The Zen of Python

Use the document at <https://www.python.org/dev/peps/pep-0020/> to complete the following guiding principles:

Beautiful is better than \_\_\_\_ugly\_\_\_\_.

Explicit is better than \_\_\_\_implicit\_\_\_\_.

Simple is better than \_\_\_\_complex\_\_\_\_.

Complex is better than \_\_\_\_complicated\_\_\_\_.

Flat is better than \_\_\_\_nested\_\_\_\_.

Sparse is better than \_\_\_\_dense\_\_\_\_.

Readability \_\_\_\_counts\_\_\_\_.

Special cases aren't special enough to break the \_\_\_\_rules\_\_\_\_.

Although practicality beats \_\_\_\_purity\_\_\_\_.

Errors should never pass \_\_\_\_silently\_\_\_\_.

Unless explicitly \_\_\_\_silenced\_\_\_\_.

In the face of ambiguity, refuse the temptation to \_\_\_\_guess\_\_\_\_.

There should be \_\_\_\_one\_\_\_\_ -- and preferably only one --obvious way to do it.

Although that way may not be obvious at first unless you're \_\_\_\_Dutch\_\_\_\_.

Now is better than \_\_\_\_never\_\_\_\_.

Although \_\_\_\_never\_\_\_\_ is often better than \*right\* now.

If the implementation is hard to explain, it's a \_\_\_\_bad\_\_\_\_ idea.

If the implementation is easy to explain, it may be a \_\_\_\_good\_\_\_\_ idea.

Namespaces are one honking great idea -- let's do \_\_\_\_more\_\_\_\_ of those!