With WS\_PedAb

*' ---&--- Tabla Columnas*

Total\_Col = 4

Empezamos en la columna 4 y vamos saltando de 3 en 3 las columnas para leer los códigos de artículo.  
Las celdas combinas el dato siempre esta en la primera celda de la izquierda y la primero superior.

El **Redim** sirve para cambiar la dimensión de la tabla.  
Si crea un entrada en la tabla para cada artículo en lugar de tres porque así la tabla es más pequeña y se busca de forma más rápida. Además hay una relación directa entre el índice de la tabla y la columna en la que esta.

La tabla tachada no se usa y se me olvido quitar.

While .Cells(5, Total\_Col).Value <> ""

ReDim Preserve Tabla\_Col\_Inv((Total\_Col - 1) / 3)

ReDim Preserve Tabla\_Col\_Ped((Total\_Col - 1) / 3)

Estas tres tablas son para los totales de **Transito,** **Reserva** y **Retenido** Tendrían que ser del mismo tipo de dimensión que la de los artículos, pero como no sabía que tenía que ponerse, En lugar de 3 columnas por artículo con 1 era suficiente.

ReDim Preserve Tabla\_Transit(Total\_Col)

ReDim Preserve Tabla\_Reserva(Total\_Col)

ReDim Preserve Tabla\_Retenid(Total\_Col)

En la tabla guarda el código de articulo más el Lot/Ser con un separador entre campos

La tabla tachada no se usa y se me olvido quitar.

Tabla\_Col\_Inv((Total\_Col - 1) / 3) = Val(.Cells(4, Total\_Col)) & "·" & UCase(.Cells(3, Total\_Col))

Tabla\_Col\_Ped((Total\_Col - 1) / 3) = Val(.Cells(4, Total\_Col))

Total\_Col = Total\_Col + 3

Wend

*' ---&--- Añado las columnas Orden y Asigna del ultimo articulo*

Esto de aquí sobra porque estas dos columnas no se usan

Total\_Col = Total\_Col + 2

ReDim Preserve Tabla\_Transit(Total\_Col)

ReDim Preserve Tabla\_Reserva(Total\_Col)

ReDim Preserve Tabla\_Retenid(Total\_Col)

*' ---&--- Tabla de datos*

Con el total de filas y columna creamos una tabla donde se irán sumando las diferentes cantidades que después se pondrán en la hoja.

La posición 1,1 de la tabla corresponde a la celda A1

ReDim Tabla\_DAT(Total\_Fil, Total\_Col)

*' </> ------------------------------------------------------------- </>*

*' </> ---&--- Lee la hoja de datos de INVENTARIO*

*' </> ------------------------------------------------------------- </>*

Recorre la hoja inventario desde la fila 2, la 1 es la cabecera, hasta que encuentra la celda de la columna **A** vacía

With WS\_Inven

Fila = 2

While .Cells(Fila, "A") <> ""

Filtra el almacén y el Origen

If (UCase(.Cells(Fila, "A")) = "200VR" Or .Cells(Fila, "A") = 200) And \_

UCase(.Cells(Fila, "O")) = "UHT" Then

Monta en la variable Texto la clave a buscar, en este caso Artículo + Lot/Ser. Te preguntaras porque pongo el punto, bueno es una buena práctica porque según el tipo de código, no es el caso puede provocar confusión, no es lo mismo buscar pan·carta que buscar pancarta por poner un ejemplo simple.

Texto = .Cells(Fila, "C") & "·" & UCase(.Cells(Fila, "E"))

Col = Buscar\_Col\_Inv(Texto, Tabla\_Col\_Inv) *' --- Busca el articulo*

Pos = (Col \* 3) + 1

Si encontró un resultado en la tabla continua

If Col > 0 Then

Busca la fecha en la tabla de Fechas, como las fechas están ordenadas realiza una búsqueda que es mucho mas rápida, en la que consultando un máximo de 6 elemento de la tabla se puede saber si el elemento buscado esta o no.

Es muchísimo más rápido que una búsqueda secuencial.

A las fechas se les da formato año.mes.día para saber si es mayor o menor al comparar en la tabla.

Fil = Buscar\_Fecha(Format(.Cells(Fila, "I"), "yyyy.mm.dd"), Tabla\_Fil) *' --- Busca la fecha*

Si encontró un resultado en la tabla continua

If Fil > 0 Then

El ultimo filtro para incluirlo en la tabla, este filtro no esta incluido en el primer IF porque no afecta a las tablas de más abajo.

POS es la columna del artículo que se calculó con una sencilla formula ( Índice \* 3 ) + 1

En este caso a POS se le suma 0 porque es la primera de las tres columnas del artículo en la hoja.

If UCase(.Cells(Fila, "N")) = "STOCK LIB" Then

Tabla\_DAT(Fil, Pos + 0) = Tabla\_DAT(Fil, Pos + 0) + .Cells(Fila, "G")

End If

End If

Select Case UCase(.Cells(Fila, "N"))

Case "TRANSITO": Tabla\_Transit(Pos) = Tabla\_Transit(Pos) + .Cells(Fila, "G")

Case "RESERVA": Tabla\_Reserva(Pos) = Tabla\_Reserva(Pos) + .Cells(Fila, "G")

Case "RETENIDO": Tabla\_Retenid(Pos) = Tabla\_Retenid(Pos) + .Cells(Fila, "G")

End Select

End If

End If

Fila = Fila + 1

Wend

End With

*Básicamente es lo mismo que el anterior.*

*' </> ------------------------------------------------------------- </>*

*' </> ---&--- Lee la hoja de datos de Pedidos*

*' </> ------------------------------------------------------------- </>*

Recorre la hoja pedido desde la fila 2, la 1 es la cabecera, hasta que encuentra la celda de la columna **A** vacía

With WS\_Pedid

Fila = 2

While .Cells(Fila, "A") <> ""

Filtra el almacén y el Origen

If (UCase(.Cells(Fila, "A")) = "200VR" Or .Cells(Fila, "A") = 200) And \_

UCase(.Cells(Fila, "AE")) = "UHT" Then

Monta en la variable Texto la clave a buscar, en este caso Artículo + Lot/Ser. Te preguntaras porque pongo el punto, bueno es una buena práctica porque según el tipo de código, no es el caso puede provocar confusión, no es lo mismo buscar pan·carta que buscar pancarta por poner un ejemplo simple.

Texto = .Cells(Fila, "I") & "·" & UCase(.Cells(Fila, "AO"))

Col = Buscar\_Col\_Inv(Texto, Tabla\_Col\_Inv) *' --- Busca el articulo*

Pos = (Col \* 3) + 1

Si encontró un resultado en la tabla continua

If Col > 0 Then

Busca la fecha en la tabla de Fechas, como las fechas están ordenadas realiza una búsqueda que es mucho mas rápida, en la que consultando un máximo de 6 elemento de la tabla se puede saber si el elemento buscado esta o no.

Es muchísimo más rápido que una búsqueda secuencial.

A las fechas se les da formato año.mes.día para saber si es mayor o menor al comparar en la tabla.

Fil = Buscar\_Fecha(Format(.Cells(Fila, "AN"), "yyyy.mm.dd"), Tabla\_Fil) *' --- Busca la fecha*

If Fil > 0 Then

En este caso a POS se le suma 1 y 2 porque son la segunda y tercera columna del artículo en la hoja.

Tabla\_DAT(Fil, Pos + 1) = Tabla\_DAT(Fil, Pos + 1) + .Cells(Fila, "L")

Tabla\_DAT(Fil, Pos + 2) = Tabla\_DAT(Fil, Pos + 2) + .Cells(Fila, "AL")

End If

End If

End If

Fila = Fila + 1

Wend

End With

Aquí escribe la tabla en la hoja.

Empieza en la columna 4 porque es donde empieza el primer ARTICULO

Empieza en la fila 6 porque es donde empieza la primera FECHA

El total se coloca dos filas mas abajo que la ultima fecha porque el numero de fecha es variable y la fila total subirá o bajara según se quiten o añadan fecha.  
Según va rellenando la columna va sumando el total y al finalizar la columna lo coloca en el libro.

*' ---&--- Escribe los datos en la hoja Refrigerado*

With WS\_PedAb

For Col = 4 To Total\_Col - 2

For Fil = 6 To Total\_Fil

If Tabla\_DAT(Fil, Col) <> 0 Then

.Cells(Fil, Col) = Tabla\_DAT(Fil, Col)

Total = Total + Tabla\_DAT(Fil, Col)

End If

Next

.Cells(Total\_Fil + 1, Col) = Total

If Tabla\_Transit(Col) <> 0 Then .Cells(Total\_Fil + 3, Col) = Tabla\_Transit(Col)

If Tabla\_Reserva(Col) <> 0 Then .Cells(Total\_Fil + 4, Col) = Tabla\_Reserva(Col)

If Tabla\_Retenid(Col) <> 0 Then .Cells(Total\_Fil + 5, Col) = Tabla\_Retenid(Col)

Total = 0*#*

Next

End With

Y ahora voy a responder a una pregunta no formulada:

¿No es más fácil recorrer las celdas para buscar el articulo y la fecha en lugar de copiar los datos en una tabla?

Puede ser que sea más fácil pero no más rápido leer una celda cuesta más tiempo que leer un elemento de una tabla.

Puede que usando formulas como COINCIDIR, BUSCARV o BUSCARH en las macros sea más rápido, no lo he probado pero también puede que sean más difíciles de entender las macro que ya de por si lo son, sobre todo si las hace otro.

Si falta alguna cosa por explicar dímelo y te lo cuento.