# Avolites Wiki Writing Guidelines

When writing or editing this wiki please follow these rules - it is good practice to follow some coding and formatting standards (if you want to have a look at a much more sophisticated example: https://pear.php.net/manual/en/coding-standards.php), and this helps the audience to read and follow:

1. The main intention of this wiki is to provide **as much benefit and information to the reader as possible**. If in doubt re-read your writing with the eyes of another person. Do you still understand it?
2. **Use links wherever possible**. Examples should link to the used functions and properties etc. This way it is easy to retrieve more information related to a particular question.
3. Also **use tags** in your examples. There is a tags section. Just throw in some words which describe the example best. This way it can be found more easily.
4. **Indent**. In particular your xml files should be neatly indented. Indention is done by two spaces, no tabs. Same xml level = same indention, xml childs = one indention more, parents = one indention less. New xml tag = new line!
5. **Use comments** if you feel this might help: `<!– This is a comment –>`
6. **Only post working code**: code which you have tried, tested and found working. Not code which should or could work, but code which you know it works. If you need to post assumptions then clearly name it like this (and give away the reasons).
7. **Downloadable xml files must be ready to run without fault.** If you feel a particular snippet is worth posting then simply post the code without a filename (which would make it a downloadable file), or give it a txt extension. There must be no xml file which only holds fragments, incomplete xml syntax etc.
8. Please **avoid redundant entries**. This is not just a collection of macros - there is no point in posting the umpteenth macro which opens some windows which is already described in a bunch of other postings. Instead, put your comments in the discussion section - and if you find something to improve then edit what's already there, possibly with proper annotations.
9. **Adhere to the templates** (where provided), or copy similar articles and edit to your needs. This make it easier for the casual user as they will find what they seek always at the same spot and in the same context.
10. Try to follow the **naming conventions**. Of course you are free to name your macros however you like on your own machines. But in order to make a more concise impression why not follow the predicate-object-rule: name a macro which does something with anotherthing 'dosomething toanotherthing'. A macro which clears the programmer should be named 'Clear Programmer', and one which closes all windows should be called 'Close All Windows'. This is more human-readable. However, the according filenames and macro IDs might me nore object-like (e.g. xyz.macros.programmer.clear or abc.windows.closeAll).

These guidelines are work in progress - let us know if you feel they should be updated or amended.

Following these guidelines makes life much easier for you, for your co-authors, and for the readers. All your effort is highly appreciated - thanks for contributing!

From:
https://avosupport.de/wiki/ - **AVOSUPPORT**

Permanent link:
**https://avosupport.de/wiki/wiki_writing_guidelines?rev=1535135545**

Last update: **2018/08/24 18:32**