

# Facebook Ad-Creator text macros

## Text Flags

Text Tags works by "find and replace" a static tag pattern in a line of text. with a dynamic value founds in myThings 'recommendation server'. flag implement's its logic on all of the line text.

### Visual Flags

Flag	Tag	Operation
!	#!_varName_#	<b>Must</b> find and replace
+	#+_varName_#	<b>Optional</b> find and replace

Example...	Value Found	Value Not Found
You Pay only #!_price_# per item.	You Pay only <i>100\$</i> per item.	
You Pay only #+_price_# per item.	You Pay only <i>100\$</i> per item.	You Pay only per item.

### Hidden Flags (Dependency)

Flag	Tag	Operation
X	#X_varName_#	<b>Must Find - No</b> replace (hidden)
N	#N_varName_#	<b>Must Not Find - No</b> replace (hidden)

Example...	Value Found	Value Not Found
#X_price_# Now on Sale Best Price Ever...	Now on Sale Best Price Ever...	
#N_price_# Call us for more details		Call us for more details

## Text Functions

Function find specific product data and run a custom calculation on it. the return data is per one product and always **Optional**. function's **don't!** have a "find and replace" logic. they uses the regular dependency tags in order to force a full line text manipulation.

### The Function Flag

Flag	Tag	Description	Operation
F	#F_funcName_#	Run a function	<b>Optional</b> find and replace
	#F_funcName(parameters, ...)_#	Run function with Parameters ...	

Example - function	Value Found	Value Not Found
Save Now #F_discount_#% in each buy	Save Now <i>30%</i> in each buy	Save Now % in each buy

Example - function with dependency flags	Value Found	Value Not Found
#X_oldprice_# Save Now #F_discount_#% in each buy	Save Now <i>30%</i> in each buy	

### The Important tag

Marks a Reco value or simple text as important and the rest of the expression as allowed for trimming, according to Facebook's ad text lengths specifications.

Tag	Parameters	Operation	Returns
#!_expression_#	none	Marks <i>expression</i> as not allowed for trimming.	If expression is a Reco parameter, replaces it with it's value. If not, #!_ and _# are omitted.

Example	Result (ad title – 25 chars max)
#!_pricetext_# only for a lifetime experience	€388.95 only for a...
Crazy sale - only #!_pricetext_# just today!	Crazy sale... €388.95...
#!_productname_# #!_pricetext_#	Google Nexus ...€388.95  (when given Reco product name for a product is <i>Google Nexus 10 (Black) WiFi 16GB 10-inch Android 4.2 Tablet</i> )

## Facebook ad text conditioning

In order to support some conditioning in Facebook ad text manipulation, similar to G2 capabilities, the body and title text has alternative fields which are named alt\_body and alt\_title. For example, If the expression in body field returns nothing, the alternative (alt\_body) is then evaluated. This allows some conditioning and dynamic manipulation based on data from the reco.

Config key	Config value
body	#X_oldpricetext_# #!_productname_# ON SALE! old price: #!_oldpricetext_#
alt_body	#N_oldpricetext_# #!_productname_# GREAT PRICE, just #!_pricetext_#

Examples based on the config values above:

Reco example	Body Result
"title": "Google Nexus 4 (Black) 16GB Quad-Core Android 4.2 Sim Free / Unlocked", "pricetext": "€359.95", "oldpricetext": "€469.95", ...	Google Nexus 4 (Black) 16GB Quad-Core Android 4.2 Sim Free / Unlocked ON SALE! old price: €469.95
"title": "Google Nexus 4 (Black) 16GB Quad-Core Android 4.2 Sim Free / Unlocked", "pricetext": "€359.95", "oldpricetext": "",	Google Nexus 4 (Black) 16GB Quad-Core Android 4.2 Sim Free / Unlocked GREAT PRICE, just €359.95

Typical Config structure:

```
config={  
  
  "title": "#!_productname_# #I_pricetext_#",  
  
  "alt_title": "#!_domain_#,  
  
  "body": " #X_oldpricetext_# #!_productname_# ON SALE! old price: #!_oldpricetext_#",  
  
  "alt_body": "#N_oldpricetext_# #!_productname_# GREAT PRICE, just #I_pricetext_#,  
  
};
```