

Formatting Word fields with switches

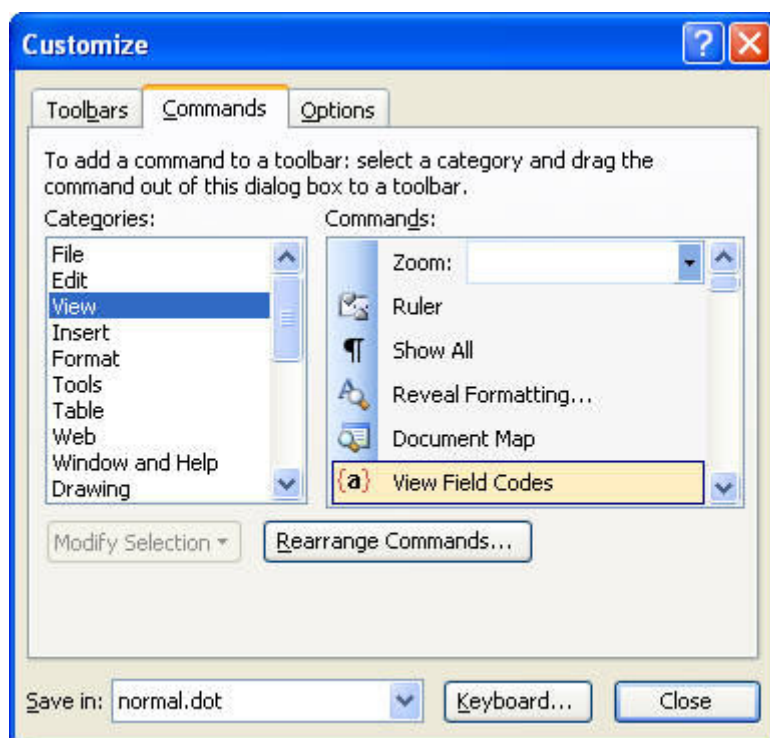
Word provides the means to format the contents of fields inserted into the document, by the use of 'switches' or masks. The full range of these switches can be found [at the end of this page](#).

Another very useful document, created by fellow Word MVP 'Macropod' can be downloaded from <http://www.wopr.com/w3tfiles/10-365442-WordFieldMaths.zip> which explains field calculations in some depth in an easy to follow style - strongly recommended reading.

Note: Where field combinations are inserted manually, the field boundaries { } are inserted with CTRL+F9 and not typed directly from the keyboard.

The following tips are intended to cover some of the less obvious issues.

To edit the field switches, toggle between the field display and the result with ALT+F9 or use **Tools > Customize** to add a **View Field Codes** toolbar button to toggle the display. Select the item from the list and drag it to a toolbar:



The F9 key will update the field content to reflect the changes, or you could use [macro code to update all the fields](#) in the document.

Note: Microsoft changed the way office programs are connected for the purposes of mail merging with Office 2002 and this change was maintained in Office 2003. Word is now expected to provide the formatting. It is still possible to revert to the old way of doing things. To this end, see the Excel Data section of '[Mail Merge to Labels with Office XP](#)'.

Number fields ([For negative numbers in calculations, see the separate entry](#).)

Word's formatting switches can handle numbers up to 14 digits after which it rounds off the numbers. This can cause problems when serial numbers contain more than 14 digits and the last digits are rather important!

Take the example of the number **3456 7123 0987 7652**, assembled as four blocks of four digits. In the data file this may be stored as a single number **3456712309877652**.

Inserting the Mergefield

```
{ MERGEFIELD Serial_No }
```

will produce:

```
3456712309877652
```

Logic, and Word's help on the subject, suggests the addition of a numeric switch thus:

```
{ MERCFIELD Serial_No \# "0000 0000 0000 0000" }
```

will produce the required result, but in fact what happens is that the number is rounded to 14 digits thus:

```
3456 7123 0987 7700
```

The answer should be to split up the field into two parts with the aid of the SET field to assign parts of the data to bookmarks, then to display the bookmarks with the switches.

```
{ SET A "{ MERCFIELD Serial_No }" } { SET B "{ =INT(A / 100000000) }" } { SET C "{ =A - (B * 100000000) }" } { B \# "0000 0000" } { C \# "0000 0000" }
```

The result is:

```
3456 7123 0987 7653
```

unless there is a zero as the penultimate number in the sequence, where thanks to a bug in Microsoft's mathematics, the last number is rounded down to zero also.

```
From the database - 2234567890124506
```

```
From the calculation - 2234 5678 9012 4500
```

It requires a little brute force to overcome this by adding the last digit back into the bookmark C. As that digit is not usually known, this means that you have to add a calculation for each of the 9 possible alternatives, so the resulting construction is never going to look elegant. But it does work.

```
{ SET A "{ Mergefield Serial_No }" } { SET B "{ =INT(A / 100000000) }" } { SET C "{ =A - (B * 100000000) }" } { IF { A } = "*01" "{ SET C { =C+1 } }" } { IF { A } = "*02" "{ SET C { =C+2 } }" } { IF { A } = "*03" "{ SET C { =C+3 } }" } { IF { A } = "*04" "{ SET C { =C+4 } }" } { IF { A } = "*05" "{ SET C { =C+5 } }" } { IF { A } = "*06" "{ SET C { =C+6 } }" } { IF { A } = "*07" "{ SET C { =C+7 } }" } { IF { A } = "*08" "{ SET C { =C+8 } }" } { IF { A } = "*09" "{ SET C { =C+9 } }" } { B \# "0000 0000" } { C \# "0000 0000" }
```

```
Corrected - 2234 5678 9012 4506
```

Note: Thanks to fellow Word MVP Peter Jamieson for suggesting the above solution.

Insert field information and associated text conditionally

Sometimes when merging data you may need to include additional text only when the associated field has content, or you may wish to insert a particular field without leaving a space when there is no content in that field. The simplest method is to use a conditional field or fields to place the data and associated text.

In the following example using data from an Outlook contacts list, I have prepared a simple Directory (Catalog) merge which includes various fields and associated text depending on whether the fields have content. With

the three records used for the illustration, one of the conditionally included fields and text does not appear. In the first of the three illustrations, note the positions of the quote marks "" which set the limits of what is printed.

```
{ MERCEFIELD File_As }
{ MERCEFIELD Business_Address } { IF { MERCEFIELD Business_Phone } <> "" "
Phone: { MERCEFIELD Business_Phone } " } { IF { MERCEFIELD Home_Phone } <> "" "
Home Phone: { MERCEFIELD Home_Phone } " } { IF { MERCEFIELD Other_Address } <> "" "

Other address:
{ MERCEFIELD Other_Address } }

*****
```

For the sake of clarification, for the second illustration I have switched on the display of formatting marks (CTRL+* or click the ¶ button on the toolbar).

```
{ MERCEFIELD File_As } ¶
{ MERCEFIELD Business_Address } { IF { MERCEFIELD Business_Phone } <> "" ¶
Phone: { MERCEFIELD Business_Phone } " } { IF { MERCEFIELD Home_Phone } <> "" ¶
Home Phone: { MERCEFIELD Home_Phone } " } { IF { MERCEFIELD Other_Address } <> "" ¶
¶
Other address: ¶
{ MERCEFIELD Other_Address } } ¶
¶
***** ¶
¶
```

In the third of the pictures is the result of the merge

KLM UK
Customer Relations
P.O. Box 104
Crawley
West Sussex
RH10 6YH

Cyprus Airways
37 Gladstone Str.
Paphos
Cyprus
Phone: +357 (26) 233556

Helios Airways Ltd
22 Nietzsche Street
Ria Court No. 9, 1st Floor
6028 Larnaca
Cyprus
Phone: +357 (24) 815 700

Other address:
Kenville House Unit 3
Spring Villa Business Park
Spring Villa Road
Edgware, Middlesex
London
HA8 7EB

Conditionally Insert a graphical image

Where images are included as part of a conditional (IF) field construction, the IncludePicture field does not work as anticipated. The results of the conditional field are hard coded into the merge, which does not provide the opportunity to update, following a merge to a new document. (See also [Mail Merge Graphics](#)).

Thus conventional wisdom would suggest that:

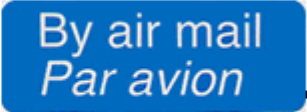
```
{ IF { Mergefield Fieldname } <> "" "{ INCLUDEPICTURE D:\\My Documents\\My Pictures\\{ Mergefield Fieldname }" }" "" }
```

should insert the picture identified from the mergefield 'Fieldname' and nothing when the field is empty. Unfortunately this is not the case.

The following workaround demonstrates the insertion of a blank image where the condition is not met (save e.g. a 1 pixel transparent image as [Blank.jpg](#) to give the effect of nothing being inserted - or download such an image by clicking the link). Note that INCLUDEPICTURE is the primary field and the condition is achieved as part of its content rather than the other way around.

```
{ INCLUDEPICTURE { IF { Mergefield "Fieldname" } <> "" "D:\\My Documents\\My Pictures\\{ Mergefield "Fieldname" }" "D:\\My Documents\\My Pictures\\Blank.jpg" } }
```

Note: You can insert the required picture into the result of the conditional field directly. and that will work also. e.g. to add an airmail image to envelopes intended for other countries, you could use the following:

```
{ IF{ Mergefield Country \*upper} <> "CYP*" "  }
```

which will insert the [airmail image](#) when the country field is other than Cyprus (where I live).

Conditionally insert a Stylerref Field in a header/footer

Stylerref fields are used to display the content of a nominated paragraph or character style - and are most frequently found in page header or footer.

Header

```
{ STYLEREF "Heading 1" }
```

This is Heading 1

Header

This is Heading 1

This is Heading 1

If the nominated style does not exist, then the following error message is displayed:

Header

Error! No text of specified style in document.

This isn't!

This is not very helpful when building a template that may not have this particular style. It is, however possible to conditionally search for the error message in the field result and only insert the Stylerref field when the error message is not present. You can use a wildcard in the search pattern as shown.

Header

```
{ IF{ STYLEREF "Heading 1" } <> "Error*" "{ STYLEREF "Heading 1" }" }
```

This is Heading 1

This produces the required content when the style is present

Header
This is Heading 1
This is Heading 1

..... but nothing when it isn't

Header
This isn't

Insert quotation marks in a conditional merge field result

Peter Jamieson also came up with the following approach to the problem of inserting quotation marks "" in the result of a conditional mail merge.

In a recent newsgroup question the questioner wanted to insert Exhibit "A" when a certain field was empty.

The logical code for this would appear to be

```
{ IF{ Mergefield Test } = "" "Exhibit "A":}
```

but this chops off the text at the first quote mark:

Exhibit

You may then consider 'escaping' the quote mark with a \' eg:

```
{ IF{ Mergefield Test } = "" "Exhibit \A":}
```

but unfortunately Word inserts the quotes but also the slash 😞

Exhibit \A\:

One solution is to set the quote and the slash as a bookmark using a SET field and then insert the bookmarks using REF fields (the REF part is optional) thus:

```
{ SET q "" } { IF{ Mergefield Test } = "" "Exhibit { q }A{ q }:" }
```

An alternative approach is to use a pair of Quote fields to place the quotes characters directly e.g.

```
{ IF{ Mergefield Test } = "A" "Exhibit { Quote 34 }A{ Quote 34 }:" }
```

either of which produces the desired result 😊:

Testing for a value in one of several fields.

Sometimes when preparing a mail merge document, you may wish to insert text based upon whether a value has been inserted in one of a number of fields.

The following example tests for whether Field1 contains "A", or Field2 contains "B", or Field3 contains "C". If any or all of those fields meet the condition, "True" is entered in the merge document. If not, "False" is entered. In a real merge situation True or False can be replaced by whatever you wish.

You could test for any content using this method, which uses a collection of conditional fields that insert a value of 1 or 0 into a calculation according to whether the value required is contained in the field. The results are added together. If the result of that calculation is 1 or more, then one or more of those fields contains the required data. Thus:

```
{ IF{ =( { IF{ MERGEFIELD Field1 } = "A" 1 0 } + { IF{ MERGEFIELD Field2 } = "B" 1 0 } + { IF{ MERGEFIELD Field3 } = "C" 1 0 } ) } >= 1 "True" "False" }
```

The following shows the content of the three fields and the result of the merge calculation.

Field 1 = A	Field 2 = B	Field3 = C	Result = True
Field 1 = B	Field 2 = V	Field3 = J	Result = False
Field 1 = V	Field 2 = B	Field3 = T	Result = True
Field 1 = Q	Field 2 = L	Field3 = C	Result = True
Field 1 = K	Field 2 = H	Field3 = P	Result = False

Convert upper case data to lower case with the first word capitalized.

The *FirstCap switch is normally used to format fields to lower case with the first word capitalized as in a normal sentence. However with some upper case data sources, the switch doesn't appear to do anything. The solution here is to convert the field to lower case first, by using a *Lower switch, thus:

```
{ Mergefield Fieldname \*Lower \*FirstCap }
```

The *Lower switch is also useful when used in conjunction with data - e.g. user forms where the content may be entered in upper or lower case, but you wish to test for the entered content. Let's take the example of a bookmark from a user form which requires YES to be entered in the document when the content of the bookmark is 'yes' or 'YES' or 'Yes'. In this case add the *Lower switch to the REF field and test for y followed by any other characters.

```
{ IF{ REF bookmarkname \*lower } = "y*" "YES" "NO" }
```

Format cash amounts to 2 decimal places

Sometimes when merging data, comprising cash amounts, especially when importing from Excel, the mergefield displays up to 14 decimal places. This is caused by the way Excel handles numeric data internally, but is easily tamed by the addition of a switch to limit the data to 2 decimal places.

e.g.

```
25.00000000
```

\$25.00

are produced by the following 2 fields respectively

```
{ MERGEFIELD Amount }
```

```
{ MERGEFIELD Amount \# "$,0.00;($,0.00)" }
```

In the second field, the switch \# "\$,0.00;(\$,0.00)" includes a comma, which provides for the commas to indicate thousands and millions eg

\$9,826,154,231,417,700.00

and an optional currency symbol - here a dollar sign. The numeric mask can also be expressed as "\$,#.00;(\$,#.00)" where the '#' will suppress the 0 for amounts less than a dollar.

One variation I use a lot is for Cyprus Pound currency amounts:

```
{ MERGEFIELD Amount \# "CY£,0.00;(CY£,0.00)" }
```

The section after the semi colon dictates what to do with negative amounts - here they are bracketed and coloured red. The semi-colon and following section are optional.

Format cash amounts to 0 decimal places

Sometimes with decimal amounts, when the amount to the right of the decimal is zero, you may wish to show a whole number, without the decimal, whilst retaining the amount to the right of the decimal where the amount is greater than zero eg

\$12.00, should display as \$12, but \$12.95 should retain its decimal amount. This cannot be achieved with a simple switch and therefore you need to insert a conditional field to display one or the other according to what comes after the decimal. In the following example I have suppressed the display of zero amounts (see section below).

```
{ IF { =MOD({ Mergefield Amount },1) } = 0 "{ Mergefield Amount \# "$,0;" } "{ Mergefield Amount \# "$,0.00" }
```

Formatting cash amounts in words

Word provides special switches - ***cardtext** and ***dollartext** (detailed in the table at the end of this page) but these are somewhat limiting if you want to express amounts in words. It is, however, possible to derive amounts in words using conditional, and formula fields with the ***cardtext** switch.

The following example takes the decimal output from a form field, using the default form field bookmark name **Text1**. The fields can be easily adapted to use mergefields and any decimal currency.

```
{ IF{ =INT (Text1) } <> 0 "{ =INT (Text1) \*cardtext } dollar" } { IF { =INT(Text1) } > 1 "s" } { IF{ =INT (Text1) } <> 0 " { IF { =( { REF Text1 } - { =INT (Text1) } ) * 100 } <> 0 " and " "" } "" } { IF { =( { REF Text1 } - { =INT (Text1) } ) * 100 } <> 0 "{ =( { REF Text1 } - { =INT (Text1) } ) * 100 \*cardtext } cents" }
```

Where **Text1** contains the amount **1.00** - the fields produce **one dollar**

Where **Text1** contains the whole number **2.00** (or greater) the fields produce **two(or more) dollars**

Where **Text1** contains **.50** the fields produce **fifty cents**

Where **Text1** contains **5.75** the fields produce **five dollars and seventy-five cents**

etc

Percentages

Frequently, percentages will display as decimals - eg 41% from the data may display as 0.41354836739 when merged into Word. To display the decimal as a percentage, you should create a calculated field eg:

```
{ = { MERCEFIELD Amount } * 100 \# "0% " }
```

41%

Suppress field display for numbers <=0

You may wish to suppress the field result altogether, when the numeric content is 0 or a negative number. This can be achieved with a conditional field, or more simply with a variation on the above numeric switch eg \# "\$,0.00;"; Any positive number will display in a dollar format (the dollar sign being optional again). Zero or negative numbers will display nothing.

The two semi colons at the end determine what happens with negative numbers and zeros. To understand the principle consider

```
\# "$,0.00;negative;zero"
```

In other words for a number greater than zero the result is formatted as \$,0.00

For numbers less than zero - the number is formatted as negative - in this case "negative" will be entered

For a result of zero - the number is formatted as zero - and again in this example will be entered as "zero"

Lets say you have a data file with a single field called 'Number', which contains the following records:

-1

0

1

Merging that field - {Mergefield Number \# "\$,0.00;negative;zero"}

will result in:

negative

zero

\$1.00

Calculated Form Fields - suppress zero

The currency switches available in a calculated form field are limited, but you can suppress the zero in a calculated form field, by adding a switch to the calculation part of the field. Toggle the display (ALT+F9) to display:

```
Quantity { FORMTEXT } Price { FORMTEXT } Sub Total { FORMTEXT {  
=Quantity * Price } }
```

Then add the switch.

```
Quantity { FORMTEXT } Price { FORMTEXT } Sub Total { FORMTEXT {  
=Quantity * Price \# "£,0.00;(£,0.00);" } }
```

Alternatively abandon the calculated form field and use instead a Word formula field thus:

```
Quantity { FORMTEXT } Price { FORMTEXT } Sub Total { = { Quantity }  
* { Price } \# "£,0.00;(£,0.00);" }
```

Then toggle back or lock the form to display the results, which would be the same in either case.

```
Quantity 2 Price £30.00 Sub Total £60.00
```

```
Quantity -3 Price £30.00 Sub Total (£90.00)
```

```
Quantity 0 Price £30.00 Sub Total
```

US Zip Codes

Five digit zip codes are reported as dropping the leading zero when merging from Excel/Access data files. A simple numeric switch should fix that one

```
{ MERCEFIELD Zip \# "00000" }
```

Formatting problem: the zip codes in my data source are 5-digit or 9-digit. What I'd like to do is apply a number format that will convert all ZIP codes to ZIP + four format, i.e. 12345 becomes 12345-0000, 123450000 becomes 12345-0000 and 123456789 becomes 12345-6789

This requires the use of a conditional field which tests whether the field has five or nine digits then applies appropriate formats.

```
{ IF { MERCEFIELD Zip } > "99999" "{ MERCEFIELD Zip \# "00000'-0000'" } "{  
MERCEFIELD Zip \# "00000"-0000" }
```

```
Zip = 12345
```

```
12345-0000
```

```
Zip = 123456789
```

```
12345-6789
```

If you wish to display the 5 digit zips without the '-0000' suffix used in the above example, then leave out that part of the formatting switch thus:

```
{ IF{ Mergefield Zip } > 99999 "{ Mergefield Zip \# "00000'-0000" }" "{ Mergefield Zip \# "00000" }" }
```

Note: Calculations cannot be performed on text, so in order for the above to be of use, the data should be entered as a continuous number of 5 or 9 digits. 9 digit numbers can be formatted using Excel's cell formatting as a 5+4 Zip, which will provide the hyphen in the Excel display whilst retaining the number as the raw data.

Testing for events that occur before or after a certain date

When running a mail merge, you may want to test for events that happen before or after a certain date. Mail merge does not convert dates to numbers, so if you cannot automatically derive a date number from the date in the data file, as you could in (say) Excel, a different plan is called for.

Let's say the date comes into Word in the format d/MM/yyyy or 1/10/2002 (1st October 2002) from a MERGEFIELD called Start_Date. In this example, we are looking to identify records with Start_Date entries before 1st October 2002.

```
{ IF{ MERCEFIELD Start_Date } < 1/10/2002 "True" "False" }
```

The above would appear the logical check, but the check treats the date as a number and identifies that number as 1, which is the first part of the number before the slash '/'. All dates other than the first of the month will be greater than 1, so all will produce the result "False".

We therefore need to display the date in numbers that represent the date in a unique way by using the date mask **yyyyMMdd** which displays the date as a series of digits for year month and finally day, without any breaks. This is a number that the conditional field will view as a whole.

```
{ IF{ MERCEFIELD Start_Date \@ "yyyyMMdd" } < 20021001 "True" "False" }
```

Thus any date before 1/10/2002 would produce "true" and any other date would produce "false"

Testing for fields that may contain numbers/date or text

While it would be better to ensure that the data file has separate fields for numbers and text, sometimes you have to work with what you get. In a recent newsgroup question a user wanted to test a field that contained dates or text and reproduce only those fields that contained dates. The solution relies on the fact that if you perform a calculation on a field containing text, an error message is produced e.g.:

!Syntax Error, TEXT

This can be trapped in a conditional field structure by comparing the result of the field with the error message thus:

```
{ MERCEFIELD Result } { IF{ =( { MERCEFIELD Result } *1 ) } <> "!" "The field contains a date" "The field contains text" }
```

produces

```
13/03/2006            The field contains a date
```

if the field contains a date,

or

Text The field contains text

If the field contains text.

There is no need to test against the complete error message, as the condition will accept wildcards. Thus you can search for the leading exclamation mark and the wildcard character "*" as shown.

Date fields with ordinals

UK dates are often displayed using a superscripted ordinal such as **23rd November 2002**. This is simple enough to produce if the date is typed from the keyboard, but is difficult to produce automatically with fields (though there is a possible solution suggested in newsgroup contributor Macropod's impressive study of date fields at <http://www.wopr.com/cgi-bin/w3t/showflat.pl?Cat=&Board=wr&Number=249902>.) as shown below:-

```
{QUOTE{DATE \@ "dddd 'the' d"}-IF{=(mod({DATE \@ d},10)<4)*(mod({DATE \@ d},10)<>0)*({DATE \@ d}<>11)*({DATE \@ d}<>12)*({DATE \@ d}<>13)}=1 {=mod({DATE \@ d},10)-2 \# rd:nd} th}{ DATE \@ "' of' MMMM, yyyy" }}
```

which produces:-

Monday the 11th of April, 2005

Without such complexity, Word can readily manage 23rd November 2002 i.e. without the superscript, as the superscript is usually applied by Word's autofmt function, which does not affect insertions by field.

The required format can be achieved by converting the field to text and then applying the auto-formatting to the result.

For a document template start with a combination of fields:

```
{ CREATEDATE \@ "d" \* Ordinal } { CREATEDATE \@ "MMMM yyyy" }
```

Select both fields and save to a bookmark - call it **Date**

The following macro will run automatically when a new document is opened, which will convert the fields to text and autofmt to add the superscript.

```
Sub AutoNew()  
'  
' FormatDate Macro  
Selection.GoTo What:=wdGoToBookmark, Name:="Date"  
Selection.Find.ClearFormatting  
With Options  
AutoFormatReplaceOrdinals = True  
End With  
Selection.Range.AutoFormat  
Selection.EscapeKey  
Selection.Fields.Unlink  
Selection.EndKey Unit:=wdLine  
End Sub
```

The macro MUST be saved in the relevant document template and not in the default template - normal.dot

While the above is ideal where the date field is pre-inserted into a document template, where you simply wish to insert a similarly formatted current date into an existing document then with a small variation the same principle can be applied. Here the date fields are saved to an autotext entry - **this time save in normal.dot** to enable such dates to be available to all documents. Call the autotext entry **Date**. The following macro inserts the autotext and then converts to text and formats the result.

```
Sub InsertFormattedDate()
NormalTemplate.AutoTextEntries("Date").Insert Where:=Selection.Range
Selection.HomeKey Unit:=wdLine, Extend:=wdExtend
With Options
AutoFormatReplaceOrdinals = True
End With
Selection.Range.Fields.Update
Selection.Range.AutoFormat
Selection.Fields.Unlink
Selection.MoveRight Unit:=wdCharacter, Count:=1
End Sub
```

Changing minutes to hours and minutes

In a database used for mail merge, you may have a field called (e.g) MINUTES that contains a time in minutes e.g. 165 minutes, that you want to display in hours and minutes. Normal date/time switches will not help here, so you need to resort to mathematics to produce the required result

```
{ =INT({ MERGEFIELD MINUTES \# "0" }/60) \# "0'h"} { = 60 * (({ MERGEFIELD MINUTES \# "0" }/60) - { =INT({ MERGEFIELD MINUTES \# "0" }/60) }) \# "0'mins" }
```

alternatively

```
{ =INT({ MERGEFIELD MINUTES \# "0" }/60) \# "0'h" } { =MOD({ MERGEFIELD MINUTES \# "0" },60) \# "0'mins" }
```

either of which will produce

2h 45mins

In both examples the use of the formatting switches indicated below will remove any text following the number, thus if the field MINUTES actually contains 165 minutes, the content of the field is treated as 165.

```
{ =INT({ MERGEFIELD MINUTES \# "0" }/60) \# "0'h" } { =MOD({ MERGEFIELD MINUTES \# "0" },60) \# "0'mins" }
```

The basic switch information - reproduced and edited from Word Help

The following information extracted and edited from Word's own help files completes the picture relating to the use of formatting switches.

Format (*) field switch

The following is a list of switches and items that they capitalize:

<p>* Caps</p>	<p>The first letter of each word.</p> <p>For example, { FILLIN "Type your name:" * Caps } displays "Graham Mayor" even if the name is typed in lowercase letters - graham mayor. However see the previous section which demonstrates how to overcome the anomaly that occurs if you enter the name in upper case.</p>
-----------------------	--

* FirstCap	<p>The first letter of the first word.</p> <p>For example, { COMMENTS * FirstCap } might display "Weekly report on sales".</p> <p>Note: See also the previous section concerning the anomaly where the underlying text is formatted as all caps.</p>
* Upper	<p>All letters are upper case</p> <p>For example, { QUOTE "word" * Upper } displays "WORD".</p>
* Lower	<p>All letters are lowercase.</p> <p>For example, { FILENAME * Lower } displays "weekly sales report.doc".</p> <p>Note: This switch has no effect if the entire field that contains the switch is formatted as small capital letters.</p>

Number formats

The following is a list of number switches and their results:

*alphabetic	<p>Displays results as alphabetic characters. The result has the same case as the word "alphabetic" in the switch.</p> <p>For example, { SEQ appendix * ALPHABETIC } displays "B" (instead of "2"), and { SEQ appendix * alphabetic } displays "b".</p>
*Arabic	<p>Displays results as Arabic cardinal numerals.</p> <p>For example, { PAGE * Arabic } displays "31".</p> <p>Note: If the Number format setting in the Page Number Format dialog box (Page Numbers command, Insert menu) is not Arabic, this switch overrides the Number format setting.</p>
*CardText	<p>Displays results as cardinal text. The result is formatted in lowercase letters unless you add a format switch to specify a different capitalization.</p> <p>For example, { = SUM(A1:B2) * CardText } displays "seven hundred ninety", and { = SUM(A1:B2) * CardText * Caps } displays "Seven Hundred Ninety".</p>
*DollarText	<p>Displays results as cardinal text. Microsoft Word inserts "and" at the decimal place and displays the first two decimals (rounded) as Arabic numerators over 100. The result is formatted in lowercase letters unless you add a format switch to specify a different capitalization.</p> <p>For example, { = 9.20 + 5.35 * DollarText * Upper } displays "FOURTEEN AND 55/100".</p>
*Hex	<p>Displays results as hexadecimal numbers. For example, { QUOTE "458" * Hex } displays "1CA".</p>
*OrdText	<p>Displays results as ordinal text. The result is formatted in lowercase letters unless you add a format switch to specify a different capitalization.</p> <p>For example, { DATE \@ "d" * OrdText } displays "twenty-first", and { DATE \@ "d" * OrdText * FirstCap } displays "Twenty-first".</p>
*Ordinal	<p>Displays results as ordinal Arabic numerals. For example, { DATE \@ "d" * Ordinal } displays "30th".</p> <p>Note: See the section above relating to the insertion of superscripted ordinals</p>
*roman	<p>Displays results as Roman numerals. The result has the same case as the word "roman" in the field code. For example, { SEQ CHAPTER * roman } displays "xi", and { SEQ CHAPTER * ROMAN } displays "XI".</p>

Character formats and protecting previously applied formats

The following are character formatting switches and their results:

<p>*Charformat</p>	<p>Applies the formatting of the first letter of the field to the entire result. The result of the following example has bold formatting because the "R" in "REF" is bold. <code>{ REF chapter2_title * Charformat }</code> displays "Whales of the Pacific".</p> <p>You can of course use this to ensure that the formatting retains the formatting of the line that the field is inserted into. Thus <code>{ REF chapter2_title * Charformat }</code> would display <i>"Whales of the Pacific"</i></p>
<p>*MERGEFORMAT</p>	<p>Applies the formatting of the previous result to the new result. For example, if you select the name displayed by the field <code>{ AUTHOR * MERGEFORMAT }</code> and apply bold formatting, Microsoft Word retains the bold formatting when the field is updated when the author name changes.</p> <p>Note: For some odd reason known only to Microsoft, when you insert fields by using the Field dialog box (Insert menu, Field command), the *MERGEFORMAT switch is included by default. You can turn this option off on an individual basis by clearing the Preserve formatting during updates check box in the Field dialog box.</p>

Numeric Picture (\#) field switch

Specifies the display of a numeric result.

For example, the switch `\# $,0.00` in `{ = SUM(ABOVE) \# $,0.00 }` displays a result such as "\$4,455.70." If the result of a field is not a number, this switch has no effect.

Note: Quotation marks are not required around simple numeric pictures that do not include spaces—for example, `{ MarchSales \# $,0.00 }`. For more complex numeric pictures and those that include text or spaces, enclose the numeric picture in quotation marks, as shown in the following picture item examples. Microsoft Word adds quotation marks to numeric picture switches if you insert a field by using the Field command (Insert menu) or the Formula command (Table menu).

Combine the following picture items to build a numeric picture switch.

<p>0 (zero)</p>	<p>Specifies the requisite numeric places to display in the result. If the result does not include a digit in that place, Word displays a 0 (zero). For example, <code>{ = 4 + 5 \# 00.00 }</code> displays "09.00".</p>
<p>#</p>	<p>Specifies the requisite numeric places to display in the result. If the result does not include a digit in that place, Word displays a space. For example, <code>{ = 9 + 6 \# \$### }</code> displays "\$ 15".</p>
<p>x</p>	<p>Drops digits to the left of the "x" placeholder. If the placeholder is to the right of the decimal point, Word rounds the result to that place. For example: <code>{ = 111053 + 111439 \# x## }</code> displays "492". <code>{ = 1/8 \# 0.00x }</code> displays "0.125". <code>{ = 3/4 \# .x }</code> displays ".8".</p>
<p>. (decimal point)</p>	<p>Determines the decimal point position. For example, <code>{ = SUM(ABOVE) \# \$,0.00 }</code> displays "\$495.47".</p> <p>Note: Use the decimal symbol specified as part of the regional settings in Microsoft Windows Control Panel.</p>
<p>, (digit grouping symbol)</p>	<p>Separates a series of three digits. For example, <code>{ = NetProfit \# \$#,###,### }</code> displays "\$2,456,800".</p> <p>This can be written more simply as <code>{ = NetProfit \# \$,0 }</code></p> <p>Note: Use the digit grouping symbol specified as part of the regional settings in Windows Control Panel.</p>

- (minus sign)	Adds a minus sign to a negative result, or adds a space if the result is positive or 0 (zero). For example, { = 10 - 90 \# -0 } displays "-80" and { = 10 + 90 \# -0 } displays " 190".
+ (plus sign)	Adds a plus sign to a positive result, a minus sign to a negative result, or a space if the result is 0 (zero). For example, { = 100 - 90 \# +0 } displays "+10" and { = 90 - 100 \# +0 } displays "-10".
%, \$, *, and so on	Includes the specified character in the result. For example, { = netprofit \# "##%" } displays "33%".
"positive; negative"	Specifies different number formats for positive and negative results. For example, if the bookmark Sales95 is a positive value, the field { Sales95 \# "\$,0.00;-\$, 0.00" } displays the value with regular formatting—for example, "\$1,245.65". A negative value is displayed with bold formatting and a minus sign—for example, "-\$ 345.56 ".
"positive; negative; zero"	Specifies different number formats for a positive result, a negative result, and a 0 (zero) result. For example, depending on the value of the Sales95 bookmark, { Sales95 \# "\$,0.00;(\$, 0.00);\$0" } displays positive, negative, and 0 (zero) values as follows: \$1,245.65, (\$ 345.56), \$0 See also the previous example
'text'	Adds text to the result. Enclose the text in single quotation marks. For example, { = { Price } *8.1% \# "\$,0.00 'is sales tax' " } displays "\$347.44 is sales tax".
`numbereditem`	Displays the number of the preceding item that you numbered by using the Caption command (Insert menu, Reference submenu) or by inserting a SEQ field. Enclose the item identifier, such as "table" or "figure," in grave accents (`). The sequential number is displayed in Arabic numerals. For example, { = SUM(A1:D4) \# ",0.00 'is the total of Table' `table`" } displays "456.34 is the total of Table 2".

Date-Time Picture (\@) field switch

Specifies the display of a date or time.

For example, the switch \@ "dddd, MMMM d, yyyy" in the field { DATE \@ "dddd, MMMM d, yyyy" } displays "Friday, November 24, 2002." Combine the following date and time instructions— day (d), month (M), and year (y); hours (h) and minutes (m)— to build a date-time picture. You can also include text, punctuation, and spaces.

Date instructions	
Month (M)	The letter "M" must be uppercase to distinguish months from minutes.
	M Displays the month as a number without a leading 0 (zero) for single-digit months. For example, February is "2".
	MM Displays the month as a number with a leading 0 (zero) for single-digit months. For example, February is "02".
	MMM Displays the month as a three-letter abbreviation. For example, February is "Feb".
	MMMM Displays the month as its full name. e.g. February
Day (d)	Displays the day of the month or the day of the week. The letter "d" can be either uppercase or lowercase.
	d Displays the day of the week or month as a number without a leading 0 (zero) for single-digit days. For example, the sixth day of the month is displayed as "6".
	dd Displays the day of the week or month as a number with a

		leading 0 (zero) for single-digit days. For example, the sixth day of the month is displayed as "06".
	ddd	Displays the day of the week or month as a three-letter abbreviation. For example, Tuesday is displayed as "Tue".
	dddd	Displays the day of the week as its full name.
Year (y)		Displays the year as two or four digits. The letter "y" can be either uppercase or lowercase.
	yy	Displays the year as two digits with a leading 0 (zero) for years 01 through 09. For example, 1999 is displayed as "99", and 2006 is displayed as "06".
	yyyy	Displays the year as four digits.
Time instructions		
Hours (h)		A lowercase "h" bases time on the 12-hour clock. An uppercase "H" bases time on the 24-hour, or military, clock; for example, 5 P.M. is displayed as "17".
	h or H	Displays the hour without a leading 0 (zero) for single-digit hours. For example, the hour of 9 A.M. is displayed as "9".
	hh or HH	Displays the hour with a leading 0 (zero) for single-digit hours. For example, the hour of 9 A.M. is displayed as "09".
Minutes (m)		The letter "m" must be lowercase to distinguish minutes from months.
	m	Displays minutes without a leading 0 (zero) for single-digit minutes. For example, { TIME \@ "m" } displays "2".
	mm	Displays minutes with a leading 0 (zero) for single-digit minutes. For example, { TIME \@ "mm" } displays "02".
A.M. and P.M. (AM/PM)		Displays A.M. and P.M. To change the A.M. and P.M. symbols for Microsoft Windows, change the regional settings in Windows Control Panel.
	am/pm or AM/PM	Displays A.M. and P.M. as uppercase. For example, { TIME \@ "h AM/PM" } and { TIME \@ "h am/pm" } display "9 AM" or "5 PM". Note: To display lower case, add a *Lower switch i.e. { TIME \@ "h am/pm" *Lower } displays "9 am" or "5 pm"
Other text and punctuation		
	'text'	Any specified text in a date or time. Enclose the text in single quotation marks. For example, { TIME \@ "HH:mm 'Greenwich mean time' " } displays "12:45 Greenwich mean time".
	character	Includes the specified character in a date or time, such as a : (colon), - (hyphen), * (asterisk), or space. For example, { DATE \@ "HH:mm MMM-d, 'yy" } displays "11:15 Nov-6, '99".
	`numbereditem`	Includes in a date or time the number of the preceding item that you numbered by using the Caption command on the Insert menu (Reference submenu), or by inserting a SEQ field. Enclose the item identifier, such as "table" or "figure," in grave accents (`). Microsoft Word displays the sequential number in arabic numerals. For example, { PRINTDATE \@ "'Table' `table` 'was printed on' M/d/yy" } displays "Table 2 was printed on 9/25/02". Note: Quotation marks are not required around simple date-time pictures that do not include spaces or text - for example, { DATE \@ MM/yy }. For more complex date-time pictures and those that

		<p>include spaces or text, enclose the entire date-time picture in quotation marks, for example, { DATE \@ "dddd MMMM d, yyyy", at' h:mm" }.</p> <p>Microsoft Word adds quotation marks to date-time picture switches if you insert a field by using the Date and Time command or the Field command (Insert menu).</p>
	<p>Lock Result (!) field switch</p>	<p>Prevents a field that is included in the result of a BOOKMARK, INCLUDETEXT, or REF field from being updated unless the field result in the original location has changed.</p> <p>Without this switch, Microsoft Word updates fields included in a field result whenever the BOOKMARK, INCLUDETEXT, or REF field is updated.</p> <p>For example, the field { INCLUDETEXT C:\\Sales\\Qtr4 Sales.doc ! } inserts the contents of the document "Qtr4 Sales.doc," which contains a DATE field and an EMBED field.</p> <p>If you update the INCLUDETEXT field, the "!" switch prevents Word from updating the DATE and EMBED fields in the included text unless they are first updated in the original document ("Qtr4 Sales.doc").</p> <p>The switch ensures that the text inserted by the INCLUDETEXT field matches the text in the original document. To update the DATE and EMBED fields in both locations, update fields in the original document ("Qtr4 Sales.doc"), and then update the INCLUDETEXT field.</p>