**, evitamos tener que definir constantemente el objeto selection.interior con sus diferentes propiedades, solamente lo hacemos una vez, y luego (dentro del **WITH**) definimos las distintas propiedades de dicho objeto, con sus valores correspondientes. Cuando dejamos de hacer referencia al objeto en cuestión, cerramos la estructura con **END WITH**.

Nuestra primera conclusión, es que **grabar macros es una muy buena forma de aprender a entender el código VBA** al momento de dar nuestros primeros pasos en programación en Excel.

4.2.5. Ejemplo (Referencias Relativas)

En este caso vamos a grabar una macro, con las mismas características que la anterior, pero utilizando referencias relativas. Activando dicha opción en la pestaña **Vista -> Macros**, procederemos a grabar una macro que siga los mismos pasos que la anterior. El código generado será el siguiente:

```
Sub Celdas_Naranjas_Ref_Relativas()  
'  
' Celdas_Naranjas_Ref_Relativas Macro  
' Le da formato color naranja a tres celdas contiguas horizontalmente  
'  
' Acceso directo: CTRL+r  
'  
    With Selection.Interior  
        .Pattern = xlSolid  
        .PatternColorIndex = xlAutomatic  
        .ThemeColor = xlThemeColorAccent2  
        .TintAndShade = 0  
        .PatternTintAndShade = 0  
    End With  
End Sub
```

Para estas alturas, ya tenemos la capacidad de comprender cuál es el nombre de la macro, cuál es el atajo del teclado y que acciones genera el código, muy similar al anterior. **La diferencia radica en que no hace referencia en particular, a una celda o rango, sino que solamente expone una selección** y a dicha selección le aplica el lote de instrucciones dentro de la estructura **WITH**.

En la figura 4.12 vemos que, al aplicar la macro a una selección cualquiera, ésta le aplicará el formato que figura en el código.



Figura 4.12. Al usar Referencias Relativas, la macro se aplicará a toda la selección.

4.2.6. Optimizando el código generado por la macro

Como mencionamos anteriormente, si bien no es indispensable entender el código **VBA** creado por la macro, esto resulta muy recomendable para poder optimizar o modificar la generación automática de dicho código manualmente.

Tomemos por ejemplo el caso de la macro anterior y su código generado:

```
Sub Celdas_Naranjas_Ref_Relativas()  
'  
' Celdas_Naranjas_Ref_Relativas Macro  
' Le da formato color naranja a tres celdas contiguas horizontalmente  
'  
' Acceso directo: CTRL+r  
'  
    With Selection.Interior  
        .Pattern = xlSolid  
        .PatternColorIndex = xlAutomatic  
        .ThemeColor = xlThemeColorAccent2  
        .TintAndShade = 0  
        .PatternTintAndShade = 0  
    End With  
End Sub
```

Vemos que existen algunas líneas, dentro de la estructura **WITH**, que hacen referencia a ciertos parámetros que Excel utiliza por defecto; por ejemplo:

```
.Pattern = xlSolid  
.PatternColorIndex = xlAutomatic  
.TintAndShade = 0  
.PatternTintAndShade = 0
```

En estas líneas, antes de editar el código, agregaremos un apóstrofe al principio de cada una, para convertirlas en comentarios y que Excel las ignore al ejecutar la macro.



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



```
Sub Celdas_Naranjas_Ref_Relativas ()
'
' Celdas_Naranjas_Ref_Relativas Macro
' Le da formato color naranja a tres celdas contiguas horizontalmente
'
' Acceso directo: CTRL+r
'

    With Selection.Interior
        .Pattern = xlSolid
        .PatternColorIndex = xlAutomatic
        .ThemeColor = xlThemeColorAccent2
        .TintAndShade = 0
        .PatternTintAndShade = 0
    End With
End Sub
```

Excel, al ejecutar la macro, solamente tendrá en cuenta, dentro de la estructura **WITH**, la línea correspondiente al color del interior de las celdas.

Al correr la macro, vemos que se obtiene el mismo resultado, con este código modificado.

La estructura WITH ahora no tiene sentido, ya que solamente se utiliza una línea dentro de ésta, con lo cual podemos eliminar la estructura y los comentarios dentro de la misma, quedando el código de la siguiente manera:

```
Sub Celdas_Naranjas_Ref_Relativas ()
'
' Celdas_Naranjas_Ref_Relativas Macro
' Le da formato color naranja a tres celdas contiguas horizontalmente
'
' Acceso directo: CTRL+r
'

    Selection.Interior.ThemeColor = xlThemeColorAccent2

End Sub
```

A continuación, veremos cómo generar atajos del teclado personalizados con macros:



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



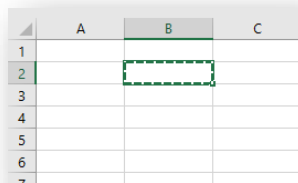
4.2.7. Ejemplo: Pegar Valores con atajo de teclado

Como sabemos, lamentablemente Excel no trae configurado por defecto un atajo de teclado para **Pegar Valores**. Vamos a valernos de la grabación de macros para conseguir esto. En este módulo proponemos la siguiente secuencia.

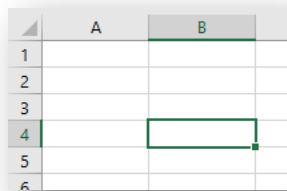
- 1) Utilizar referencias relativas para grabar la macro.



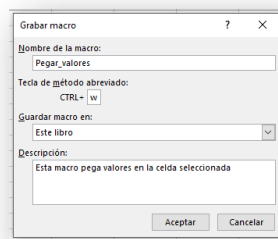
- 2) En una celda cualquiera presionar **CTRL+c**, para copiar su contenido.



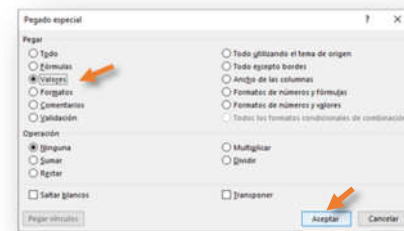
- 3) Seleccionar otra celda cualquiera.



- 4) En este punto, comenzar a grabar la macro, vamos a seleccionar **CTRL+q** como atajo del teclado.



- 5) Con la celda seleccionada presionar **CTRL+ALT+v** (ventana de pegado especial), seleccionamos Valores, hacemos clic en Aceptar.



- 6) Detener la grabación de macros.



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



El código generado será el siguiente:

```
Sub Pegar_valores ()  
'  
' Pegar_valores Macro  
' Esta macro pega valores en la celda seleccionada  
'  
' Acceso directo: CTRL+w  
    Selection.PasteSpecial Paste:=xlPasteValues, Operation:=xlNone, SkipBlanks _  
        :=False, Transpose:=False  
End Sub
```

Modificando manualmente el código **podemos agregar saltos de líneas utilizando el separador “_”**. De esta manera, resultará más sencillo leer el código instrucción a instrucción, quedando:

```
Sub Pegar_valores ()  
'  
' Pegar_valores Macro  
' Esta macro pega valores en la celda seleccionada  
'  
' Acceso directo: CTRL+w  
    Selection.PasteSpecial _  
    Paste:=xlPasteValues, _  
    Operation:=xlNone, _  
    SkipBlanks:=False, _  
    Transpose:=False  
End Sub
```

Es sencillo entender el código si lo comparamos con la ventana de pegado especial (Figura 4.13):

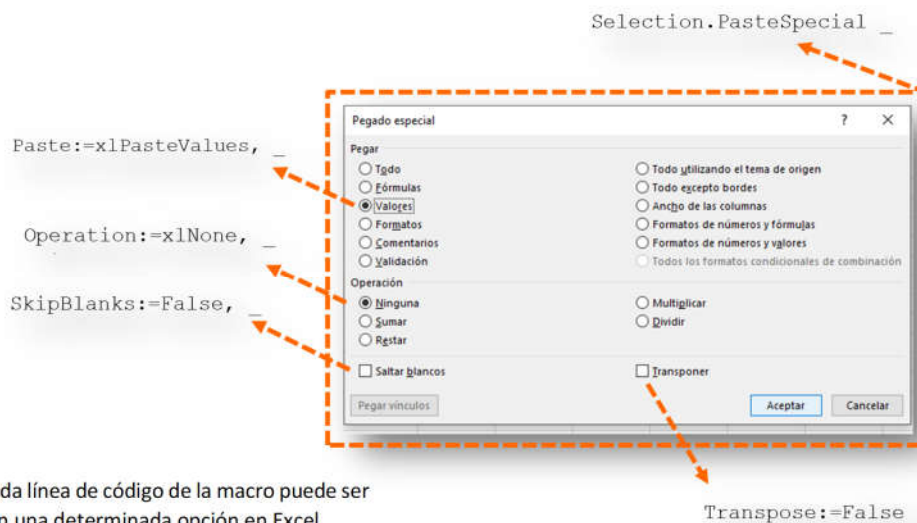


Figura 4.13. Cada línea de código de la macro puede ser relacionada con una determinada opción en Excel.



4.3. Personalizando los mensajes de error

Siguiendo con la macro del ejemplo anterior, supongamos que queremos aplicar el atajo de teclado **Ctrl+w** sin antes haber seleccionado nada (es decir, con el portapapeles vacío), por defecto, Excel nos mostrará el siguiente mensaje emergente. (Figura 4.14):



Figura 4.14. Si el portapapeles está vacío y activamos la macro grabada, Excel devolverá este críptico mensaje.

VBA maneja determinados códigos de errores, dependiendo de su tipo y procedencia.

El error 1004, entre otras cosas, se da cuando al utilizar en una macro la opción Paste (pegar), el portapapeles está vacío.

Podemos realizar algunas modificaciones al código para obtener un mensaje de error menos críptico, para ellos debemos aprender a utilizar la función **MSGBOX** que nos permite mostrar un cuadro de diálogo con algunos grados de personalización.

4.3.1. La función MSGBOX

Esta función tiene 5 argumentos (1 obligatorio y 4 optativos), de los cuales en este módulo explicaremos los tres primeros, ya que serán los más utilizados. Veamos el código propuesto:

```
Sub Pegar_valores ()  
,  
' Pegar_valores Macro  
' Esta macro pega valores en la celda seleccionada  
,  
' Acceso directo: CTRL+w  
On Error Resume Next  
Selection.PasteSpecial _  
    Paste:=xlPasteValues, _  
    Operation:=xlNone, _  
    SkipBlanks:=False, _  
    Transpose:=False  
If Err.Number = 1004 Then  
Msgbox "El portapapeles está vacío", vbOKOnly + vbExclamation, _  
    "No hay ningun valor para pegar"  
End If  
End Sub
```



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



Como dijimos la función **MSGBOX** tiene tres parámetros principales, a saber:

PROMPT (Obligatorio): Es el texto que mostrará el interior del cuadro de diálogo, en el código debe escribirse entre comillas. En nuestro ejemplo será el texto “El portapapeles está vacío”

BUTTONS (Opcional): Por defecto el cuadro de diálogo mostrará el botón Aceptar. El segundo argumento en **MSGBOX** nos permitirá indicar los botones que deseamos mostrar. Podremos, también, elegir el icono y el comportamiento del cuadro de diálogo. La siguiente tabla indica los valores que se usan en este parámetro:

Bloquea	Constante	Valor	Descripción
A	vbOKOnly	0	Muestra solamente el botón Aceptar.
	vbOKCancel	1	Muestra los botones Aceptar y Cancelar.
	vbAbortRetryIgnore	2	Muestra los botones Anular, Reintentar y Omitir
	vbYesNoCancel	3	Muestra los botones Sí, No y Cancelar.
	vbYesNo	4	Muestra los botones Sí y No.
	vbRetryCancel	5	Muestra los botones Reintentar y Cancelar.
B	vbCritical	16	Muestra el icono de mensaje Crítico
	vbQuestion	32	Muestra el icono de Pregunta
	vbExclamation	48	Muestra el icono de mensaje de advertencia.
	vbInformation	64	Muestra el icono de mensaje de Información.
C	vbDefaultButton1	0	El primer botón es el predeterminado.
	vbDefaultButton2	256	El segundo botón es el predeterminado.
	vbDefaultButton3	512	El tercer botón es el predeterminado.
	vbDefaultButton4	768	El cuarto botón es el predeterminado.
D	vbSystemModal	0	Se suspenden todas las aplicaciones hasta que el usuario responda al cuadro de mensaje.
	vbApplicationModal	4096	Hasta que no se cierre el cuadro de diálogo no es posible utilizar ninguna otra función del programa.
E	vbMsgBoxHelpButton	16384	Muestra un botón de ayuda, pero no se mostrará ninguna ayuda al hacer clic en el botón.
	VbMsgBoxSetForeground	65536	La ventana del mensaje se establece como ventana en primer plano
	vbMsgBoxRight	524288	Alinea el texto a la derecha
	vbMsgBoxRtlReading	1048576	Especifica que el texto debe ser leído de derecha a izquierda en árabe y hebreo



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



El bloque **A** determina el número y el tipo de botones a mostrar en el cuadro de diálogo.

El bloque **B** indica el estilo de icono.

El bloque **C** hace referencia a cuál será el botón predeterminado.

El bloque **D** hacen referencia al tipo de modalidad del cuadro.

El bloque **E** determina la ventana en primer plano, la alineación y la dirección del texto.

La forma tradicional, y más recomendada de hacer referencia a los bloques en el código es con el nombre de las constantes (también se pueden usar los números correspondientes). **Solamente se puede elegir UN valor por bloque y estos deben separarse con el símbolo de suma "+"**.

TITLE (opcional): Es la Cadena de texto que se mostrará como el título del cuadro de diálogo, es recomendable utilizar este argumento. (Si no se utiliza, el valor a mostrar en el título será el de la aplicación en donde se utiliza el **VBA**, en este caso Microsoft Excel).

En nuestro caso, el cuadro a mostrar será. (Figura 4.15):

**Msgbox "El portapapeles está vacío", vbOKOnly + vbExclamation, _
"No hay ningun valor para pegar"**

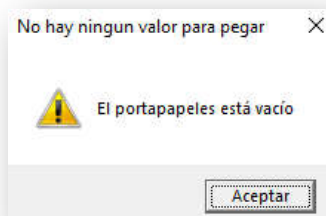


Figura 4.15. Hemos reemplazado el anterior mensaje de error 1004 con éste. Mucho más entendible.

El cuadro de diálogo debe aparecer "reemplazando" a la ventana emergente que indicaba el **error 1004**. Para ello debemos decirle a VBA que intente desestimar el error producido con la instrucción **On Error Resume Next**.

```
Sub Pegar_valores()  
'  
' Pegar_valores Macro  
' Acceso directo: CTRL+w  
On Error Resume Next  
Selection.PasteSpecial _  
    Paste:=xlPasteValues, _  
    Operation:=xlNone, _  
    SkipBlanks:=False, _  
    Transpose:=False  
If Err.Number = 1004 Then  
Msgbox "El portapapeles está vacío", vbOKOnly + vbExclamation, _  
"No hay ningun valor para pegar"  
End If  
End Sub
```



4.4. On Error Resume Next

Esta sentencia nos dice que, si la instrucción genera un error, salte a la siguiente. En nuestro caso, si el pegado especial da error por estar el portapapeles vacío, en vez de mostrar la ventana de **error 1004**, el código saltará a la siguiente instrucción, la declaración **IF**.

La declaración **IF** nos permite usar condiciones durante nuestro código ejecutando ciertas partes de él en función de alguna prueba lógica. Existen tres tipos de declaración **IF**, las cuales son:

1) Forma A: **IF... THEN** (la que usamos en nuestro código anterior).

Ésta es la más utilizada y sencilla. Básicamente el código chequea el cumplimiento de determinada condición (IF) y, si ésta se cumple, ejecuta la instrucción luego (THEN). Con esta forma solamente se realiza la acción si la condición es verdadera.

En nuestro ejemplo, si ocurre el error 1004, entonces el código ejecuta las instrucciones correspondientes al cuadro de diálogo. **Es importante que luego de este código, se termine la declaración IF con un END IF.**

Recapitulando:

Si el portapapeles está vacío y se ejecuta la macro “pegar valores”, ésta generará un **error 1004**.

Al código de la macro le antecedemos la instrucción **On Error Resume Next** para que, si ocurre algún error, automáticamente ejecute la instrucción posterior dicho código.

Como instrucción posterior utilizamos la declaración **IF... THEN**, ésta evaluará la ocurrencia del **error 1004** para luego desplegar un cuadro de dialogo personalizado.

```
If Err.Number = 1004 Then
Msgbox "El portapapeles está vacío", vbOKOnly + vbExclamation, _
"No hay ningun valor para pegar"
End If
```

2) Forma B: **IF...THEN...ELSE**

Puede suceder que queramos que el código realice una acción si la condición es verdadera y otra si la condición es falsa, lo ejemplificaremos con un sencillo ejemplo. El **ELSE** debe estar acompañado del doble punto.

```
Sub IF_ELSE()

    If Range("A1") = 1 Then
        'Si la celda A1 tiene el valor 1, entonces
        Range("A1").Interior.Color = vbYellow
        'Pintar el interior de la celda de amarillo
    Else: Range("A1").Interior.Color = vbRed
        'De otro modo, pintar el interior de la celda de rojo
    End If
End Sub
```



3) Forma C: IF... ELSEIF

En este caso podemos pedirle al código que realice varias comprobaciones, en donde se podrá especificar una acción para cada una de ellas.

```
Sub IF_ELSEIF()  
  
    If Range("A1") < 1 Then  
        'Si la celda A1 tiene un valor menor a 1, entonces  
        Range("A1").Interior.Color = vbYellow  
        'Pintar el interior de la celda de amarillo  
    ElseIf Range("A1") > 1 Then  
        'De lo contrario si la celda A1 tiene un valor mayor a 1, 'entonces  
        Range("A1").Interior.Color = vbGreen  
        'Pintar el interior de la celda de verde  
    End If  
End Sub
```

Se puede agregar un **ELSE** luego del (o los) **ELSEIF**. Como veremos a continuación:

```
Sub IF_ELSEIF_ELSE()  
  
    If Range("A1") > 1 Then  
        'Si la celda A1 tiene el valor mayor a 1, entonces  
        Range("A1").Interior.Color = vbGreen  
        'Pintar el interior de la celda A1 de verde  
    ElseIf Range("A1") = 0 Then  
        'De lo contrario si la celda A1 tiene el valor igual a 0, 'entonces  
        Range("A1").Interior.Color = vbYellow  
        'Pintar la celda A1 de amarillo  
    Else: Range("A1").Interior.Color = vbRed  
        'De lo contrario, pintar la celda A1 de rojo  
    End If  
End Sub
```

4.5. La instrucción Select CASE

Puede suceder que queramos contrastar varias condiciones, lo que en principio nos obligaría a utilizar iteraciones del **IF... ELSEIF**. En estos casos, **VBA nos brinda la instrucción Select CASE que permite la contrastación de varias condiciones con un determinado curso de acción para cada una.**

En el ejemplo, contrastaremos el valor de la celda A1 y le asignaremos colores dependiendo de su valor numérico.



```
Sub SELEC_CASE ()  
Select Case Range("A1").Value  
Case Is = 0  
Range("A1").Interior.Color = vbGreen  
Case Is = 1  
Range("A1").Interior.Color = vbBlue  
Case Is = 2  
Range("A1").Interior.Color = vbYellow  
Case Is = 3  
Range("A1").Interior.Color = vbCyan  
Case Is = 4  
Range("A1").Interior.Color = vbMagenta  
Case Else  
Range("A1").Interior.Color = vbWhite  
End Select  
End Sub
```

Los anteriores ejemplos nos permitieron introducir algunos conceptos básicos de programación en **VBA** a modo de introducción. Es importante recordar que las macros representan parte muy pequeña de la potencialidad del lenguaje **VBA**.

A continuación, describiremos brevemente el concepto de programación orientada a objetos.

4.6. ¿Qué es la programación orientada a objetos?

Los lenguajes de programación orientada a objetos se caracterizan por trabajar con ciertas estructuras denominadas **objetos** (existentes o creadas en el lenguaje) en donde todas las instrucciones hacen referencias a alguno de estos objetos.

VBA es un lenguaje de programación orientado a objetos embebido dentro de las aplicaciones Office. En el caso de Excel, cuando hablamos de objetos, hablamos de todos los elementos existentes dentro del programa; por ejemplo: Celdas, Rangos, Hojas, Libros, Gráficos, Tablas, archivos externos, etc. Los objetos poseen una jerarquía, propiedades y métodos.

4.6.1. Jerarquía

Para entender el concepto de jerarquía en Excel, podemos considerar como unidad de organización mínima son las celdas. Un conjunto de celdas formaría un rango y al mayor conjunto de celdas que puede contener un rango lo podemos denominar hoja. Por encima de la hoja encontramos el libro, que es un conjunto de hojas, y por encima de todo tenemos la aplicación Excel.



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



4.6.2. Colecciones

Una colección es un conjunto de objetos que tienen características similares, en Excel se suelen definir colecciones y luego explicitar el objeto correspondiente.

Componente	VBA - Object	VBA - Colection
Excel	Application	
Libro	Workbook	Workbooks
Hojas	Worksheet	Worksheets
Rangos (mínimo 1 celda)	Range	

Cuando definimos en el código objetos con jerarquías inferiores, debemos separarlas por medio de un punto (.) El objeto que se encuentre a la izquierda del punto será el contenedor mientras que, el que se encuentre a la derecha del punto será el objeto contenido.

4.6.3. Propiedades

En Excel, los objetos tienen propiedades, por ejemplo, una **celda (Range)** tiene como algunas de sus propiedades el **valor (Value)**, su **tipografía (Font)**, su tipo de **borde (Borders)**, entre otras.

```
Sub Hola()  
Application.Workbooks("Libro1").Worksheets("Hojal").Range("A1").value="Hola"  
  
'El Código anterior devuelve el valor "Hola" a la celda A1 de la hoja "hojal", 'del archivo  
'"Libro1".  
End sub
```

4.6.4. Métodos

Los métodos son las actividades o acciones que un objeto puede realizar. Siguiendo el ejemplo de una celda (Range), los métodos pueden ser: **activarla (Activate)** o **borrarla (Clear)**, entre muchos otros.

```
Sub Borrar_Celda_A1()  
Application.Workbooks("Libro1").Worksheets("Hojal").Range("A1").clear  
  
'El Código anterior borra el contenido de la celda A1.  
End sub
```

4.6.5. Eventos

Profundizando la característica de los objetos, podemos nombrar también de la existencia de eventos de los objetos en Excel, por ejemplo, el evento **Workbook_SheetActivate** nos permite ejecutar una instrucción específica cada vez que se activa una determinada hoja dentro de un libro. No nos detendremos en el estudio de los eventos.



4.7. Explorando los métodos y propiedades de un objeto

Si dentro del **Editor de VBA (Alt+F11)** escribimos el nombre de uno de estos objetos seguido de un punto (.) se mostrará una lista con los métodos y propiedades del objeto (Figura 4.16)

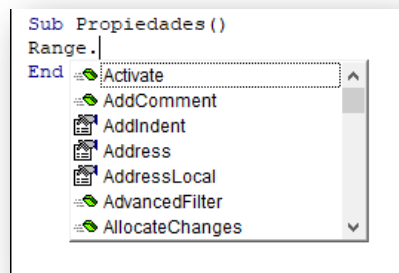


Figura 4.16. Si bien se muestran en la misma lista las propiedades y métodos de un objeto, éstos se distinguen entre sí por sus iconos correspondientes.

4.7.1. Variables y Funciones en VBA

En la medida que avanzamos en nuestro aprendizaje de **VBA**, nos damos cuenta qué muchas veces necesitamos “almacenar” resultados de alguna operación o guardar valores; es justamente para esto que **VBA** nos permite usar contenedores polivalentes de datos denominados **variables**.

En VBA existen distintos tipos de variables, dependiendo la clase de dato que queramos albergar dentro de ellas. **Existen variables tipo número entero, número decimal, moneda, fecha, texto; entre las más importantes.**

VBA no nos obliga a definir qué tipo de variables vamos a utilizar, a menos que, en el código, antepongamos la expresión **Option Explicit**.

Si utilizamos dicha expresión debemos, antes de poder utilizar la variable, declararla utilizando la expresión **Dim** **As** y la palabra especial que defina su tipo.

Los principales tipos de variables son (vamos a suponer que utilizamos **Option Explicit**):

1) Variables de tipo número entero (Integer)

Las variables de tipo entero son utilizadas para guardar números enteros. Cuando utilizamos este tipo de variable debemos declararla con la palabra **Integer**.

`Option Explicit`

```
Sub Variable_Numero_Entero()  
Dim MiVariable As Integer  
MiVariable = Range ("A1").value  
Msgbox (MiVariable)  
End sub
```



4.7.1.1. Variables de tipo número decimal (Double)

Las variables de tipo doble pueden almacenar números decimales. Se declaran con la palabra **Double**. Si bien pueden almacenar números enteros, este tipo de variables ocupa más memoria que las variables tipo **Integer**. Si solamente vamos a utilizar números enteros (por ejemplo, para la edad en años cumplidos), no es recomendable definirla la variable respectiva como **Double**.

Option Explicit

```
Sub Variable_Numero_Decimal()  
Dim MiVariable2 As Double  
MiVariable2 = Range ("A1").value  
Msgbox (MiVariable2)  
End sub
```

4.7.1.2. Variable tipo cadena de texto (String)

Las variables tipo texto se declaran con la palabra **String**. Cuando se le asigna algún valor de texto a la variable, esta debe estar contenida por doble comillas (""). Se pueden asignarle números a este tipo de variables, pero solamente se tendrá en cuenta el símbolo y no el valor numérico de este. Ver figura 4.17.

Option Explicit

```
Sub Variable_Cadena_Texto()  
Dim MiVariable3 As String  
Dim MiVariable4 As String  
Dim MiVariable5 As String  
MiVariable3 = Range ("A1").value  
MiVariable4 = Range ("A2").value  
MiVariable5 = MiVariable3 + MiVariable4  
Msgbox (MiVariable5)  
End sub
```

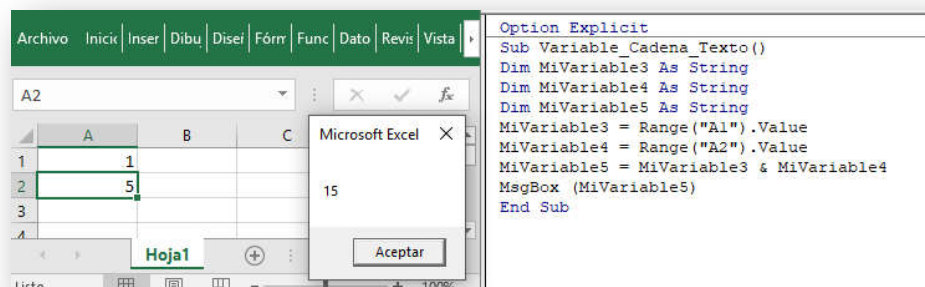


Figura 4.17. Cuando trabajamos con variables **String**, el símbolo "+" concatena las cadenas.



4.7.1.3. Variable tipo Lógico (Boolean)

Las variables tipo lógicas solamente admiten dos valores: verdadero (true) o falso (false). Se declaran con la palabra **Boolean**. Ver figura 4.18.

Option Explicit

```
Sub Variable_Logica_Texto()  
Dim MiVariable6 As Boolean  
MiVariable6 = Range ("A1").value  
Msgbox (MiVariable6)  
End sub
```

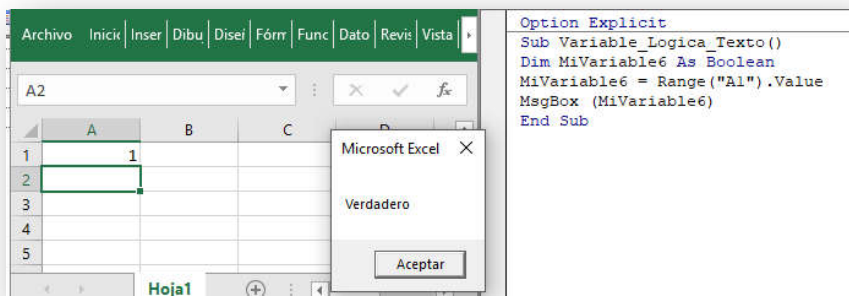


Figura 4.18. El cuadro de diálogo resultante mostrará la leyenda “Verdadero” o “Falso”, dependiendo el valor de la celda A1. Si la celda no tiene valor, el resultado será “Falso”. Caso contrario será “Verdadero”

4.7.1.4. Variable tipo miscelánea (Variant)

Este tipo de variable puede almacenar cualquier tipo de dato (numéricos, textos o fechas). Ocupa un espacio variable en la memoria (un tamaño fijo de 22 bytes + la longitud de los datos). Es la variable que más espacio de memoria ocupa por defecto.

Si una variable no se declara como un tipo de los anteriores, en el caso de no anteponer **Option Explicit** al código, se supone que es una **Variant**.

```
Sub Variable_Sin_Declarar()  
MiVariable7 = Range ("A1").value  
Msgbox (MiVariable6)  
End sub
```

Remarcamos la recomendación de definir la variable según el tipo de dato que se utilizará. Esto redundará en una mayor eficiencia del código a generar en lo que respecta a velocidad de ejecución y optimización de la memoria utilizada.



4.7.1.5. Variables tipo Objeto (Set)

Una variable objeto, o tipo objeto, almacena un objeto completo, tal como una hoja de trabajo o un rango de celdas. Para asignar un objeto a una variable se utiliza la instrucción **Set**.

Las variables objeto almacenarán también las propiedades del objeto referenciado

Resulta muy recomendable el uso de las variables objeto al escribir el código, ya que permite simplificar la programación y optimizar la velocidad de ejecución de éste.

A continuación, procederemos a comparar dos códigos, uno que **no contenga variables** objetos y otros que **sí las contenga**, observando cómo se simplifica el código.

```
Sub Sin_Variable_Objeto()
```

```
'Sin variable objeto
```

```
Worksheets("Hoja1").Range("A1").Value = "Hola mundo"
```

```
Worksheets("Hoja1").Range("A1").Font.Bold = True
```

```
Worksheets("Hoja1").Range("A1").Font.Italic = True
```

```
Worksheets("Hoja1").Range("A1").Font.Size = 10
```

```
Worksheets("Hoja1").Range("A1").Font.Name = "Arial"
```

```
End Sub
```

```
Sub Con_Variable_Objeto ()
```

```
Dim MiCelda As Range
```

```
'Declaración de la variable objeto
```

```
Set MiCelda = Worksheets("Hoja1").Range("A1")
```

```
'Uso de la instrucción SET para asignar valores a la
```

```
'variable objeto Celda
```

```
Celda.Value = 200
```

```
Celda.Font.Bold = True
```

```
Celda.Font.Italic = True
```

```
Celda.Font.Size = 10
```

```
Celda.Font.Name = "Arial"
```

```
End Sub
```



4.8. El bucle For – Next

En los lenguajes de programación, se denomina **bucle o ciclo** a aquella instrucción que nos permite ejecutar repetidamente un conjunto de instrucciones hasta que se cumpla una determinada condición que hayamos especificado.

El bucle **For – Next** es una de las instrucciones más útiles al programar en **VBA**. Tiene una sintaxis muy sencilla de utilizar y también de interpretar al analizar el código:

El siguiente código **escribe números consecutivos del 1 al 10** en la primera columna, comenzando en la primera fila.

```
Sub Contar()  
For Cuenta = 1 To 10  
    fila = Cuenta  
    Cells(fila, 1) = Cuenta  
Next  
End Sub
```

Inicialización de variable: La instrucción **For – Next** repite un conjunto de instrucciones un número de veces específico. Debemos especificar una variable que irá contando cada una de las repeticiones.

La instrucción suele iniciar habitualmente como **For i = 1** lo cual indica que la variable **i** llevará la cuenta de las repeticiones que deseamos **iniciando en el valor 1**.

Límite: Luego de especificar la variable que llevará la cuenta de las repeticiones, debemos explicitar un límite donde se detendrá el ciclo.

Este límite se indica con la instrucción **To**. De esta manera, si deseamos hacer un bucle que vaya desde 1 hasta 20, escribiremos: **For i = 1 To 20**.

Incrementar variable: La instrucción **Next** seguida del nombre de la variable que lleva la cuenta, será la que repetirá la acción de incrementar su valor hasta llegar al límite preestablecido.

El bucle se termina con la instrucción **Next i**.

Por defecto, la variable se incrementa de uno en uno, a menos que agreguemos la instrucción **Step**, luego del límite, cuando especificamos la variable.

El siguiente código escribirá en las filas 1, 3, 5, 7 y 9 de la segunda columna los números empezando desde el 1 y contando de dos en dos.

```
Sub Contar_2_en_2_Celdas_Impares()  
For Cuenta = 1 To 10 Step 2  
    fila = Cuenta  
    Cells(fila, 1) = Cuenta  
Next  
End Sub
```



Si prestamos atención al código anterior, la fila en donde se escribirá será igual al valor de la variable **cuenta**. ¿Cómo podemos hacer para que las filas sean sucesivas y continuas? Para ello tenemos la función integrada **Offset**.

4.9. Uso de la función Offset en VBA

Junto con la instrucción **For – Next** podemos usar la función **Offset**, que nos permite desplazarnos a determinada cantidad de filas o de columnas.

En este caso cada vez que se ejecute una instrucción del ciclo **For – Next**, el código desplazará una fila hacia abajo, hasta llegar al límite del bucle.

```
Sub Contar_2_en_2_Celdas_Contiguas()  
fila = 0  
For Cuenta = 1 To 10 Step 2  
    Cells(1, 1).Offset(fila, 0) = Cuenta  
    fila = fila + 1  
Next  
End Sub
```

A comparación del código anterior, el código escribirá en celdas contiguas, comenzando desde la primera, números del 1 al 10 contados de dos en dos. (es decir 1, 3, 5, 7 y 9). Ver figura 4.19.

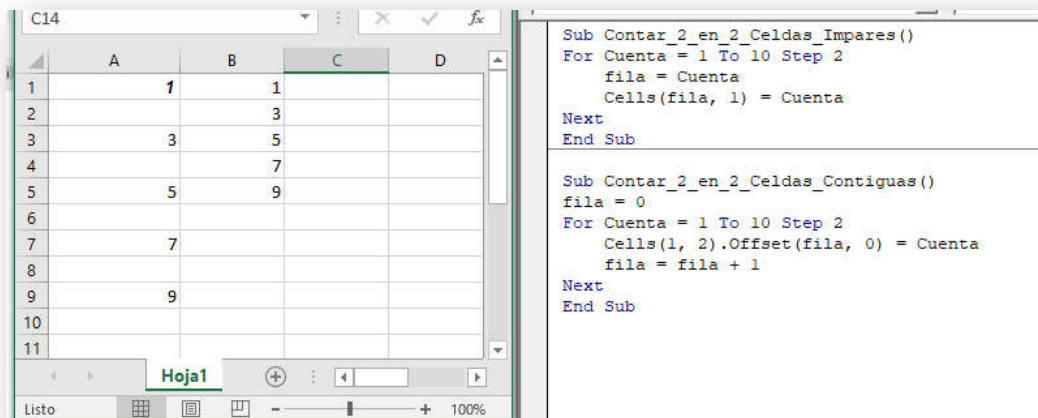


Figura 4.19. La función **Offset (fila, columna)** desplazará las filas hacia arriba (valores negativos del primer argumento) o hacia abajo (valores positivos del primer argumento). En cuando a las columnas, la función **Offset** las desplazará hacia la izquierda (valores negativos del segundo argumento), o hacia la derecha (valores positivos del segundo argumento). La función **offset** desplazará la cantidad de filas o columnas que figure en el valor numérico de los argumentos correspondientes.



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



Estamos listos para estudiar el último tema del presente módulo, la creación de funciones personalizadas en VBA. *Este tipo de funciones personalizadas nos permitirán ampliar en gran medida la ya basta cantidad de funciones integradas que nos ofrece Excel.*

4.10. Funciones personalizadas en Excel – VBA

En módulos anteriores vimos que una función es una fórmula predefinida en Excel que opera sobre uno o más argumentos.

Excel 2016 trae más de 470 funciones incorporadas, por ejemplo: SUMA, PROMEDIO, AHORA, HOY, BUSCARV.

Utilizando los conocimientos adquiridos hasta ahora, vamos a utilizar VBA para crear nuestras propias funciones de una manera muy sencilla.

4.10.1. Creando nuestras primeras funciones en VBA

A las funciones personalizadas, también se las conoce como **funciones predefinidas por el usuario o UDF (en inglés User Defined Function)**. Es necesario mencionar que no se puede crear funciones personalizadas a partir de la grabadora de macros, ya que la misma solamente es funcional para crear subrutinas.

Iniciamos con una necesidad específica que no pueda ser resuelta con alguna función de Excel. En este caso, tenemos un total que tiene cargado un IVA del 21% y queremos saber el valor correspondiente sin IVA. Por ejemplo, de un total de 121, sabemos que el neto sin IVA será 100.

El código será el siguiente:

```
Function NETO_SIN_IVA(Valor)
NETO_SIN_IVA = Valor / 1.21
End Function
```

En el ejemplo anterior estamos limitados a un valor de IVA del 21%, la pregunta que nos hacemos es cómo podríamos agregarle versatilidad a nuestra **UDF**. Podemos adicionarle un segundo parámetro a la función, en donde introduciríamos el valor de la tasa manualmente (o haciendo referencia a otra celda).

```
Function NETO_SIN_IVA_TASA(Valor , Tasa)
NETO_SIN_IVA = Valor / (1 + Tasa)
End Function
```

Como podemos observar, no resulta demasiado complicado crear **UDFs** sencillas a partir de fórmulas que introduzcamos dentro del código **VBA**.



A continuación, daremos un ejemplo de cómo construir una **UDF** bastante útil que nos permitirá sumar celdas que tengan determinado color de relleno. Antes de analizar el código propuesto para dicha función, introduciremos una estructura muy útil cuando trabajamos con un rango de celdas en VBA, ésta es el bucle **For Each – Next**.

4.10.2. El bucle For Each – Next

Hemos visto con anterioridad el bucle **For – Next**, el cual repite un conjunto de instrucciones un número de veces específico. Ahora analizaremos una estructura similar, pero algo más versátil, el bucle **For Each – Next**. La estructura **For Each – Next** nos permite recorrer todos los elementos de un rango y realizar acciones para cada uno de estos elementos. Veamos un código de ejemplo y luego su sintaxis.

```
Sub Ejemplo_For_Each()  
Dim celda As Range  
For Each celda In Range("A1:A6")  
    celda.Value = celda.Value * 2  
Next celda  
End Sub
```

El anterior código, luego de declarar una determinada variable como parte de un rango de celdas, le asigna a cada celda un valor que duplica al anterior que tenía, siempre y cuando la celda pertenezca al rango A1:A6.

Vamos a utilizar esta estructura para nuestra **UDF** que **sumará los valores siempre y cuando la celda cumpla con la condición de tener un color de relleno igual al de una celda de referencia**.

```
Function SumaColores(Color As Range, Range As Range) As Double  
Dim celda As Range  
Dim colornumero As Integer  
Dim sumacol  
  
colornumero = Color.Interior.ColorIndex  
  
For Each celda In Range  
If celda.Interior.ColorIndex = colornumero Then  
    sumacol = WorksheetFunction.Sum(celda.Value) + sumacol  
End If  
Next celda  
  
SumaColores = sumacol  
  
End Function
```



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



Primero definimos los campos que tomarán la función, luego definimos que el primer argumento tomará los colores de dicha celda, y el segundo campo sólo escogerá los valores de un rango de datos.

los cuales deberán coincidir con el primer campo, el resultado de la función será un valor numérico que admita decimales (**Double**)

```
Function SumaColores(Color As Range, Range As Range) As Double
```

Luego definimos a las variables que utilizaremos, y le diremos al Excel que la variable **colornumero** será igual al valor del color que se encuentren en la respectiva celda que escogeremos.

```
Dim celda As Range  
Dim colornumero As Integer  
Dim sumacol
```

Una vez que ya definimos la celda a escoger, decimos que, para cada celda del rango elegido en el segundo argumento, si la celda coincide con el color que estamos buscando, la incluya en la sumatoria.

luego con **Next** indicamos que busque a la siguiente celda y si esta coincide la agrega a la sumatoria, de esta forma se seleccionarán a todas las celdas del mismo color que se encuentren en el rango seleccionado.

```
For Each celda In Range  
If celda.Interior.ColorIndex = colornumero Then  
sumacol = WorksheetFunction.Sum(celda.Value) + sumacol  
End If  
Next celda
```

```
SumaColores = sumacol
```

```
End Function
```

4.10.3. Funciones personalizadas que recalculen automáticamente

Por defecto, al crear una **UDF**, ésta no se actualiza si cambian los valores de las celdas usadas como referencia en los argumentos de la función creada. Para que esto suceda hay que agregarle, luego de la definición del nombre y cantidad de argumentos de la UDF, la instrucción **Application.Volatile**

```
Function NETO_SIN_IVA_TASA(Valor , Tasa)  
Application.Volatile  
NETO_SIN_IVA = Valor / (1 + Tasa)  
End Function
```



TÓPICOS DE EXCEL

(Material teórico gratuito 2019)



5. Conclusiones

El objetivo de este material teórico es simplemente el intentar introducir la inquietud sobre la utilidad empírica que tiene hoy en día Excel en el ámbito de las ciencias económicas en particular y en el ámbito laboral en general.

Al ser un material de carácter introductorio y no abarcativo, sabemos que han quedado infinidad de características propias de Excel que no han sido tratadas en profundidad o siquiera han sido mencionadas.

Entendemos que es virtualmente imposible, y poco sensato, intentar capturar en un material escrito, que siempre es estático, algo tan dinámico como lo son las potencialidades que ostenta Excel.

El trabajo de aprendizaje para conseguir dominar Excel del lector recién empieza, de hecho, podemos decir, sin temor a equivocarnos que es un apasionante camino que nunca termina.

La ventaja actual es que existe infinidad de material y documentación en la web, para quién esté realmente interesado en aprender, sin limitación horario o física.

¡Invitamos a nuestros lectores a seguir participando del grupo de Facebook CECE FCE-UBA EXCEL con sus inquietudes y sugerencias! Recordamos también que todos los archivos y ejemplos a los que eventualmente hiciera referencia este material pueden ser descargados desde dicho grupo.

6. Bibliografía

Alexander, M., & Kusleika, D. Excel (2016) Power programming with VBA. Indianapolis, Wiley.

Collie, R., & Singh, A. (2015) Power Pivot and Power BI, Tickling Keys.

Jelen, B. (2015) Excel 2016 in depth. Pearson Education.

Walkenbach, J. (2016). Microsoft® Excel® 2016 bible. Indianapolis, Wiley.