

EJEMPLOS INTEGRADORES

Ejemplo 1 – Macros para copiar, cortar y pegar un contenido

Las dos primeras macros copian y cortan el contenido de la celda, rango o tabla seleccionada por el usuario al momento de ejecutar la macro.

La última macro pega el contenido copiado o cortado previamente en la hoja activa (sobre la celda en donde se encuentre el usuario).

Sub Copiar()

Selection.Copy 'Copy es un método / acción para copiar un objeto

End Sub

Sub Cortar()

Selection.Cut 'Cut es un método / acción para cortar un objeto

End Sub

Sub Pegar()

ActiveSheet.Paste 'Paste es un método / acción para pegar el contenido copiado sobre una hoja de Excel

End Sub

Nota: El objeto *SELECTION* representa a la celda o rango que el usuario esté seleccionando al momento de ejecutar la macro.

Ejemplo 2 – Macro para aplicar un filtro

Esta macro aplica un filtro sobre la tabla perteneciente a la celda activa que el usuario haya presionado.

Si ocurriera un error, por ejemplo, que el usuario no esté seleccionando ninguna celda de la tabla, se ejecutarán los comandos de la etiqueta denominada "ErrorFiltro". Esta etiqueta, que se define al final de la macro, muestra un mensaje de información: "Debe haber un filtro aplicado para realizar esta operación".

Sub AplicarFiltro()

'Verifica si existe o no un error en la aplicación del filtro. Si lo hay, ejecuta los comandos de la etiqueta "ErrorFiltro"

On Error GoTo ErrorFiltro

ActiveCell.AutoFilter

Exit Sub

'Esta es la etiqueta "ErrorFiltro" que se aplicará si aparece algún error en la ejecución de la macro

ErrorFiltro:

MsgBox "Debe haber un filtro aplicado para realizar esta operación"

End Sub

Nota: Recordar que el comando **EXIT SUB** permite finalizar en forma anticipada la ejecución de la macro dentro del Código. Es necesario incluirlo antes de definir la etiqueta de error para que ésta sólo se ejecute en caso de error.

Ejemplo 3 – Macro para generar un reporte

Esta macro genera un reporte de la información filtrada en una tabla. Copia dicha información en un nuevo libro de Excel, autoajusta sus columnas y renombra a la hoja de ese nuevo libro como "Reporte".

Si ocurriera un error, por ejemplo, que no hubiera información filtrada en la tabla, se ejecutarán los comandos de la etiqueta denominada "ErrorReporte". Esta etiqueta, que se define al final de la macro, muestra un mensaje de información: "Debe haber un filtro aplicado para realizar esta operación".

Sub GenerarReporte()

'Verifica si existe o no un error en la generación del reporte. Si lo hay, ejecuta los comandos de la etiqueta "ErrorReporte"

On Error GoTo ErrorReporte

'Copia la información filtrada en un nuevo libro que se crea

ActiveSheet.AutoFilter.Range.Copy

Workbooks.Add

Sheets(1).Paste

'Autoajusta el ancho de las nuevas columnas

Cells.EntireColumn.AutoFit

'Establece el nombre de la hoja

Sheets(1).Name = "Reporte"

Exit Sub

'Esta es la etiqueta "ErrorReporte" que se aplicará si aparece algún error en la ejecución de la macro

ErrorReporte:

MsgBox "Debe haber un filtro aplicado para realizar esta operación"

End Sub

Ejemplo 4 – Macro para mostrar información filtrada (borrar filtro)

Esta macro borra el filtro aplicado sobre una planilla mostrando toda la información e la hoja.

Si ocurriera un error, por ejemplo, que no haya información filtrada, se ejecutarán los comandos de la etiqueta denominada "ErrorMostrarInfo". Esta etiqueta, que se define al final de la macro, muestra un mensaje de información: "Debe haber un filtro aplicado para realizar esta operación".

Sub MostrarInformacion()

'Verifica si existe o no un error al mostrar toda la información. Si lo hay, ejecuta los comandos de la etiqueta "ErrorMostrarInfo"

On Error GoTo ErrorMostrarInfo

ActiveSheet.ShowAllData

Exit Sub

'Esta es la etiqueta "ErrorMostrarInfo" que se aplicará si aparece algún error en la ejecución de la macro

ErrorMostrarInfo:

MsgBox "Debe haber un filtro aplicado para realizar esta operación"

End Sub