

# INTRODUCTION

## What is the MZPack add-in?

MZ Excel Utility Pack is set of tools and functions prepared for Microsoft Excel for Microsoft Windows. The tools were designed to save some time in my daily work and make the most common jobs in Excel easier to perform.

The main idea was to create a tool which is simple and fast – you will not be overwhelmed with unnecessary features such as sophisticated windows, charming wizards or animated characters. I have focused on functionality and simplicity. Everything what you are to do is just click and things happen.

Even though I have created the utility for myself, I publish it under the terms of liberal GNU General Public License (GPL) version 3: you can use, distribute it for free. If you find a bug, or you have an idea how to make the utility even better - please send me an e-mail using the contact form on my web page

## Do I need the add-in?

If you hesitate whether you should install the MZPack, just try to answer the following questions:

1. Are you annoyed with performing the same operations in Excel over and over again?
2. Do you suspect that some of these operations can be automatized but you are not sure how to do it?
3. Do you have to work with reports prepared by non-professional people, where the data you would like to filter are marked with different color?
4. Do you want to have the real impact on how the tools you are using will look like in the future?

If you answered *yes* to at least one of them, it means that the add-in really **is** for you. There is no obscure installation steps - just download the file and use it. You can remove it at any time from your computer without leaving any piece of software on your disk.

## Requirements

In order to install and use the add-in the only thing you need is Microsoft Excel 2007 or later for Microsoft Windows. The current version of the add-in has been developed and tested both for 32-bits and 64-bits Windows. Versions of Excel that were tested on are: 2007/2010/2013.

The add-in will not work with Excel 2003 or earlier.

The add-in was not tested on Mac OS X.

## Installation & De-installation

Because you got to that point you probably had already downloaded the archive file with MZPack add-in from my site. To unpack the archive you can use any kind of ZIP program, such as 7z. The archive contains add-in file in \*.xlam format. You can save the files wherever you want on your hard disk, but please remember the path to that folder. You will use it to properly install the add-in.

There is nothing special in installing this add-in – you install it as any other kind of add-ins in Excel. Here are the steps to follow:

1. Click the Microsoft Office Button, and then click Excel Options.
2. Click the Add-Ins category.
3. In the Manage box, click Excel Add-ins, and then click Go.
4. To load an Excel add-in, do the following:
5. In the Add-Ins available box, select the check box next to the add-in that you want to load, and then click OK. If the add-in that you want to use is not listed in the Add-Ins available box, click Browse, and then locate the add-in (do you remember where you saved the un-zipped files?).

To unload and de-install MZPack add-in, do the following:

1. In the Add-Ins available box, clear the check box next to the add-in, and then click OK.
2. To remove the add-in from the Office Fluent Ribbon, restart Excel.
3. Remove the files from your hard disk.

## MACROS

### System

#### Save with backup

*This macro saves the active workbook on disk with current file name and creates its backup copy.*

After running the macro, the active workbook is saved on the disk with current file name, and also a backup copy of the workbook is created with suffix ‘(Backup copy)’ in the same folder.

### Cells

#### Format list

*Macro formats selected range of cells as a list: special formatting is given to first row to mark it as header of the list and borders to each cell is applied.*

After running the macro a question pops up about the range of cells, which contain data that are to be formatted as a list. If there is only one cell selected macro will suggest the current working range, otherwise the selected range will

be proposed by default. Of course you can change the range by selecting a new one with the mouse, or you can type it.

First row of the list will be filled with black and the text will be white and bolded. The rest of cells will be rounded with borders and standard font and size will be applied.

The macro does not change any values or formulas in the cells.

## **Conversion**

**Convert text to number** *Macro converts text values in selected range of cells to numbers.*

After running the macro a question pops up about the range of cells where values are to be converted. If there is only one cell selected then macro will suggest the current working range, otherwise the selected range will be proposed by default. Of course you can change the range by selecting a new one with the mouse or just typing it. You can give non-adjacent ranges as well.

All number values, which are recognized by Excel as text values before, will be converted to number values. If macro finds a decimal separator (a default one - usually a dot or a comma) then it will convert both integer part and decimal part - so you will get a number with decimal fraction.

**Convert number to text** *Macro converts values in selected cells to text.*

After running the macro a question pops up about the range of cells, where values will be converted. If there is only one cell selected macro will suggest the current working range, otherwise the selected range will be proposed by default. Of course you can change the range by selecting a new one with the mouse or just typing it. You can give non-adjacent ranges as well.

All number values in selected range(s) will be converted to text values.

**Change formulas into values** *This macro replaces the formulas in the selected cells with their calculated values.*

After running the macro a question pops up about the range of cells, where formulas are to be converted. If there is only one cell selected macro will suggest the current working range, otherwise the selected range will be proposed by default. Of course you can change the range by selecting a new one with the mouse or just typing it. You can give non-adjacent ranges as well.

All formulas in the selected area will be replaced with their values.

**Change numbers' signs** *This macro changes the positive numbers to negative and negative to positive.*

After running the macro a question pops up about the range of cells, where the signs of numeric values are to be changed from positive to negative and from negative to positive. If there is only one cell selected macro will suggest the current working range, otherwise the selected range will be proposed by default. Of course you can change the range by selecting a new one with the mouse or just typing it. You can give non-adjacent ranges as well.

Note, the macro does not change cells containing formulas: only those, which contain values.

**Apply formula** *Macro applies a given formula to existing formulas/values in selected cells.*

If in a cell a formula already exists, it will be put in brackets and the given formula will be applied to this new-created expression. If a cell contains a constant value then macro will add the “=” sign at the beginning and the given formula will be applied afterwards. Remember, that the original formula or value will NOT be changed - macro only adds new elements to create a new expression.

The rule for building the expressions/formulas by using the macro is very simple: a given formula HAS TO contain one of the four basic operator (+, -, \*, /, &) as the first character and only then you can add any expression, cells/ranges or worksheet function to create a new formula.

### Rows/Columns

**Delete hidden rows** *This macro deletes all hidden rows in selected range(s).*

After running the macro a question pops up about the range of cells, where hidden rows are to be removed. If there is only one cell selected macro will suggest the current working range, otherwise the selected range will be proposed by default. Of course you can change the range by selecting a new one with the mouse or just typing it. You can give non-adjacent ranges as well.

**Delete hidden columns** *This macro deletes all hidden columns in selected range(s).*

After running the macro a question pops up about the range of cells, where hidden columns are to be removed. If there is only one cell selected macro will suggest the current working range, other-wise the selected range will be proposed by default. Of course you can change the range by selecting a new one with the mouse or just typing it. You can give non-adjacent ranges as well.

### Mark duplicated cells

*Macro marks with chosen color the cells, which contain the same values in a selected range.*

After running macro will ask about the range in which the cells should be checked. The default range depends on how many cells were selected before running the macro; if only one - macro will propose the current region with data as default range, otherwise the selected range will be proposed. You can change the default range in this step by selecting a new one with your mouse or just typing it.

In the next step macro will ask for color, which should be used to mark the duplicated cells. After choosing the color macro will mark all duplicated cells and will finish its job.

### **Create workbook with selected range**

*This macro creates new workbook and pastes the previously selected range into it.*

After running the macro a question pops up about the range of cells, which are to be copied to new workbook. If there is only one cell selected, the macro will suggest the current working range, otherwise the selected range will be proposed by default. Of course you can change the range by selecting a new one with the mouse or just typing it. Remember, you cannot give non-adjacent ranges – only one area can be selected. After that, new workbook will be created, selected range copied and pasted into this new workbook in first worksheet and starting from the cell A1.

## **Sheets**

Macros included in this group perform actions on sheets in the active workbook.

### **Sort**

After running the macro will sort all sheets in alphabetical order.

### **Index**

*Macro creates index of all (visible) sheets or only selected sheets in an active worksheet adding hyperlinks to their names.*

If you select only one sheet before running the macro, there will be created ordered (by names) index of all visible sheets in the worksheet. But if you want to create an index only for a few of your sheets, then you have to select those sheets (by clicking them with pressed CTRL key) BEFORE running the macro.

The index will be created in a new sheet (called “Sheets index”) and sorted by the sheets’ names. You can create more than one index in you worksheet - next ones will be created in new sheets called respectively “Sheets index 1”, “Sheets index 2” and so on.

**REMARK:** The macro doesn’t include in index the “Chart” sheets - it is because Excel does not allow creating a hyperlink to that kind of sheets.

## **(Un)Hide**

**Unhide all** *Macro unhide all hidden sheets in an active workbook.*

After running the macro all sheets, which have been hidden before, will be visible. This utility un-hides even those sheets which were hidden using VBA editor with the property Visible set to xlSheetVeryHidden (value: 2).

**Hide selected sheets** *Macro hides selected sheets in an active workbook.*

After running the macro, the selected sheets will be hidden.

**Hide deeply selected sheets** *Macro deeply hides selected sheets in an active workbook.*

After running the macro, the selected sheets will be deeply hidden, i.e. the property Visible will be set to xlSheetVeryHidden. Normally, that operation is available only by using VBA editor, and a user cannot make such sheet visible from any Excel's menu.

The only possibility to make such sheet visible again is either setting the property Visible to xlSheetVisible in VBA editor or using another macro from the OFXMLMacros set: Sheets:Unhide all.

## **Cells**

**Add from cells** *Macro adds new sheets to active worksheet giving them names taken from selected cells.*

After running the macro you will be asked for range(s) containing names for the new sheets. If you select a range before running the macro the range will become a default proposal, otherwise the used working range will be used by default. You can either accept the default range by clicking OK or select a new range or even few ranges.

When you accept the range by clicking OK the macro will add as many new sheets as cells you have selected.

**Name by cell** *Macro gives names to the selected sheets taken from chosen cell.*

Let's assume we have three sheets and in each of those sheets we have words in cell A1 as follows: \* Data \* Summary \* Temporary

Before you run the macro remember to select all the sheets for which you would like to change names, otherwise only active sheet will have its name changed. After running the macro asks for the cell containing sheet's name.

You can choose or type the cell address and subsequently accept you choice by clicking OK. Remember to choose only one cell - not a range of cells. The select sheets will have their names changed respectively.

If it so happens that name from cell is the same like a name of already existing sheet, the macro will modify the new name by adding index number to it's end starting from number 1. E.g. if there is a name "Data" in an appointed cell and a sheet with a such name exists in an active workbook, the macro will modify the new name to "Data 1".

**Name to cell** Macro inserts sheet's name (or a few selected sheets) to chosen cell. Before you run the macro please select the sheets the names of which you would like to insert into cells. After starting, the macro asks question about destination cell (i.e. the cell which will contain the sheet's name). By default the macro use the active cell. Remember, that you should always choose single cell.

Next, macro copies sheets' names to the same cell in each of selected sheets.

**Empty headers and footers** *This macro empties all headers and footers in selected sheets.*

After running the macro, all headers and footers in the selected sheets will be removed.

## Comments

### Show

*Macro shows (unhide) all comments from cells in selected range.*

After running the macro an input box pops to select or type a range of cells where the comments are to be shown. If before running macro there was only one cell selected then macro will propose the current working range by default, otherwise the selected range will be given by default. You can change the default range by selecting or typing a new one.

### Hide

*Macro hides comments in selected cells.*

After running the macro an input box pops up to select or type a range of cells where the comments are to be hidden. If before running macro there was only one cell selected then macro will propose the current working range by default, otherwise the selected range will be given by default. You can change the default range by selecting or typing a new one.

### Create

**...new** *Macro adds the same comments to selected cells.*

After running the macro an input box will pop up to select or type a range of cells, where the comments are to be added. If before running macro there was only one cell selected then macro will propose the current working range by

default, otherwise the selected range will be given by default. You can change the default range by selecting or typing a new one.

In the next step macro will ask for the text that will be added as the comments.

**...from cells** *Macro creates new comments in selected range and fills in them with text from selected cells.*

After running the macro you have to select the range where the comments are to be inserted. In the next step you select the source for the comments: cells, from which the text will be used for comments.

Remember: both ranges (range where you put comments and range from which the text is taken for comments) have to be THE SAME SIZE.

## **Add**

**...at the beginning** *Macro adds given text to existing comments putting it at the beginning of existing text in comments for selected cells.*

After running the macro an input box will pop up to type or select a range of cells for which the text is to be added to the existing comments. If before running macro there was only one cell selected then macro will propose the current working range by default, otherwise the selected range will be given by default. You can change the default range by selecting or typing a new one.

In the next step macro will ask for the text that is to be added to the comments. It is worth to remember to put a comma, semicolon, dot or any other character at the end of the text in order to visually separate the new-added text from the old one.

If it happens that in selected cells is no comment, a new comment will be created with the given text.

**...at a position** *Macro inserts text into comments in selected cells at specified position (character number).*

After running the macro an input box will pop up to select or type a range of cells, where the comments are to be edited. If before running macro there was only one cell selected then macro will propose the current working range by default, otherwise the selected range will be given by default. You can change the default range by selecting or typing a new one.

In the next step macro asks for text, which is to be inserted into the comments. It is worth to remember to put a comma, semicolon or any other sign at the beginning and/or at the end of the text to separate it visually from the existing comments content.

Last step to typing the number of position (character) where the text should be inserted. If in one or a few selected cells there are no comments then macro will



create new ones with the given text.

**...at the end** *Macro adds new text to existing comments in a selected range putting it at the end of comments' text.*

After running the macro a question pops up about the range, where comments will be edited. If there was only one cell selected macro suggests the current working range, otherwise the selected range will be proposed by default. Of course you can change the range by selecting a new one with the mouse or just typing it.

In the next step macro asks question about the text you would like to add to existing comments. It worth to remember about space or colon/semicolon (or any other sign) at the beginning of your text so that it was separated from the old one.

If there is a cell(s) in a selected range which does not have any comment then new comment will be created with the given text.

## Text

### Add

**...at the beginning** *Macro takes a given by a user text and inserts it at the beginning of existing strings in selected cells.*

After running the macro a question pops up about the range of cells, where additional text is to be added. If there is only one cell selected macro will suggest the current working range, otherwise the selected range will be proposed by default. Of course you can change the range by selecting a new one with the mouse or just typing it. You can give non-adjacent ranges as well.

In the next step you adds the text, which is to be inserted at the beginning of each selected cell. It is worth to add at the end of the typed text a space or any other sign separating visually the new text from the old one. After clicking OK macro will add the typed text to selected cells inserting it at the beginning of ex-isting texts.

**WARNING:** If there are formulas in any of selected cells, the formulas' results will be changed to values.

**...at a position** *Macro adds a given text to all selected cells and inserts it at a given position into existing text.*

After running the macro a question pops up about the range of cells, where the spaces are to be re-moved. If there is only one cell selected macro will suggest the current working range, otherwise the selected range will be proposed by default. Of course you can change the range by selecting a new one with the mouse or just typing it. You can give non-adjacent ranges as well. Next we have

to type the text, which should be inserted to the cells. It is worth to add at the beginning and at the end of the typed text a space or any other sign separating visually the new text from the old one. In the example we'd like to insert a word after the word "Country". It means we are putting the new word at the 8th position (after the letter "y" in the word "Country"). Additionally we put a space to separate the new word from the old one. Notice, that there is no need to put a space at the end, because there is already a space between words "Country" and the name of the country.

Remember, the macro does not change cells containing formulas: only those, which contain values.

**...at the end** *Macro adds a given by a user text to selected cells and inserts it at the end of existing strings.*

After running the macro a question pops up about the range of cells, where additional text is to be added. If there is only one cell selected macro will suggest the current working range, otherwise the selected range will be proposed by default. Of course you can change the range by selecting a new one with the mouse or just typing it. You can give non-adjacent ranges as well. In the next step you add the text, which is to be inserted at the end of each selected cell. It is worth to add at the beginning of the typed text a space or any other sign separating visually the new text from the old one. After clicking OK macro will add the typed text to selected cells inserting it at the end of existing texts.

Remember, the macro does not change cells containing formulas: only those, which contain values.

## **Remove**

**...spaces at the beginning** *Macro removes all spaces, which exist BEFORE the characters in selected cells.*

After running the macro a question pops up about the range of cells, where the spaces are to be re-moved. If there is only one cell selected macro will suggest the current working range, otherwise the selected range will be proposed by default. Of course you can change the range by selecting a new one with the mouse or just typing it. You can give non-adjacent ranges as well.

After clicking OK macro will remove all spaces, which are before the first character in selected cells. Remember, the macro does not change cells containing formulas: only those, which contain values.

**...spaces at the end** *Macro removes all spaces, which are at the end of text in selected cells.*

After running the macro a question pops up about the range of cells, where the spaces are to be re-moved. If there is only one cell selected macro will suggest the current working range, otherwise the selected range will be proposed by

default. Of course you can change the range by selecting a new one with the mouse or just typing it. You can give non-adjacent ranges as well.

After clicking OK macro will remove all spaces, which existed after the last character in the selected cells.

Remember, the macro does not change cells containing formulas: only those, which contain values.

**... carriage returns** “Carriage returns” is nothing more than “Enter” signs causing the text in a cell is displayed in a few rows (wrapped text) regardless the width of the cell. This macro can remove all the signs so that the text is displayed in one row.

After running the macro a question pops up about the range of cells, where the carriage returns are to be removed. If there is only one cell selected macro will suggest the current working range, otherwise the selected range will be proposed by default. Of course you can change the range by selecting a new one with the mouse or just typing it. You can give non-adjacent ranges as well.

After clicking OK all carriage returns will be removed.

Remember, the macro does not change cells containing formulas: only those, which contain values.

## Case

**Change to UPPERcase** *This macro makes all text in your selected cells uppercase.*

After running the macro a question pops up about the range of cells, where the text is to be changed to uppercase. If there is only one cell selected macro will suggest the current working range, otherwise the selected range will be proposed by default. Of course you can change the range by selecting a new one with the mouse or just typing it. You can enter or select non-adjacent ranges as well.

Remember, the macro does not change cells containing formulas: only those, which contain values.

**Change to lowercase** *This macro makes all text in your selected cells lowercase.*

After running the macro a question pops up about the range of cells, where the text is to be changed to lowercase. If there is only one cell selected, the macro will suggest the current working range, otherwise the selected range will be proposed by default. Of course you can change the range by selecting a new one with the mouse or just typing it. You can enter or select non-adjacent ranges as well.

Remember, the macro does not change cells containing formulas: only those, which contain values.

**Start first word with uppercase** *This macro changes the first character in each of the selected cells to a capital letter. The remaining characters will be converted to lowercase.*

After running the macro a question pops up about the range of cells, where the first letter of a text is to be changed to uppercase. If there is only one cell selected, the macro will suggest the current working range, otherwise the selected range will be proposed by default. Of course you can change the range by selecting a new one with the mouse or just typing it. You can enter or select non-adjacent ranges as well.

Remember, the macro does not change cells containing formulas: only those, which contain values.

**Each word with capital letter** *This macro changes the first character in each word in selected cells to a capital letter.*

After running the macro a question pops up about the range of cells, where the first letter of words are to be changed to uppercase. If there is only one cell selected, the macro will suggest the current working range, otherwise the selected range will be proposed by default. Of course you can change the range by selecting a new one with the mouse or just typing it. You can enter or select non-adjacent ranges as well.

Remember, the macro does not change cells containing formulas: only those, which contain values.

## FUNCTIONS

New added functions are classified in the same way like original functions – into categories: Information, Math and Text. Remember that Excel recalculates sheet when any value is changed, so in case of the functions referring to cell's format you have to press CTRL+ALT+F9 to force the calculation and obtain actual results.

### Information

#### **FILEEXISTS(Path\_to\_file)**

Returns TRUE if a specified file exists. Otherwise returns FALSE.

#### **PATHEXISTS(Path\_to\_directory)**

Returns TRUE if a specified path exists. Otherwise returns FALSE.

**ISITALICS(Cell)**

Returns TRUE if a format for a given cell is italics. Otherwise it returns FALSE.

**ISUNDERLINED(Cell)**

Returns TRUE if a format for a given cell is underline. Otherwise it returns FALSE.

**ISBOLDED(Cell)**

Returns TRUE if a format for a given cell is bold. Otherwise it returns FALSE.

**GETNUMBERFORMAT(Cell)**

Returns the number format set for a given cell.

**GETFORMULA(Cell)**

Returns formula for a given cell. In case the cell contains constant (string or number) the function returns empty string (“”).

**GETHYPERLINK(Cell)**

Returns text, which is the hyperlink, which a given cell contains. If the cell does not contain a hy-perlink, the function returns empty string (“”).

**GETFONTCOLOR(Cell)**

Returns the color index for the background in a given cell. If the background color is not set the function returns a negative value.

**GETBGCOLOR(Cell)**

Returns the color index for the font in a given cell. If the font color is not set the function returns a negative value.

**GETCOMMENT(Cell)**

Returns the comment’s text for a given cell. It is advisable to switch off the option “Wrap text” (Cell format/Alignment) in order to proper display of comments with more than one line of text. If the comments contents changes press F9 for function result refreshment.

**GETSHEETNAME()**

Returns the name of the sheet in which the function is used.

**GETFONTNAME(Cell)**

Returns font's name for a given cell.

**GETWKBNAME()**

Returns the file's name for the spreadsheet in which the function is used.

**GETWBKFULLNAME()**

Returns the full path to the spreadsheet in which the function is used (together with the file's name).

**GETPATH()**

Returns the path the spreadsheet in which the function is used (WITHOUT the file's name).

**GETFONTSIZE(Cell)**

Returns the font's size for a given cell.

**EXTRACTFOLDERNAME(Cell)**

Extracts full path to the folder from a path to the file in a given cell.

**EXTRACTFILENAME(Cell)**

Extracts file's name from a path to the file in a given cell.

**SHEETSCOUNT()**

Returns the total number of sheets in an active workbook (requires no arguments).

**GETUSERNAME()**

Returns the current user's name (requires no arguments). The returned value can differ depending on the currently logged in user.

**Date & Time functions****DAYSINMONTH(Given\_Date)**

Returns the number of days in the month for a given date.

## Math functions

### **COUNTCOLOURED FONTS(Range)**

Counts cells in a given range for which the font color is different from the default (black).

### **COUNTCOLOURED BG(Range)**

Counts cells in a given range for which the background color is different from the default (no color).

### **COUNTIF FONT COLOR(Range, ColorNumber)**

Counts cells in a given range for which the font color is the same as given ColorNumber.

### **COUNTIF BG COLOR(Range, ColorNumber)**

Counts cells in a given range for which the background color is the same as given ColorNumber.

### **SUMIF FONT COLOR(Range, ColorNumber)**

Sums up cells' values in a given range for which the font color is the same as given ColorNumber.

### **SUMIF BG COLOR(Range, ColorNumber)**

Sums up cells' values in a given range for which the background color is the same as given Color-Number.

### **EXTRACTNUMBER(Cell, [Separator])**

Extracts a number hidden in cell (string) together with its decimal part. The second argument (Sep-arator) is optional - if it is not given, the function takes the default decimal separator to recognize the decimal part. If it is given, the function treats the given character as a decimal separator to recognize the decimal part, but the returned number still has the format set according to the user set-tings.

### **STATICRND()**

Returns a static (such that does not change during sheet's recalculation) random number. The re-turned number is bigger or equal to 0 and less or equal to 1. Warning! The number may change when the workbook is saved and open again.

**STATICRANDBETWEEN(LowerBound, UpperBound)**

This function returns a random integer number, which is bigger or equal to argument LowerBound and less or equal to argument UpperBound. The number does not change when the worksheet is recalculated. Warning! The number may change when the workbook is saved and open again.

**Text functions****CONCATENATEWITHSEP(Range, [Separator], [LeaveEmpty])**

Returns string which concatenating the cells' values from a given range creates and separates the values with an optional Separator. The argument LeaveEmpty is optional - by default set to TRUE.

**REMOVEDIGITS(Cell)**

Removes all digits from a given cell and returns all the other characters (letter, commas etc.).

**WEEKDAYNAME(Cell)**

Returns weekday name for given date.