验证了“信息表达3.1Python入门（36页）-20210407” 和 “信息表达3.2Python进阶 - 20210414”（截至第20页以前）中可能存在疑问的一些说法。

斜体字部分全部引用自Python官方文档：PEP 8 -- Style Guide for Python Code

https://www.python.org/dev/peps/pep-0008/

3.1PPT

1. 第22页“每行不超过**80**个字符”

实际应为79个，*“Limit all lines to* ***a maximum of 79 characters.****”* 其目的一为*“The limits are chosen to* ***avoid wrapping in editors with the window width set to 80****, even if the tool places a marker glyph in the final column when wrapping lines.”*，即为了兼容一些传统的工作窗口只有80字符宽度的编辑器，以避免Python文件在编辑器内打开时因不确定的换行产生阅读困难；目的二为便于工作时打开多个窗口：*“Limiting the required editor window width makes it possible to* ***have several files open side by side.****”* 所以实际上这一规范并不十分严格，可以根据团队需求改变。

1. 第22页“不要使用反斜杠连接行。”

PEP8文档中建议使用圆括号，但是有例外，存在无法使用圆括号连接的情形：

*“Backslashes may still be appropriate at times. For example, long, multiple with-statements cannot use implicit continuation, so backslashes are acceptable:”*

*Example:*

*with open('/path/to/some/file/you/want/to/read') as file\_1, \*

*open('/path/to/some/file/being/written', 'w') as file\_2:*

*file\_2.write(file\_1.read())*

1. 25页至27页的空格规范

事实上这些空格规范都是建议规则，基本不影响程序运行结果。

ppt第25页中的三种示例都出自PEP8规范，与以下三个示例一一对应，因此都是三种规范都是存在的。但是三者都不影响程序正常执行。

*Immediately before a comma, semicolon, or colon:*

*# Correct:*

*if x == 4: print x, y; x, y = y, x*

*# Wrong:*

*if x == 4 : print x , y ; x , y = y , x*

*Immediately before the open parenthesis that starts the argument list of a function call:*

*# Correct:*

*spam(1)*

*# Wrong:*

*spam (1)*

*Immediately before the open parenthesis that starts an indexing or slicing:*

*# Correct:*

*dct['key'] = lst[index]*

*# Wrong:*

*dct ['key'] = lst [index]*

26页的两点和文档内容一致。

27页的“不要用空格来垂直对齐多行间的标记, 因为这会成为维护的负担”文档也有提到：

More than one space around an assignment (or other) operator to align it with another:

# Correct:

x = 1

y = 2

long\_variable = 3

# Wrong:

x = 1

y = 2

long\_variable = 3

综上所述，以上所有关于空格的规范都是语法风格指导（Style Guide），不会影响程序进行。事实上它们类似英语的书面语法规则，（个人推测）主要目的应该是便于人和编译器阅读代码。

但有一种情况例外：一行末尾多余的空格需要注意避免，因为它会使编译器无法识别

***“Avoid trailing whitespace anywhere****. Because it's usually invisible, it* ***can be confusing****: e.g. a backslash followed by a space and a newline does not count as a line continuation marker.”*

文档中举例为行末的“\+空格”不会被认为是将两行连起来的符号。经测试验证，用圆括号连接两行时无此问题，这可能是建议使用圆括号的原因之一。

测试代码：



即使用圆括号连接两行时行末的空格被编译器忽略，但使用反斜杠时编译器无法正常识别。

3.2ppt 14页

使用format时{}之间不能有空格，该规范确实存在。语法不规范会影响代码正常运行。代码运行结果如下：

>>> print("{}：计算机{}的CPU 占用率为{}%。".format("2016-12-31","PYTHON",10))

2016-12-31：计算机PYTHON的CPU 占用率为10%。

>>> print("{ }：计算机{}的CPU 占用率为{}%。".format("2016-12-31","PYTHON",10))

Traceback (most recent call last):

File "<stdin>", line 1, in <module>

KeyError: ' '

>>> print("{0}：计算机{1}的CPU 占用率为{2}%。".format("2016-12-31","PYTHON",10))

2016-12-31：计算机PYTHON的CPU 占用率为10%。

>>> print("{0 }：计算机{1}的CPU 占用率为{2}%。".format("2016-12-31","PYTHON",10))

Traceback (most recent call last):

File "<stdin>", line 1, in <module>

KeyError: '0 '