!, # , ###, ƒ, and More Tagging On-a-Sheet

Tagging names of files and folders, inside documents, for faster and targeted searches

Exclamation mark (!) as prefix to tag more important stuff, and '###' to tag unfinished "business". Make finding information quicker and more targeted.

Table 1: Tags

Tag	Where	Use	Description of use
'!'	Suffix	Yes	Using as suffix (from 2018-05-09, see '#' and 'ƒ' below for history) to highlight certain folder(s), or files, as more important, like a top folder for all sorts of things related to a specific topic, category. Examples: `Office! COMGT` - top folder for office related things for COMGT `Office! J&P` - top folder for office related things for J&P `Office! SI` - top folder for office related things for SI `Office! Suites (SW)` - top folder for office suites (MS, Libre, Open, …) J&P: Try the following: *Analytics!*, *Boat! Boats! Boating!*, *Cars!*, *Clothes! Kläder!*, *Edu!*, *Food! Mat!*, *Fun!*, *Health! & Fitness!*, *Homes! J&Ps*, *MLabs!*, *Privacy! DoNotCall! Security! Safety!*, *Project Management! PjM! PM!*, *Projects! incl Ideas*, *Travel!*, *Voting! US*. More: *7! 7+/- 2, 42!, ATT!,AT&T, Audible!, Austin Library! Hoopla!, Clipart! Graphics! Icons! Symbols!, Community Impact! Newspaper, Dallas Symphony Orchestra, DSO!, DN! Dagens Nyheter, Netflix!, OAS™-On-a-Sheet, OAS!, Pandora! Internet radio, Skiing! Ski!, Spotify!, TicketMaster!, Weather! in general, Sprinklers! Sprinkler Systems! Irrigation! Controllers*.
'###'	Any	Yes	Using for tagging a piece of text (e.g., paragraph) as 'not finished', 'still working on'. Using three '###' make it very easy to search for such areas. Using both in names of files/folders and inside documents.
' - Start Here' ' SH!'	Suffix	Yes	Especially in content management system (CMS) & wiki-type – based solutions (like jandp.biz, e.g. OAS18001 - !,#,###,ƒ and More Tagging, by J&P - Start Here, and Search! - Start Here) The " SH!" is a short form for use in file systems – like local, Dropbox, …, and primarily on folders.
'TODO'	Any	Yes	This alone is a little bit too generic – for search in larger systems – and is commonly used together with some 'qualifier(s)'. E.g. '###TODOJOHSAR' (combined w/ 3 hashes and one form of unique user ID)
X,_XX_, XXX_,… _XREF_	Prefix, +Any	Yes	To indicate a section that essentially REFERS to other section(s). E.g. 'XX_<section header>', '<section header> _XREF_'. Also been used as in 'xRef', 'xReference', similar. Similarly to zRes, zRefs (more in next). (Both 'x'… and 'z'… puts items towards end of list of files & folders when sort on name.)
z,ZZ_, ZZZ_,…	Prefix	Yes	To indicate archived materials but also reference materials. Commonly using zRes or zRefs on folders with collected references when writing a report, et c. Also using with '_', like in 'ZZ_<old header/doc>' or 'ZZZ_<old header/doc>' to indicate outdated, (possibly) 'sleeping', materials.
'ƒ', ('Î')	(Suffix)	**No**	Stopped using early May 2018, don't remember when started using 'ƒ' but it was some 5-10 years prior. 'ƒ' is Option+F on Macintosh, don't know on Win, don't know on iOS – and with increasing use on iPhone (searching for material) why finally decided to change use of 'ƒ', first to '#' for about a week, then to '!'. Now can easily search for (e.g.) 'food!' also on iOS. (Another issue with 'f' is that it's 'Î' when CAPSLOCK is active.
'#'	(Suffix)	**No**	Used/tried as suffix for about a week in early May 2018, but when tests on iOS failed to correctly find items (…!!), changed to '!' on May 9th. Changing this takes lots of time, due to Dropbox, and resync of hundreds of thousands, actually a million files and folders, takes many days. Rec. doing in phases.
~,$,^,…	(Suffix)	**No**	Why once upon a time selected use of 'ƒ'? Because avoiding any interference of any characters commonly used in regular expressions (note below) and such.

Table 2: Special Characters – EXAMPLES of More Common Use

Char	As Prefix	As Separator	As Suffix
#	*Channels* in Slack (e.g. #sales)	Can't come up with any example.	Can't come up with any example.
@	*Individuals* in Slack, Twitter (@johan)	Email addresses (my@test.com)	Can't come up with any example.
$, *, ?, !, …	Lots of 'special characters (not the regular alphanumeric a..z or numeric 0..9) are used for regular expressions (REs) – more on next page.		

Regular Expressions ('RE')

"A **regular expression**, **regex** or **regexp**[1] (sometimes called a **rational expression**)[2][3] is, in theoretical computer science and formal language theory, a sequence of characters that define a *search pattern*. Usually this pattern is then used by string searching algorithms for "find" or "find and replace" operations on strings, or for input validation." [wikipedia: Regular expression]. With history all the way back into the 1950s…, then heavy use in editors like vi and emacs (1970s) to now a few more commonly used families:

- POSIX – with Basic Regular Expressions (BRE), Extended Regular Expressions (ERE), and the depreciated Simple Regular Expressions (SRE). https://en.wikipedia.org/wiki/Regular_expression#POSIX (1992…)
- Perl & TCL – https://en.wikipedia.org/wiki/Perl_Compatible_Regular_Expressions (1997…)

Character classes (→)

"The character class is the most basic regex concept after a literal match." [wikipedia: Regular expression]

Description	POSIX	Perl/Tcl	Vim	Java	ASCII
ASCII characters				\p{ASCII}	[\x00-\x7F]
Alphanumeric characters	[:alnum:]			\p{Alnum}	[A-Za-z0-9]
Alphanumeric characters plus "_"		\w	\w	\w	[A-Za-z0-9_]
Non-word characters		\W	\W	\W	[^A-Za-z0-9_]
Alphabetic characters	[:alpha:]		\a	\p{Alpha}	[A-Za-z]
Space and tab	[:blank:]		\s	\p{Blank}	[\t]
Word boundaries		\b	\< \>	\b	(?<=\W)(?=\w)|(?<=\w)(?=\W)
Non-word boundaries				\B	(?<=\W)(?=\W)|(?<=\w)(?=\w)
Control characters	[:cntrl:]			\p{Cntrl}	[\x00-\x1F\x7F]
Digits	[:digit:]	\d	\d	\p{Digit} or \d	[0-9]
Non-digits		\D	\D	\D	[^0-9]
Visible characters	[:graph:]			\p{Graph}	[\x21-\x7E]
Lowercase letters	[:lower:]		\l	\p{Lower}	[a-z]
Visible characters and the space character	[:print:]		\p	\p{Print}	[\x20-\x7E]
Punctuation characters	[:punct:]			\p{Punct}	[][!"#$%&'()*+,./:;<=>?@\^_`{|}~-]
Whitespace characters	[:space:]	\s	_s	\p{Space} or \s	[\t\r\n\v\f]
Non-whitespace characters		\S	\S	\S	[^ \t\r\n\v\f]
Uppercase letters	[:upper:]		\u	\p{Upper}	[A-Z]
Hexadecimal digits	[:xdigit:]		\x	\p{XDigit}	[A-Fa-f0-9]

ASCII Table (→)

Quite useful when describing sets of characters

https://en.wikipedia.org/wiki/ASCII

Dec	Hex	Name	Char	Ctrl-char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
0	0	Null	NUL	CTRL-@	32	20	Space	64	40	@	96	60	`
1	1	Start of heading	SOH	CTRL-A	33	21	!	65	41	A	97	61	a
2	2	Start of text	STX	CTRL-B	34	22	"	66	42	B	98	62	b
3	3	End of text	ETX	CTRL-C	35	23	#	67	43	C	99	63	c
4	4	End of xmit	EOT	CTRL-D	36	24	$	68	44	D	100	64	d
5	5	Enquiry	ENQ	CTRL-E	37	25	%	69	45	E	101	65	e
6	6	Acknowledge	ACK	CTRL-F	38	26	&	70	46	F	102	66	f
7	7	Bell	BEL	CTRL-G	39	27	'	71	47	G	103	67	g
8	8	Backspace	BS	CTRL-H	40	28	(	72	48	H	104	68	h
9	9	Horizontal tab	HT	CTRL-I	41	29	)	73	49	I	105	69	i
10	0A	Line feed	LF	CTRL-J	42	2A	*	74	4A	J	106	6A	j
11	0B	Vertical tab	VT	CTRL-K	43	2B	+	75	4B	K	107	6B	k
12	0C	Form feed	FF	CTRL-L	44	2C	,	76	4C	L	108	6C	l
13	0D	Carriage feed	CR	CTRL-M	45	2D	-	77	4D	M	109	6D	m
14	0E	Shift out	SO	CTRL-N	46	2E	.	78	4E	N	110	6E	n
15	0F	Shift in	SI	CTRL-O	47	2F	/	79	4F	O	111	6F	o
16	10	Data line escape	DLE	CTRL-P	48	30	0	80	50	P	112	70	p
17	11	Device control 1	DC1	CTRL-Q	49	31	1	81	51	Q	113	71	q
18	12	Device control 2	DC2	CTRL-R	50	32	2	82	52	R	114	72	r
19	13	Device control 3	DC3	CTRL-S	51	33	3	83	53	S	115	73	s
20	14	Device control 4	DC4	CTRL-T	52	34	4	84	54	T	116	74	t
21	15	Neg acknowledge	NAK	CTRL-U	53	35	5	85	55	U	117	75	u
22	16	Synchronous idle	SYN	CTRL-V	54	36	6	86	56	V	118	76	v
23	17	End of xmit block	ETB	CTRL-W	55	37	7	87	57	W	119	77	w
24	18	Cancel	CAN	CTRL-X	56	38	8	88	58	X	120	78	x
25	19	End of medium	EM	CTRL-Y	57	39	9	89	59	Y	121	79	y
26	1A	Substitute	SUB	CTRL-Z	58	3A	:	90	5A	Z	122	7A	z
27	1B	Escape	ESC	CTRL-[	59	3B	;	91	5B	[	123	7B	{
28	1C	File separator	FS	CTRL-\	60	3C	<	92	5C	\	124	7C	|
29	1D	Group separator	GS	CTRL-]	61	3D	=	93	5D	]	125	7D	}
30	1E	Record separator	RS	CTRL-^	62	3E	>	94	5E	^	126	7E	~
31	1F	Unit separator	US	CTRL-_	63	3F	?	95	5F	_	127	7F	DEL