8cres Style Guide

8cres App
8cres
Release 1
Release Date: 22/05/2021
Revision 1: 25/05/2021

1. All swift source files end with the extension .swift
2. All Source files are encoded in UTF-8.
3. Comments describing the contents of a source file are optional.
4. A source file imports exactly the top-level modules that it needs.
5. Import statements are the first non-comment tokens in a source file.
6. In general, most source files contain only one top-level type, especially when the type declaration is large.
7. Each file and type uses some logical order.
8. Swift code has a column limit of 100 characters
9. Semicolons (;) are not used, either to terminate or separate statements.
10. There is at most one statement per line, and each statement is followed by a line break.
11. When a function call is line-wrapped, each argument is written on its own line, indented +2 from the original line.
12. When a control flow statement (such as if, guard, while, or for) is wrapped, the first continuation line is indented to the same position as the token following the control flow keyword
13. Horizontal alignment is forbidden.
14. Parentheses are not used around the top-most expression that follows an if, guard, while, or switch keyword.
15. Local variables are declared close to the point at which they are first used to minimize their scope.
16. Case statements are indented at the same level as the switch statement to which they belong
17. In general, there is only one case per line in an enum.
18. Trailing commas in array and dictionary literals are *required* when each element is placed on its own line.
19. Identifiers contain only 7-bit ASCII characters.
20. Initializer arguments that correspond directly to a stored property have the same name as the property.
21. Static and class properties that return instances of the declaring type are *not* suffixed with the name of the type.
22. Like other variables, global constants are lowerCamelCase.
23. Avoid redundancy
24. avoid ambiguity

25. prefer implicitness over explicitness unless being explicit improves readability and/or reduces ambiguity.
26. Any warnings that are able to be removed easily by the author must be removed.
27. Error types are used when there are multiple possible error states.