

EXCEL STYLE REPORT, A BUSINESS INTELLIGENCE TOOL, APPLIED IN STOCK MANAGEMENT

Horia Demian, Maria Madela Abrudan, Mihai Anton

Universitatea din Oradea

Facultatea de Stiinte Economice

Email: horia_demian@yahoo.com

Key words: business intelligence, stock management

Abstract

In this paper we will present a business intelligence tool which can help any accountant, or manager, in everyday work. A first step in making such a tool consists in the calculations of different indexes. There are a lot of indexes which can be calculated and can offer information regarding the company's state, evolution during a period of time and the prediction of future results.

In this paper we will present a business intelligence tool which can help any accountant, or manager, in everyday work. A first step in making such a tool consists in the calculations of different indexes. There are a lot of indexes which can be calculated and can offer information regarding the company's state, evolution during a period of time and the prediction of future results.

However these calculations need a lot of time, and present a high level of complexity. The major problem is represented by the lack of this information in real time, for crisis situation. If we don't have these indexes, we can not properly evaluate the impact of our decisions.

To improve this process of calculations, we created a tool which can dynamically report any previously defined index. This tool can export data into Microsoft Excel files.

Microsoft Excel permits the evaluation of different kind of expression which contains substractions, additions and other operations and functions. These functions are categorized in financial, mathematical, statistical and so on. Because advanced users are very familiar with excel like expression building, we created similar functions for text processing, mathematical calculations, and most importantly functions that can extract data from an accounting database, evaluate the results and display them. Also, we offered the possibility to create custom indexes and functions based on predefined ones. This tool is localized in the stock management module.

For example we created functions for retrieving stock values data:

- **SFVCP(ProductCode,[Month[,Year]])**
 - This is used to obtain the value of final stock, for a product which code has to be specified
 - First parameter named **ProductCode**, could be a string. The program looks for all that products, for which the Product Code begins with the string specified, and calculates the value for all these products.
 - If the string **ProductCode** finished with '!' character, the function will search for exact match
 - Another possibility is to search for a part of this code, if we used "%" character in the **ProductCode** string

- Another possibility is to specified more products in the **ProductCode** string, if we used “;” character between this products
- The second parameter represent the month
- The third parameter represent the year, so we can look to the current stock value, or consult the history for these products
- This function knows to translate stock value to the current currency.
- **SFCCP(ProductCode,[Month[,Year]])**
 - This function is used to obtained the quantitative stock value for an product or group of products at the end of a period
 - Like the preceding function, this one offer the possibility for user to consult the history for these products, if the user specified the **Month** parameter or **Year** parameter.
- **SIVCP(ProductCode,[Month[,Year]])**
 - This function is used to obtained the value of stock for an product or group of products at the beginning of a period
 - Like the preceding function, this one offer the possibility for user to consult the history for these products, if the user specified the **Month** parameter or **Year** parameter.
- **SICCP(ProductCode,[Month[,Year]])**
 - This function is used to obtained the quantitative stock value for an product or group of products at the beginning of a period
- **RIVCP(ProductCode,[Month[,Year]])**
 - This function is used to obtain all the input values during the current month, or a specified one, for a product or group of products
- **RICCP(ProductCode,[Month[,Year]])**
 - This function is used to obtain all the input quantities during the current month, or a specified one, for a product or group of products
- **REVCP(ProductCode,[Month[,Year]])**
 - This function is used to obtain all the output values during the current month, or a specified one, for a product or group of products
- **RECCP(ProductCode,[Month[,Year]])**
 - This function is used to obtain all the output quantities during the current month, or a specified one, for a product or group of products

The window of this tool looks like:

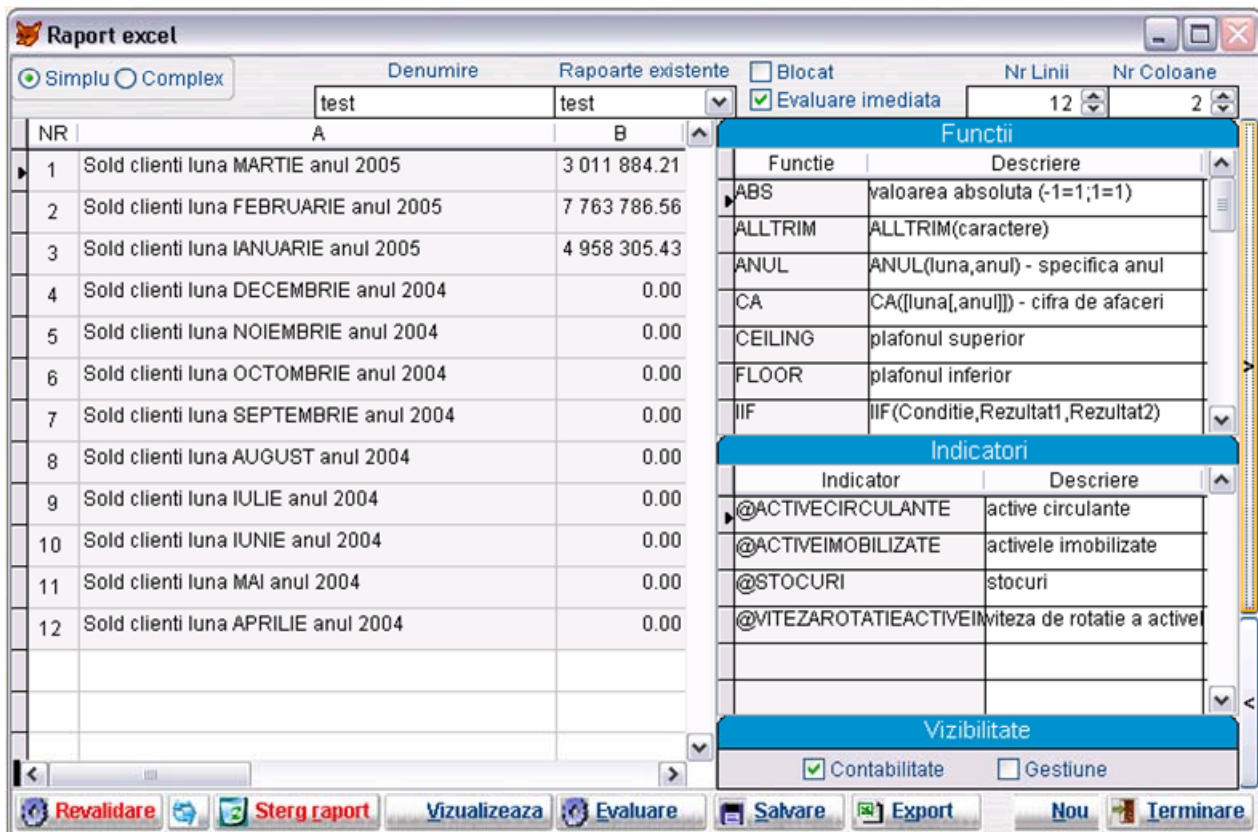


Fig. 1

Mathematical functions:

- ABS(Numerical Expression)
 - Obtains the absolute value of a number
- INT(Numerical Expression)
 - Obtains the integer part of a number
- ROUND(Numerical Expression, number of decimals)
 - Obtains the rounded value of the numerical expression to the specified number of decimal
- CEILING(Numerical Expression)
 - Obtains an integer value by rounding the numerical expression to the first above integer
- FLOOR(Numerical Expression)
 - Obtains an integer value by rounding the numerical expression to the first below integer

Text related functions:

- UPPER(string)
 - Transforms the string specified to uppercase
- LOWER(string)
 - Transforms the string specified to lowercase
- PROPER(string)
 - Transforms the first character of each word to uppercase and the rest of the word to lowercase
- LTRIM(string)
 - Eliminates the spaces from the left part of the string

- RTRIM(string)
 - Eliminates the spaces from the right part of the string
- ALLTRIM(string)
 - Eliminates the spaces from the left and right part of the string

Other functions:

- LUNA(Month)
 - Returns the name of the month number specified
- ANUL(Month,Year)
 - Returns the year
- IIF(Logical expression, Expression1, Expression2)
 - If the logical expression is true this function will return the value of Expression1, otherwise the value of Expression2
- LINIA()
 - This function return the line number on which the expression is situated

We present in the image below an example of the evolution of production during the past 12 months, for a company.

NR	A	B	C	D	E
1	Grupa de produse	ISAD		Stoc final cantitativ	Stoc final valoric
2	Aprilie	2 007.00	4.00	365.00	134 558.20
3	Martie	2 007.00	3.00	210.00	74 640.95
4	Februarie	2 007.00	2.00	351.00	148 626.95
5	Ianuarie	2 007.00	1.00	386.00	168 939.89
6	Decembrie	2 006.00	12.00	142.00	64 695.77
7	Noiembrie	2 006.00	11.00	419.00	181 453.52
8	Octombrie	2 006.00	10.00	468.00	164 795.63
9	Septembrie	2 006.00	9.00	354.00	108 001.73
10	August	2 006.00	8.00	577.00	202 148.78

Formula in celula: =SFVCP(B1,C3,B3)

Fig. 2

The expression written for each cell from column A looks like:
=luna(@lunacurenta - (linia()-2))

The expression written for each cell from column B looks like:
=anul(@lunacurenta-(linia()-2),@anulcurent)

The expression written in cell C2 looks like:

=IIF(@lunacurenta - (linia()-2)>0,@lunacurenta - (linia()-2),12-ABS((@lunacurenta - (linia()-2)%12)))

The expression written in cell D2 looks like:

=SFCCP(B1,C2,B2)

The expression written in cell E2 looks like:

=SFVCP(B1,C2,B2)

Conclusion

Because of the flexibility in defining indexes, this program represents a very important tool in everyday working environment.

Bibliography:

1. Horia Demian – „Baze de date”, University of Oradea Publishing House, 2001.
2. Lee Anne Phillips – „XML”,Editura Teora, 2001