

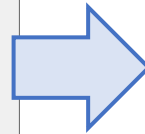
Anaconda Prompt (インタラクティブシェル)の タブの扱いに合わせたサクラエディタの設定

Anaconda Prompt で for や if を試す

- Tabキーでインデントできない...
 - IndentationError: expected an indented block 「インデントされたブロックを期待していたよ」

```
Anaconda Prompt - python
(base) C:\Users\imura>python
Python 3.8.8 (default, Apr 13 2021, 15:08:03) [MSC v.1916 64 bit (AMD64)] :: Anaconda, Inc. on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> for i in range(10):
... _
```

ここでTabキーを押すと...



```
Anaconda Prompt - python
res = ensure_str(readline_hook(prompt))
File "C:\Users\imura\Anaconda3\lib\site-packages\pyreadline\rlmain.py", line 5
71, in readline
self._readline_from_keyboard()
File "C:\Users\imura\Anaconda3\lib\site-packages\pyreadline\rlmain.py", line 5
36, in _readline_from_keyboard
if self._readline_from_keyboard_poll():
File "C:\Users\imura\Anaconda3\lib\site-packages\pyreadline\rlmain.py", line 5
56, in _readline_from_keyboard_poll
result = self.mode.process_keyevent(event.keyinfo)
File "C:\Users\imura\Anaconda3\lib\site-packages\pyreadline\modes\emacs.py", line 243, in process_keyevent
r = self.process_keyevent_queue[-1](keyinfo)
File "C:\Users\imura\Anaconda3\lib\site-packages\pyreadline\modes\emacs.py", line 286, in _process_keyevent
r = dispatch_func(keyinfo)
File "C:\Users\imura\Anaconda3\lib\site-packages\pyreadline\modes\basemode.py", line 257, in complete
completions = self._get_completions()
File "C:\Users\imura\Anaconda3\lib\site-packages\pyreadline\modes\basemode.py", line 200, in _get_completions
r = self.completer(ensure_unicode(text), i)
File "C:\Users\imura\Anaconda3\lib\rlcompleter.py", line 80, in complete
readline.redisplay()
AttributeError: module 'readline' has no attribute 'redisplay'
File "<stdin>", line 2
^
IndentationError: expected an indented block
>>>
```

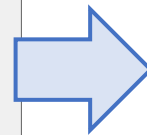
インデントされずエラーが出る

[解決法] Anaconda Prompt では空白でインデントを付ける

- スペースキーを押して空白を入力してインデントするようにしよう。
 - 空白の数は何個でもよいですが、同一ブロックでは同じインデントである必要がある。

```
Anaconda Prompt - python
(base) C:\Users\imura>python
Python 3.8.8 (default, Apr 13 2021, 15:08:03) [MSC v.1916 64 bit (AMD64)] :: Anaconda, Inc. on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> for i in range(10):
... -
```

スペースでインデントする

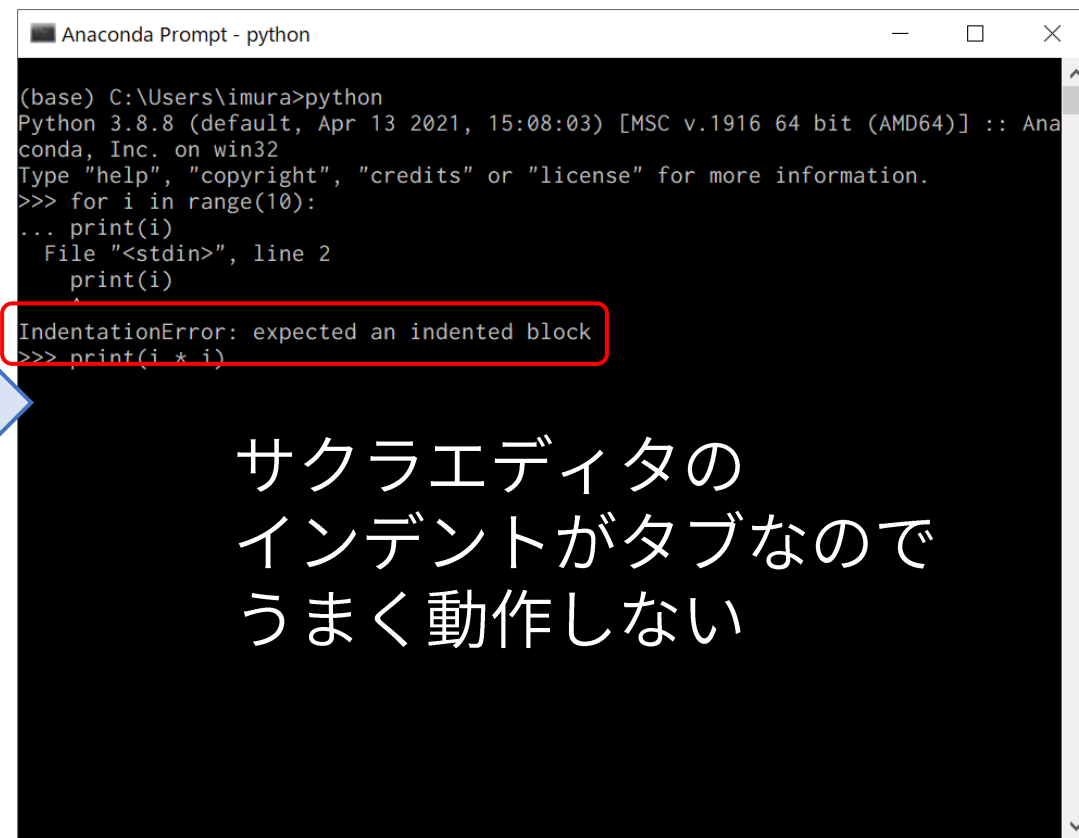
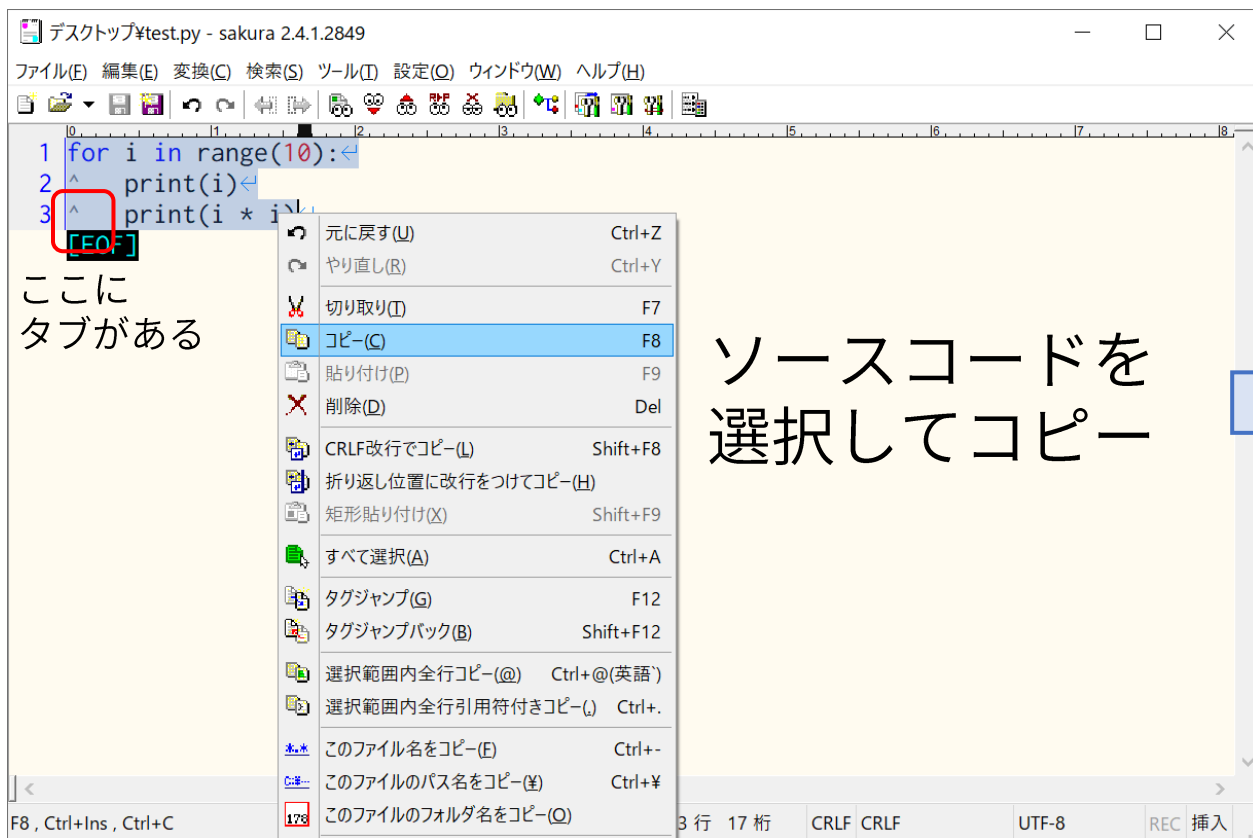


```
Anaconda Prompt - python
(base) C:\Users\imura>python
Python 3.8.8 (default, Apr 13 2021, 15:08:03) [MSC v.1916 64 bit (AMD64)] :: Anaconda, Inc. on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> for i in range(10):
...     print(i)
...     print(i * i)
...
0
0
1
1
2
4
3
9
4
16
5
25
6
36
7
49
8
64
9
81
>>>
```

適切に動作した

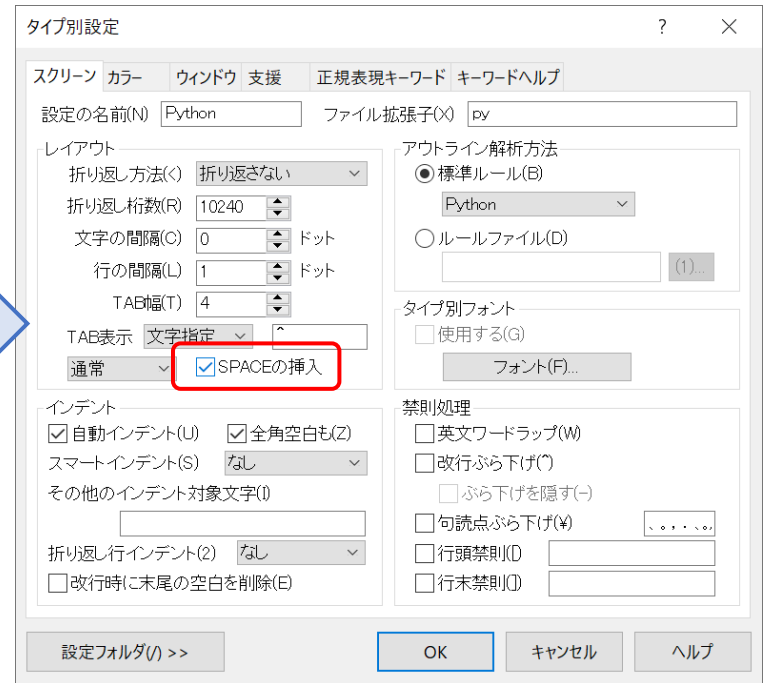
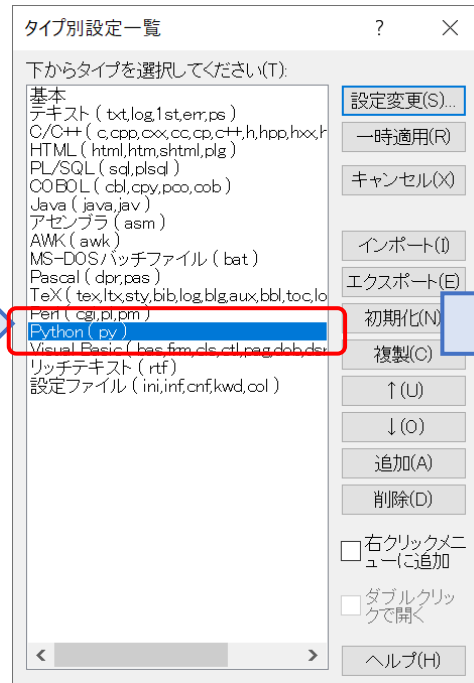
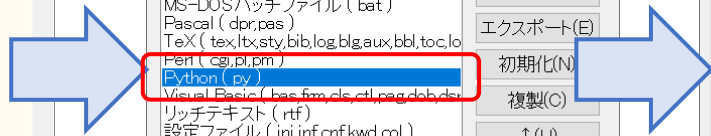
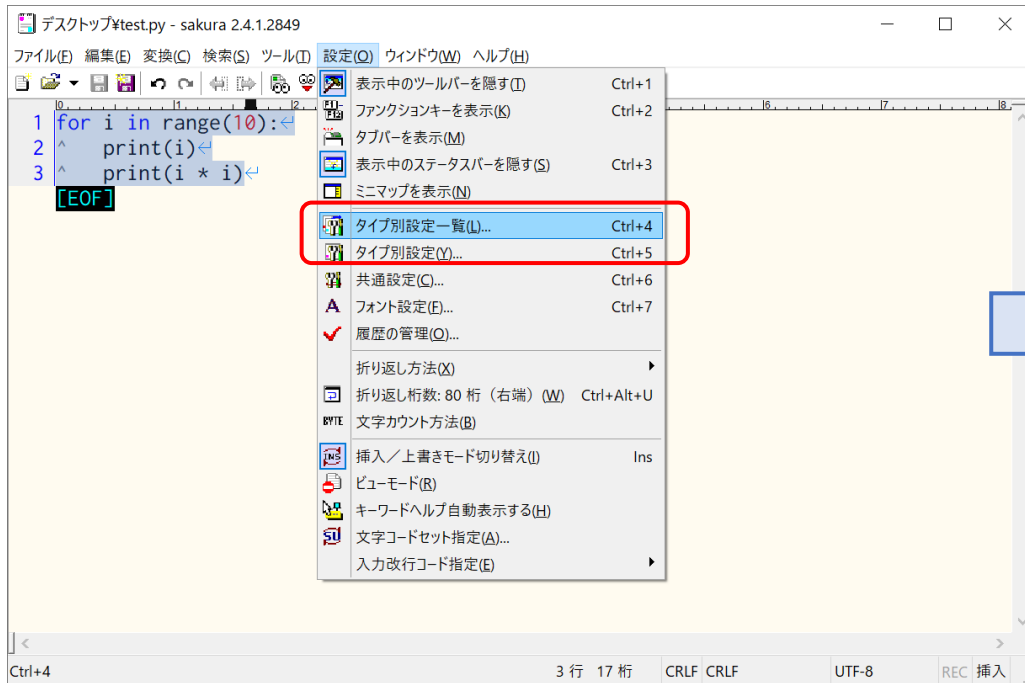
サクラエディタからコピー&ペーストして試す

- エディタで入力したソースコードの一部の動作を確認したい。
- サクラエディタからAnaconda Prompt にコピー&ペースト...



[解決法] サクラエディタでタブを空白に展開する

- Tabキーを押すと，適切な数の空白を入力する設定にしておく。
 - 設定 > タイプ別設定一覧 → Python (無い場合: 基本)を選択 > 設定変更 → スクリーン > レイアウト > SPACEの挿入 にチェック



メリット・デメリット

● メリット

- サクラエディタで入力したソースコードを，一部分だけちょっとインタラクティブシェルで試している，といったテストが柔軟に行える．
- 他のエディタで編集する場合に，タブ幅の違いによるインデントの崩れが生じない．
- インデントを付ける際にはこれまでと同様Tabキーで行える．

● デメリット

- ソースコードからブロックの一部を削除する際に，タブ1個を削除すればよかったのが，空白を複数個削除しないといけなくなる．