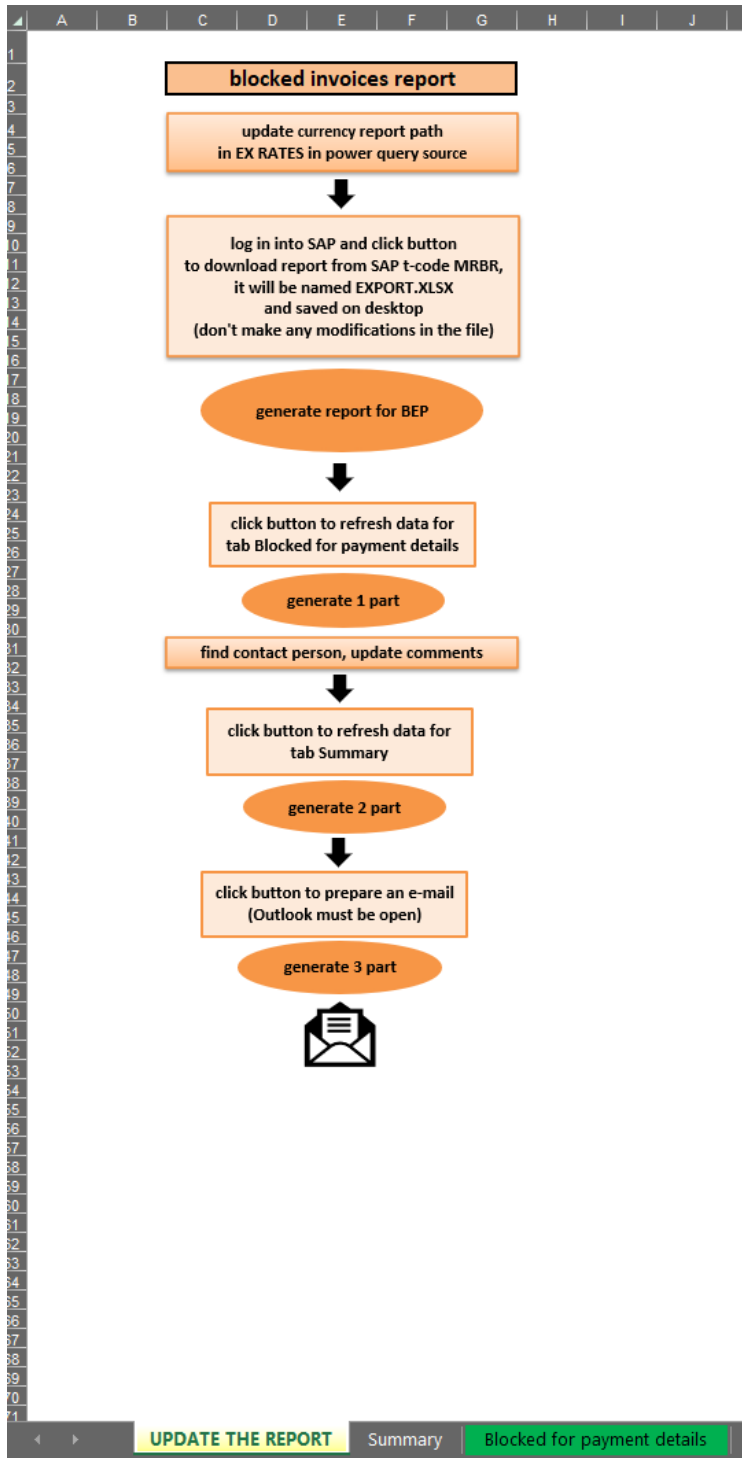


Power Query + Power Pivot + Macros => Macro for blocked invoices report for BEP

*** Company data was hidden, among others: by blurring them, replacing them in the code with xxx**

The report consists of 3 tabs in Excel. In the first tab we have detailed instructions on what to do step by step, and in the second and third tab we have parts of the report. After completing all steps, a separate file with today's date will be created on the desktop with the final report and an appropriate e-mail will be generated for sending with the attached report.



	A	B	C	D	E	F	G
1							
2	Invoices on hold per exception reason						
3		Values	Payment block reason				
4		Sum of Amount in USD		Count of Invoice Document No.	Total Sum of Amount in USD	Total Count of Invoice Document	
5	Owner	LIV - Price Discrepancy	LIV - Quantity Discrepancy	LIV - Price Discrepancy	LIV - Quantity Discrepancy		
6							
7	Grand Total						
8							
9							
10							
11							
12							
13							
14							
15							
16	Invoices on hold per aging						
17	Count of Invoice Aging Document No.						
18	Owner	[0] Not due	[1] 1-30 days	[2] 31-60 days	[3] 61-90 days	[4] 90-180 days	Grand Total
19							
20	Grand Total						
21							
22							

UPDATE THE REPORT

Summary

Blocked for payment details

(Power Pivot)

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
1	Company Code	Invoicing Party	Invoice Document No.	Invoice Item	Amount	Currency	Posting Date	Purchasing Document	Item	PO+Item	Material	Plant	Difference Quantity	Difference Value	Baseline Payment Da	Payment Terms (60 days)	FX rate	Amount in USD	Quantity check	Price check	Payment block reason	Due date	Aging days	Aging	Owner	Comments
2																										
3																										
4																										
5																										
6																										
7																										
8																										
9																										
10																										
11																										
12																										
13																										
14																										
15																										
16																										
17																										
18																										
19																										
20																										
21																										
22																										

(Power Query)

Power Query editor:

The downloaded data from SAP was loaded into Power Query and named Block for payment details. Then, data on exchange rates called EX RATES were added and a new EX RATES USD table was manually added. Both of these tables have been merged into one main table using append, named EX RATES ALL rates, whose rate column will be used in the Block for payment details table. The data in the Block for payment details table has been appropriately transformed, including: conditional and custom columns have been added, a column from the FX RATES ALL table has been added via join

The screenshots below show the data transformations.

Queries [4]

Blocked for payment details

EX RATES

EX RATES ALL

EX RATES USD

✓

fx

= Table.PromoteHeaders(Sheet1_Sheet, [PromoteAllScalars=true])

ABC 123	Company Code	ABC 123	Invoicing Party	ABC 123	Invoice Document No.	ABC 123	Invoice Item	ABC 123	Amount	ABC 123	Currency	ABC 123	Posting Date	ABC 123	Purchase
	<div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div>		<div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div>		<div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div>		<div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div>		<div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div>		<div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div>		<div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div>		<div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div>
1															
2															
3															
4															
5															
6															
7															
8															
9															
10															
11															
12															
13															
14															
15															
16															
17															
18															
19															
20															
21															
22															
23															
24															

Query Settings

PROPERTIES

Name

Blocked for payment details

All Properties

APPLIED STEPS

Source

Navigation

Promoted Headers

Inserted Merged Column

Reordered Columns

Renamed Columns

Merged Queries

Expanded EX RATES ALL

Changed Type

Inserted Multiplication

Added Conditional Column

Added Conditional Column1

Added Custom

Filtered Rows

Added Custom1

Changed Type1

Added Custom2

Added Custom3

Added Custom4

Added Custom5

Sorted Rows

Queries [4]

Blocked for payment details

EX RATES

EX RATES ALL

EX RATES USD

✓

fx

= Table.TransformColumnTypes("#Removed Other Columns",{{"Exchange Rate", type number}})

ABC 123	Exchange rate type	ABC 123	From currency	ABC 123	To-currency	ABC 123	Valid from	1.2	Exchange Rate
	<div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div>		<div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div>		<div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div>		<div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div>		
1					USD		30.06.2024		0.27226
2					USD		30.06.2024		0.00258
3					USD		30.06.2024		0.55866
4					USD		30.06.2024		0.00116

Query Settings

PROPERTIES

Name

EX RATES

All Properties

APPLIED STEPS

Source

Navigation

Promoted Headers

Removed Other Columns

Changed Type

Queries [4]

Blocked for payment details

EX RATES

EX RATES ALL

EX RATES USD

fx

= Table.TransformColumnTypes(Source,{{"Exchange rate type", type text}, {"From currency", type text}, {"To-currency", type text}, {"Valid from", type text}, {"Exchange Rate", Int64.Type}})

ABC Exchange rate type	ABC From currency	ABC To-currency	ABC Valid from	123 Exchange Rate
<div>Valid 0%</div> <div>Error 0%</div> <div>Empty 100%</div>	<div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div>	<div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div>	<div>Valid 0%</div> <div>Error 0%</div> <div>Empty 100%</div>	<div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div>
1	USD	USD		1

Query Settings

PROPERTIES

Name

EX RATES USD

All Properties

APPLIED STEPS

Source

Changed Type

Queries [4]

Blocked for payment details

EX RATES

EX RATES ALL

EX RATES USD

fx

= Table.Combine({#"EX RATES", #"EX RATES USD"})

ABC Exchange rate type	ABC From currency	ABC To-currency	ABC Valid from	1.2 Exchange Rate
<div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div>	<div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div>	<div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div>	<div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div>	<div>Valid 100%</div> <div>Error 0%</div> <div>Empty 0%</div>
1		USD	30.06.2024	0.27226
2		USD	30.06.2024	0.00258
3		USD	30.06.2024	0.55866
4		USD	30.06.2024	0.00116

Query Settings

PROPERTIES

Name

EX RATES ALL

All Properties

APPLIED STEPS

Source

Queries [4]

Blocked for payment details

EX RATES

EX RATES ALL

EX RATES USD

fx

= Table.AddColumn(#"Promoted Headers", "PO+Item", each Text.Combine({[Purchasing Document], [Item]}), type text)

ABC Material	ABC Plant	ABC Difference Quantity	ABC Difference Value	ABC Baseline Payment Dte	ABC Days 1	ABC PO+Item
1						
2						
3						

Query Settings

PROPERTIES

Name

Blocked for payment details

All Properties

APPLIED STEPS

Source

Navigation

Promoted Headers

Inserted Merged Column

Queries [4]

Blocked for payment detailsEX RATESEX RATES ALLEX RATES USD

fx

= Table.NestedJoin(#"Renamed Columns", {"Currency"}, #"EX RATES ALL", {"From currency"}, "EX RATES ALL", JoinKind.LeftOuter)

ABC123Material

ABC123Plant

ABC123Difference Quantity

ABC123Difference Value

ABC123Baseline Payment Date

ABC123Payment Terms (60 days)

EX RATES ALL

1

2

3

4

5

6

7

Query Settings

PROPERTIES

NameBlocked for payment details

All Properties

APPLIED STEPS

Source

Navigation

Promoted Headers

Inserted Merged Column

Reordered Columns

Renamed Columns

Merged Queries

Expanded EX RATES ALL

Queries [4]

Blocked for payment detailsEX RATESEX RATES ALLEX RATES USD

fx

= Table.AddColumn(#"Changed Type", "Amount in USD", each [Amount] * [FX rate], type number)

ABC123Plant

1.2Difference Quantity

1.2Difference Value

Baseline Payment Date

1.2Payment Terms (60 days)

1.2FX rate

1.2Amount in USD

1

2

3

4

5

6

7

8

9

Query Settings

PROPERTIES

NameBlocked for payment details

All Properties

APPLIED STEPS

Source

Navigation

Promoted Headers

Inserted Merged Column

Reordered Columns

Renamed Columns

Merged Queries

Expanded EX RATES ALL

Changed Type

Inserted Multiplication

- Queries [4]
- Blocked for payment details

EX RATES

EX RATES ALL

EX RATES USD

✕

✓

$\frac{f}{x}$

= Table.AddColumn("#Inserted Multiplication", "Quantity check", each if [Difference Quantity] = 0 then "-" else if [Difference Quantity] < 0 then "LIV - Quantity Discrepancy" else if [Difference Quantity] > 0 then "LIV - Quantity Discrepancy" else "check")

	1.2 Difference Quantity	1.2 Difference Value	Baseline Payment Date	1.2 Payment Terms (60 days)	1.2 FX rate	1.2 Amount in USD	Quantity check
	<div><div>Valid</div><div>Error</div><div>Empty</div></div> 100% <div>0%</div> <div>0%</div>	<div><div>Valid</div><div>Error</div><div>Empty</div></div> 100% <div>0%</div> <div>0%</div>	<div><div>Valid</div><div>Error</div><div>Empty</div></div> 100% <div>0%</div> <div>0%</div>	<div><div>Valid</div><div>Error</div><div>Empty</div></div> 100% <div>0%</div> <div>0%</div>	<div><div>Valid</div><div>Error</div><div>Empty</div></div> 100% <div>0%</div> <div>0%</div>	<div><div>Valid</div><div>Error</div><div>Empty</div></div> 100% <div>0%</div> <div>0%</div>	<div><div>Valid</div><div>Error</div><div>Empty</div></div> 100% <div>0%</div> <div>0%</div>
1							
2							
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							
13							
14							
15							
16							
17							
18							
19							

Add Conditional Column

Add a conditional column that is computed from the other columns or values.

New column name

Quantity check

Column Name	Operator	Value	Output
If Difference Quantity	equals	ABC123 0	Then ABC123 -
Else If Difference Quantity	is less than	ABC123 0	Then ABC123 LIV - Quantity Discrepancy
Else If Difference Quantity	is greater than	ABC123 0	Then ABC123 LIV - Quantity Discrepancy

Add Clause

Else

ABC123

check

OK

Cancel

Query Settings

PROPERTIES

Name

Blocked for payment details

All Properties

APPLIED STEPS

Source

Navigation

Promoted Headers

Inserted Merged Column

Reordered Columns

Renamed Columns

Merged Queries

Expanded EX RATES ALL

Changed Type

Inserted Multiplication

Added Conditional Column

Added Conditional Column1

Added Custom

Filtered Rows

Added Custom1

Changed Type1

Added Custom2

Added Custom3

Added Custom4

Added Custom5

Sorted Rows

Queries [4]

- Blocked for payment details
- EX RATES
- EX RATES ALL
- EX RATES USD

= Table.AddColumn(#"Added Conditional Column", "Price check", each if [Difference Value] = 0 then "-" else if [Difference Value] < 0 then "LIV - Price Discrepancy" else if [Difference Value] > 0 then "LIV - Price Discrepancy" else "check")

	1.2 Difference Value	Baseline Payment Date	1.2 Payment Terms (60 days)	1.2 FX rate	1.2 Amount in USD	ABC 123 Quantity check	ABC 123 Price check
	<div><div>Valid</div><div>Error</div><div>Empty</div></div> 100% <div>0%0%</div>	<div><div>Valid</div><div>Error</div><div>Empty</div></div> 100% <div>0%0%</div>	<div><div>Valid</div><div>Error</div><div>Empty</div></div> 100% <div>0%0%</div>	<div><div>Valid</div><div>Error</div><div>Empty</div></div> 100% <div>0%0%</div>	<div><div>Valid</div><div>Error</div><div>Empty</div></div> 100% <div>0%0%</div>	<div><div>Valid</div><div>Error</div><div>Empty</div></div> 100% <div>0%0%</div>	<div><div>Valid</div><div>Error</div><div>Empty</div></div> 100% <div>0%0%</div>

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18
- 19

Add Conditional Column

Add a conditional column that is computed from the other columns or values.

New column name
Price check

	Column Name	Operator	Value	Then	Output
If	Difference Value	equals	ABC 123 0	Then	ABC 123 -
Else If	Difference Value	is less than	ABC 123 0	Then	ABC 123 LIV - Price Discrepancy
Else If	Difference Value	is greater than	ABC 123 0	Then	ABC 123 LIV - Price Discrepancy

Add Clause

Else
ABC 123 check

OK Cancel

Query Settings

PROPERTIES

Name
Blocked for payment details

All Properties

APPLIED STEPS

- Source
- Navigation
- Promoted Headers
- Inserted Merged Column
- Reordered Columns
- Renamed Columns
- Merged Queries
- Expanded EX RATES ALL
- Changed Type
- Inserted Multiplication
- Added Conditional Column
- Added Conditional Column1**
- Added Custom
- Filtered Rows
- Added Custom1
- Changed Type1
- Added Custom2
- Added Custom3
- Added Custom4
- Added Custom5
- Sorted Rows

Queries [4]

- Blocked for payment details
- EX RATES
- EX RATES ALL
- EX RATES USD

```

= Table.AddColumn("#Added Conditional Column1", "Payment block reason", each if [Quantity check] = "-" and [Price check] = "LIV - Price Discrepancy" then "LIV - Price Discrepancy"
else if [Quantity check] = "LIV - Quantity Discrepancy" and [Price check] = "-" then "LIV - Quantity Discrepancy"
else if [Quantity check] = "LIV - Quantity Discrepancy" and [Price check] = "LIV - Price Discrepancy" then "LIV - Quantity and Price Discrepancy"
else if [Quantity check] = "-" and [Price check] = "-" then "no difference"
else "check")

```

	Baseline Payment Date	1.2 Payment Terms (60 days)	1.2 FX rate	1.2 Amount in USD	Quantity check	Price check	Payment block reason
	Valid Error Empty	100% 0% 0%	Valid Error Empty	100% 0% 0%	Valid Error Empty	100% 0% 0%	Valid Error Empty
1							LIV - Price Discrepancy
2							LIV - Quantity Discrepancy
3							LIV - Quantity Discrepancy
4							LIV - Quantity Discrepancy
5							LIV - Quantity Discrepancy
6							LIV - Quantity Discrepancy
7							LIV - Quantity Discrepancy
8							LIV - Quantity Discrepancy
9							LIV - Quantity Discrepancy
10							LIV - Quantity Discrepancy
11							LIV - Quantity Discrepancy
12							LIV - Quantity Discrepancy
13							LIV - Quantity Discrepancy
14							LIV - Quantity Discrepancy
15							LIV - Quantity Discrepancy
16							LIV - Quantity Discrepancy
17							LIV - Quantity Discrepancy
18							LIV - Quantity Discrepancy
19							LIV - Quantity Discrepancy

Custom Column

Add a column that is computed from the other columns.

New column name
Payment block reason

Custom column formula

```

= if [Quantity check] = "-" and [Price check] = "LIV - Price Discrepancy" then "LIV - Price Discrepancy"
else if [Quantity check] = "LIV - Quantity Discrepancy" and [Price check] = "-" then "LIV - Quantity Discrepancy"
else if [Quantity check] = "LIV - Quantity Discrepancy" and [Price check] = "LIV - Price Discrepancy" then "LIV - Quantity and Price Discrepancy"
else if [Quantity check] = "-" and [Price check] = "-" then "no difference"
else "check"

```

Learn about Power Query formulas

✓ No syntax errors have been detected.

Available columns

- Company Code
- Invoicing Party
- Invoice Document No.
- Invoice Item
- Amount
- Currency
- Posting Date

<< Insert

Query Settings

PROPERTIES

Name
Blocked for payment details

APPLIED STEPS

- Source
- Navigation
- Promoted Headers
- Inserted Merged Column
- Reordered Columns
- Renamed Columns
- Merged Queries
- Expanded EX RATES ALL
- Changed Type
- Inserted Multiplication
- Added Conditional Column
- Added Conditional Column1
- Added Custom**
- Filtered Rows
- Added Custom1
- Changed Type1
- Added Custom2
- Added Custom3
- Added Custom4
- Added Custom5
- Sorted Rows

Queries [4]

- Blocked for payment details
- EX RATES
- EX RATES ALL
- EX RATES USD

```
= Table.SelectRows("#Added Custom", each ([Payment block reason] <> "no difference"))
```

	Baseline Payment Date	1.2 Payment Terms (60 days)	1.2 FX rate	1.2 Amount in USD	ABC 123 Quantity check	ABC 123 Price check	ABC 123 Payment block reason
	Valid Error Empty	100% 0% 0%	Valid Error Empty	100% 0% 0%	Valid Error Empty	100% 0% 0%	Valid Error Empty
1							LIV - Price Discrepancy
2							LIV - Quantity Discrepancy
3							LIV - Quantity Discrepancy
4							LIV - Quantity Discrepancy
5							LIV - Quantity Discrepancy
6							LIV - Quantity Discrepancy
7							LIV - Quantity Discrepancy
8							LIV - Quantity Discrepancy
9							LIV - Quantity Discrepancy
10							LIV - Quantity Discrepancy
11							LIV - Quantity Discrepancy
12							LIV - Quantity Discrepancy
13							LIV - Quantity Discrepancy

Filter Rows

Apply one or more filter conditions to the rows in this table.

☒ Basic ☐ Advanced

Keep rows where 'Payment block reason'

does not equal All no difference

☒ And ☐ Or

Enter or select a value

OK Cancel

Query Settings

PROPERTIES

Name

Blocked for payment details

All Properties

APPLIED STEPS

- Source
- Navigation
- Promoted Headers
- Inserted Merged Column
- Reordered Columns
- Renamed Columns
- Merged Queries
- Expanded EX RATES ALL
- Changed Type
- Inserted Multiplication
- Added Conditional Column
- Added Conditional Column1
- Added Custom
- Filtered Rows**

Queries [4]

Blocked for payment details

EX RATES

EX RATES ALL

EX RATES USD

Table.AddColumn

["Filtered Rows"]

Due date

each Date.AddDays([Baseline Payment Date], [#"Payment Terms (60 days)"])

	1.2 Payment Terms (60 days)	1.2 FX rate	1.2 Amount in USD	ABC 123 Quantity check	ABC 123 Price check	ABC 123 Payment block reason	ABC 123 Due date
	Valid 100% Error 0% Empty 0%	Valid 100% Error 0% Empty 0%	Valid 100% Error 0% Empty 0%	Valid 100% Error 0% Empty 0%	Valid 100% Error 0% Empty 0%	Valid 100% Error 0% Empty 0%	Valid 100% Error 0% Empty 0%
1							
2							
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							
13							
14							

Custom Column

Add a column that is computed from the other columns.

New column name

Due date

Custom column formula

= Date.AddDays([Baseline Payment Date], [#"Payment Terms (60 days)"])

Available columns

Company Code

Invoicing Party

Invoice Document No.

Invoice Item

Amount

Currency

Posting Date

<< Insert

Learn about Power Query formulas

✓ No syntax errors have been detected.

OK Cancel

Query Settings

PROPERTIES

Name

Blocked for payment details

All Properties

APPLIED STEPS

Source

Navigation

Promoted Headers

Inserted Merged Column

Reordered Columns

Renamed Columns

Merged Queries

Expanded EX RATES ALL

Changed Type

Inserted Multiplication

Added Conditional Column

Added Conditional Column1

Added Custom

Filtered Rows

Added Custom1

Changed Type1

Queries [4]

- Blocked for payment details
- EX RATES
- EX RATES ALL
- EX RATES USD

fx = Table.AddColumn(#"Changed Type1", "Aging days", each Duration.Days(Date.From(DateTime.LocalNow()) - [Due date]))

	1.2 FX rate	1.2 Amount in USD	ABC 123 Quantity check	ABC 123 Price check	ABC 123 Payment block reason	Due date	ABC 123 Aging days
	100% 0% 0%	Valid 100% 0% 0%	Valid 100% 0% 0%	Valid 100% 0% 0%	Valid 100% 0% 0%	Valid 100% 0% 0%	Valid 100% 0% 0%
1							
2							
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							
13							
14							
15							
16							

Query Settings

PROPERTIES

Name
Blocked for payment details

All Properties

APPLIED STEPS

- Source
- Navigation
- Promoted Headers
- Inserted Merged Column
- Reordered Columns
- Renamed Columns
- Merged Queries
- Expanded EX RATES ALL
- Changed Type
- Inserted Multiplication
- Added Conditional Column
- Added Conditional Column1
- Added Custom
- Filtered Rows
- Added Custom1
- Changed Type1
- Added Custom2**

Custom Column

Add a column that is computed from the other columns.

New column name
Aging days

Custom column formula ⓘ
= Duration.Days(Date.From(DateTime.LocalNow()) - [Due date])

Available columns

- Company Code
- Invoicing Party
- Invoice Document No.
- Invoice Item
- Amount
- Currency
- Posting Date

<< Insert

Learn about Power Query formulas

✓ No syntax errors have been detected.

OK Cancel

- Queries [4]
- Blocked for payment details

EX RATES

EX RATES ALL

EX RATES USD

✓

fx

= Table.AddColumn("#Added Custom2", "Aging", each if [Aging days] <= 0 then "[0] Not due" else if [Aging days] <= 30 then "[1] 1-30 days" else if [Aging days] <= 60 then "[2] 31-60 days" else if [Aging days] <= 90 then "[3] 61-90 days" else if [Aging days] <= 180 then "[4] 90-180 days" else if [Aging days] <= 360 then "[5] 180-360 days" else "[6] >360 days")

	1.2 Amount in USD	ABC 123 Quantity check	ABC 123 Price check	ABC 123 Payment block reason	Due date	ABC 123 Aging days	ABC 123 Aging
	100% 0% 0%	Valid Error Empty	100% 0% 0%	Valid Error Empty	100% 0% 0%	Valid Error Empty	100% 0% 0%
1							
2							
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							
13							
14							
15							
16							
17							
18							
19							

Add Conditional Column

Add a conditional column that is computed from the other columns or values.

New column name

Aging

Column Name	Operator	Value	Output
if Aging days	is less than or equ...	ABC 123 0	Then ABC 123 [0] Not due
Else If Aging days	is less than or equ...	ABC 123 30	Then ABC 123 [1] 1-30 days
Else If Aging days	is less than or equ...	ABC 123 60	Then ABC 123 [2] 31-60 days
Else If Aging days	is less than or equ...	ABC 123 90	Then ABC 123 [3] 61-90 days
Else If Aging days	is less than or equ...	ABC 123 180	Then ABC 123 [4] 90-180 days
Else If Aging days	is less than or equ...	ABC 123 360	Then ABC 123 [5] 180-360 days

Add Clause

Else

ABC 123 [6] >360 days

OKCancel

Query Settings

PROPERTIES

Name

Blocked for payment details

All Properties

APPLIED STEPS

Source

Navigation

Promoted Headers

Inserted Merged Column

Reordered Columns

Renamed Columns

Merged Queries

Expanded EX RATES ALL

Changed Type

Inserted Multiplication

Added Conditional Column

Added Conditional Column1

Added Custom

Filtered Rows

Added Custom1

Changed Type1

Added Custom2

✕ Added Custom3

Added Custom4

Added Custom5

Sorted Rows

- Queries [4]
- Blocked for payment details
 - EX RATES
 - EX RATES ALL
 - EX RATES USD

✓

fx

= Table.AddColumn(#"Added Custom3", "Owner", each "")

	ABC 123	Quantity check	ABC 123	Price check	ABC 123	Payment block reason	Due date	ABC 123	Aging days	ABC 123	Aging	ABC 123	Owner
	100%	Valid	100%	Valid	100%	Valid	100%	Valid	100%	Valid	100%	Valid	100%
	0%	Error	0%	Error	0%	Error	0%	Error	0%	Error	0%	Error	0%
	0%	Empty	0%	Empty	0%	Empty	0%	Empty	0%	Empty	0%	Empty	0%
1													
2													
3													
4													
5													
6													
7													
8													
9													
10													
11													
12													
13													
14													
15													
16													
17													
18													
19													

Custom Column

Add a column that is computed from the other columns.

New column name

Owner

Custom column formula

= ""

Available columns

Company Code

Invoicing Party

Invoice Document No.

Invoice Item

Amount

Currency

Posting Date

<< Insert

Learn about Power Query formulas

✓ No syntax errors have been detected.

OK

Cancel

Query Settings

PROPERTIES

Name

Blocked for payment details

All Properties

APPLIED STEPS

Source

Navigation

Promoted Headers

Inserted Merged Column

Reordered Columns

Renamed Columns

Merged Queries

Expanded EX RATES ALL

Changed Type

Inserted Multiplication

Added Conditional Column

Added Conditional Column1

Added Custom

Filtered Rows

Added Custom1

Changed Type1

Added Custom2

Added Custom3

Added Custom4

Added Custom5

Sorted Rows

Project - VBAProject

VBAProject (Macro for blocked)

Microsoft Excel Objects

Modules

Payment_Block_R_Report

SAPreports

Properties - SAPreports

SAPreports Module

Alphabetic Categorized

(Name) SAPreports

(General)

SAPreport_BEP

Option Explicit

'macro for download SAP report

'turn on the sap library C:\Program Files (x86)\SAP\FrontEnd\SAPgui\sapfews.ocx

'log into SAP before macro execution

Public SapGuiAuto

Public objGui As GuiApplication

Public objConn As GuiConnection

Public objSess As GuiSession

Sub SAPreport_BEP()

Application.ScreenUpdating = False

Set SapGuiAuto = GetObject("SAPGUI")

Set objGui = SapGuiAuto.GetScriptingEngine

Set objConn = objGui.Children(0)

Set objSess = objConn.Children(0)

objSess.FindById("wnd[0]").Maximize

objSess.FindById("wnd[0]/tbar[0]/okcd").Text = "mrbr"

objSess.FindById("wnd[0]").SendVKey 0

'''FOR BELGIUM COMPANY CODES AND FOR INTERCOMPANY VENDORS

objSess.FindById("wnd[0]/usr/radPA_XMANU").Select

objSess.FindById("wnd[0]/usr/radPA_XSPGR").Select

objSess.FindById("wnd[0]/usr/ctxtSO_BUKRS-LOW").Text = ""

objSess.FindById("wnd[0]/usr/txtSO_BELNR-LOW").Text = ""

objSess.FindById("wnd[0]/usr/txtSO_GJAHR-LOW").Text = ""

objSess.FindById("wnd[0]/usr/ctxtSO_BUDAT-LOW").Text = ""

objSess.FindById("wnd[0]/usr/ctxtSO_FAEEL-LOW").Text = ""

objSess.FindById("wnd[0]/usr/ctxtSO_EKGRP-LOW").Text = ""

objSess.FindById("wnd[0]/usr/txtSO_USNAM-LOW").Text = ""

objSess.FindById("wnd[0]/usr/ctxtPA_VARIA").Text = ""

objSess.FindById("wnd[0]/usr/ctxtSO_LIFNR-LOW").Text = "XX*" 'selecting intercompany vendors

'selecting belgium company codes

objSess.FindById("wnd[0]/usr/btn\$SO_BUKRS \$ APP \$-VALU PUSH").Press

objSess.FindById("wnd[1]/usr/tabsTAB_STRIP/tabpsIVA/ssubSCREEN_HEADER:SAPLALDB:3010/tblSAPLALDBSINGLE/ctxtRSCSEL_255-SLOW_I[1,0]").Text = "XXXX"

objSess.FindById("wnd[1]/usr/tabsTAB_STRIP/tabpsIVA/ssubSCREEN_HEADER:SAPLALDB:3010/tblSAPLALDBSINGLE/ctxtRSCSEL_255-SLOW_I[1,1]").Text = "XXXX"

objSess.FindById("wnd[1]/usr/tabsTAB_STRIP/tabpsIVA/ssubSCREEN_HEADER:SAPLALDB:3010/tblSAPLALDBSINGLE/ctxtRSCSEL_255-SLOW_I[1,2]").Text = "XXXX"

objSess.FindById("wnd[1]/tbar[0]/btn[8]").Press 'ok

objSess.FindById("wnd[0]/tbar[1]/btn[8]").Press 'execute

.....

'layout /VIM BLOCK if this layout is not on the list to choose from first, the wrong layout will be selected by vba

objSess.FindById("wnd[0]/tbar[1]/btn[33]").Press

objSess.FindById("wnd[1]/usr/ssubD0500_SUBSCREEN:SAPLSLVC_DIALOG:0501/cntlG51_CONTAINER/shellcont/shell").CurrentCellColumn = "TEXT"

objSess.FindById("wnd[1]/usr/ssubD0500_SUBSCREEN:SAPLSLVC_DIALOG:0501/cntlG51_CONTAINER/shellcont/shell").SelectedRows = "0"

objSess.FindById("wnd[1]/usr/ssubD0500_SUBSCREEN:SAPLSLVC_DIALOG:0501/cntlG51_CONTAINER/shellcont/shell").ClickCurrentCell

'export to spreadsheet

objSess.FindById("wnd[0]/mbar/menu[0]/menu[3]/menu[1]").Select

objSess.FindById("wnd[1]/tbar[0]/btn[0]").Press

objSess.FindById("wnd[1]/usr/ctxdY_PATH").Text = "C:\Users\xxx\OneDrive - xxx\Desktop"

objSess.FindById("wnd[1]/usr/ctxdY_FILENAME").Text = "EXPORT.XLSX"

objSess.FindById("wnd[1]/tbar[0]/btn[0]").Press

'close transaction and return to sap menu

objSess.FindById("wnd[0]/tbar[0]/btn[12]").Press

objSess.FindById("wnd[0]/tbar[0]/btn[12]").Press

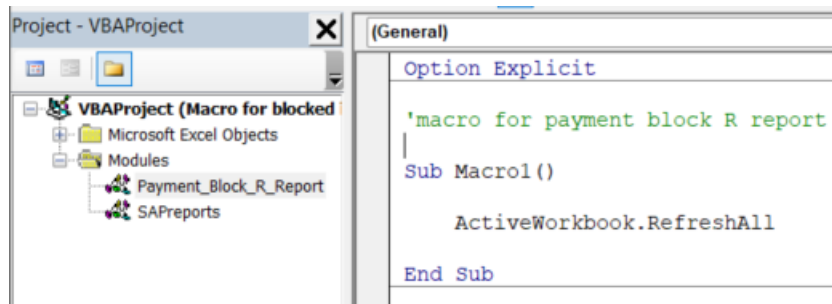
.....

Application.ScreenUpdating = True

MsgBox "Done", vbInformation, "Information"

End Sub

Code for button: generate 1 part



Code for button: generate 2 part

Project - VBAProject

(General) Macro1

VBAProject (Macro for blocked
 Microsoft Excel Objects
 Modules
 Payment_Block_R_Report
 SAPReports

Properties - Payment_Block_R_Report
 Payment_Block_R Module
 Alphabetic Categorized
 (Name) Payment_Block_R_Report

```

Sub Macro2()

    Application.ScreenUpdating = False

    Dim PmtBlockedSheet As Worksheet
    Set PmtBlockedSheet = Workbooks("Macro for blocked invoices report for BEP.xlsm").Sheets("Blocked for payment details")

    PmtBlockedSheet.Activate

    'turn off existing filters
    If PmtBlockedSheet.AutoFilterMode Then
        PmtBlockedSheet.AutoFilterMode = False
    End If

    Dim SummarySheet As Worksheet
    Set SummarySheet = Workbooks("Macro for blocked invoices report for BEP.xlsm").Sheets("Summary")

    SummarySheet.Activate

    'refresh data range of pivot tables

    Dim RangeData As Range
    Set RangeData = Worksheets("Blocked for payment details").Range("A1").CurrentRegion

    ActiveSheet.PivotTables("PivotTable1").PivotCache.Refresh
    ActiveSheet.PivotTables("PivotTable2").PivotCache.Refresh

    'save, open, close final report file

    'define paths and file names
    Dim DesktopPath As String
    DesktopPath = Environ("USERPROFILE") & "\OneDrive - XXX\Desktop\"
    Dim FinalReportName As String
    Dim FinalReportPath As String
    FinalReportName = "Invoices blocked for BEP" & " " & Format(Date, "dd.mm.yyyy") & ".xlsx"
    FinalReportPath = DesktopPath & FinalReportName

    'save the current excel file
    ActiveWorkbook.Save

    'create a new workbook
    Dim NewWorkbook As Workbook
    Set NewWorkbook = Workbooks.Add

    ' copy data from the existing sheets to the new workbook
    'copy "Summary" sheet
    ThisWorkbook.Sheets("Summary").Cells.Copy Destination:=NewWorkbook.Sheets(1).Cells
    NewWorkbook.Sheets(1).Name = "Summary"
    'add and copy "Blocked for payment details" sheet
    NewWorkbook.Sheets.Add After:=NewWorkbook.Sheets(1)
    ThisWorkbook.Sheets("Blocked for payment details").Cells.Copy Destination:=NewWorkbook.Sheets(2).Cells
    NewWorkbook.Sheets(2).Name = "Blocked for payment details"

    'save the new workbook
    NewWorkbook.SaveAs Filename:=FinalReportPath, FileFormat:=xlOpenXMLWorkbook, CreateBackup:=False
    'close the new workbook
    NewWorkbook.Close SaveChanges:=False

    Application.ScreenUpdating = True

    'display message that file is ready
    MsgBox "File is ready to sent", vbInformation, "Information"

End Sub

```

Code for button: generate 3 part

Project - VBAProject X (General) Macro1

VBAProject (Macro for blocked)

- Microsoft Excel Objects
- Modules
 - Payment_Block_R_Report
 - SAPreports

Properties - Payment_Block_R_Report X

Payment_Block_R Module

Alphabetic Categorized

(Name) Payment_Block_R_Report

```
Sub Macro3()  
  
    Workbooks("Macro for blocked invoices report for BEP.xlsm").Sheets("Summary").Activate  
  
    'sending e-mail  
  
    Dim OutApp As Object  
    Dim OutMail As Object  
    Set OutApp = CreateObject("Outlook.Application")  
    Set OutMail = OutApp.CreateItem(0)  
  
    Dim TodaysDate As String  
    Dim DeadlineDate As String  
    'calculate deadline date (today's date + 6 days)  
    DeadlineDate = Format(DateAdd("d", 6, Date), "dd/mm/yyyy")  
  
    'generate report name based on selection  
    Dim DesktopPath As String  
    DesktopPath = Environ("USERPROFILE") & "\OneDrive - XXX\Desktop\  
    Dim FinalReportName As String  
    Dim FinalReportPath As String  
    FinalReportName = "Invoices blocked for BEP" & " " & Format(Date, "dd.mm.yyyy") & ".xlsx"  
    FinalReportPath = DesktopPath & FinalReportName  
  
    'create a list of email addresses for the "To" field  
  
    Dim c As Byte  
    c = 19 ' we start from row 19,  
    ' "A" column A  
  
    Dim Recipients As String 'variable to store email addresses  
  
    Do Until Range("A" & c).Value = "Grand Total" 'run the loop until it encounters "Grand Total"  
        If Range("A" & c).Value <> "(blank)" Then 'condition to disregard "(blank)"  
            If Recipients <> "" Then  
                Recipients = Recipients & "; "  
            End If  
            Recipients = Recipients & Workbooks("Macro for blocked invoices report for BEP.xlsm").Sheets("Summary").Range("A" & c).Value  
            End If  
        c = c + 1  
    Loop  
  
    'parts of e-mail  
  
    With OutMail  
  
        .Display 'must be earlier, otherwise the footer disappears  
  
        .SentOnBehalfOfName = "XXXX"  
        .To = Recipients  
        .CC = "XXXX; XXXX; XXXX"  
        .Subject = "Invoices blocked for BEP as of " & Format(Date, "dd.mm.yyyy")  
  
        .HTMLBody = "Dear All," & "<br><br>" &  
        "Please find enclosed MRBR report including the list of documents automatically R-blocked due to quantity/price discrepancy." & "<br>" &  
        "Attached you will also find the latest email communication regarding some holds investigation." & "<br><br>" &  
        "In Details tab you can find the items referring to respective user (column Y)." & "<br><br>" &  
        "Could you please let me know by " & DeadlineDate & " which of them can be released from holds?" & "<br><br>" &  
        OutMail.HTMLBody  
  
        .Attachments.Add FinalReportPath  
  
    End With  
  
    Set OutMail = Nothing  
    Set OutApp = Nothing  
  
End Sub
```