





Pythonista







SCRIPT LIBRARY

-  This iPhone >
-  iCloud >

FAVORITES

-  Add Favorite...
-  Examples >

EXTERNAL FILES

-  Open...
-  peakdetect.py
-  File Provider Storage >
-  Python Modules >
-  File Templates >
-  Trash >

iPhoneに保存されているファイル名の表示

お気に入り

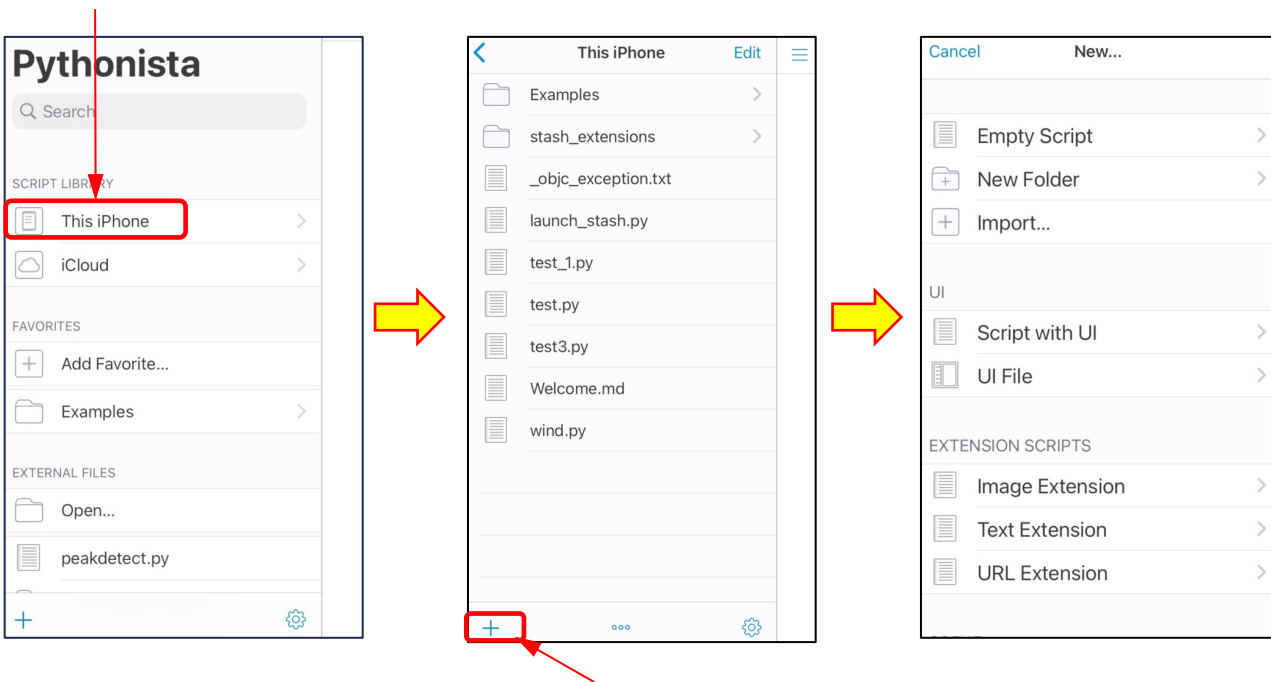
Pythonista3に用意されているサンプルフォルダ

ファイル名を選択して開く

Pythonのモジュール類のフォルダ

テンプレートファイルとして保存されたものの保存フォルダ

ゴミ箱

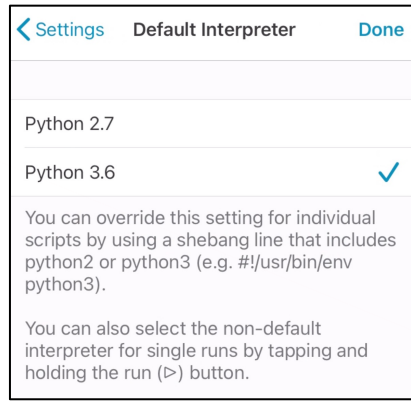
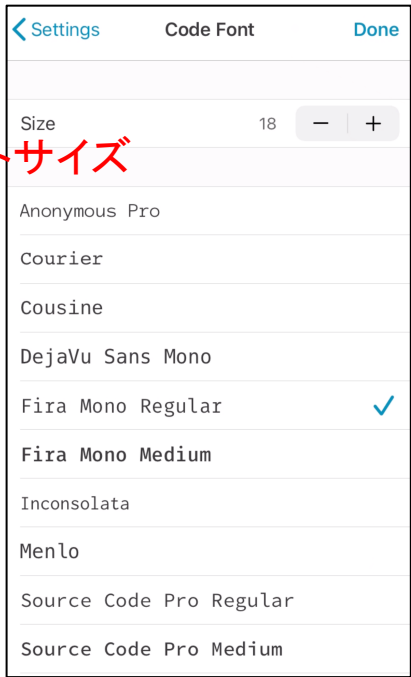


空のスク립トファイルの作成
新しいフォルダの作成
ファイルやイメージファイルのインポート
ユーザーインターフェースを作成するための項目
エクステンション関係のスク립ト作成

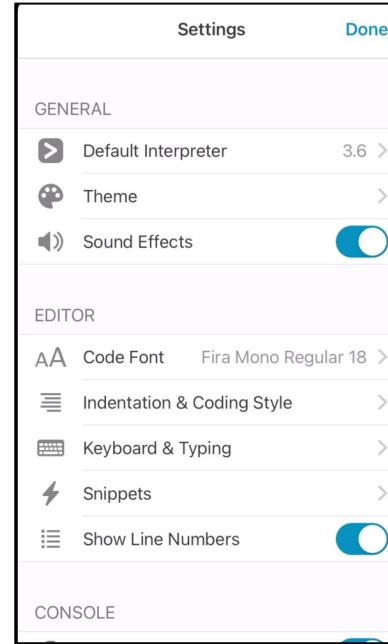
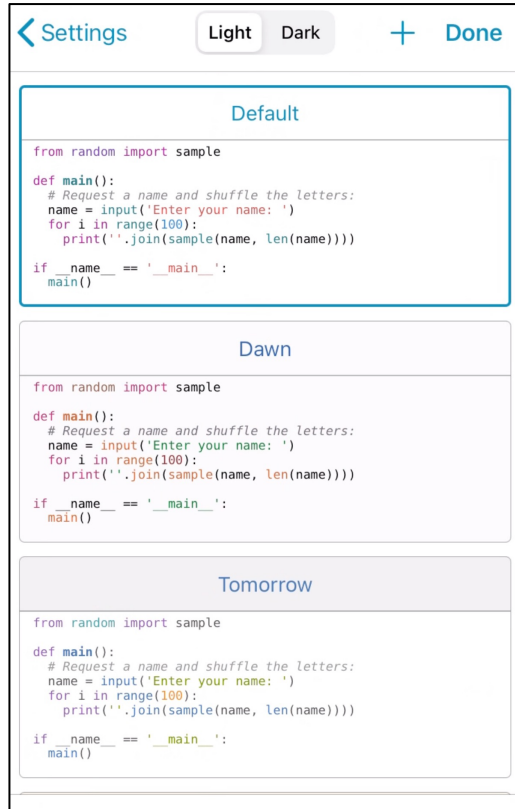
CUI(キャラクタユーザーインターフェース) : キーボード入力の文字(キャラクタ)で行う方式。
コマンドプロント(Macはターミナル)を用いて操作

GUI(グラフィカルユーザーインターフェース) : カーソルやファイルなどのアイコン(グラフィカル)を用いて操作

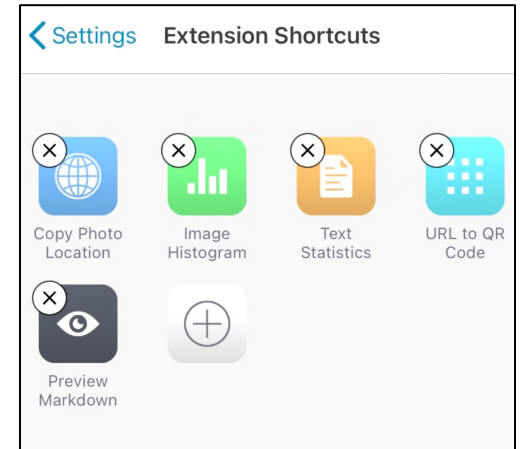
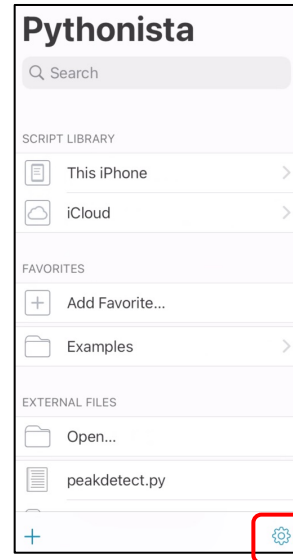
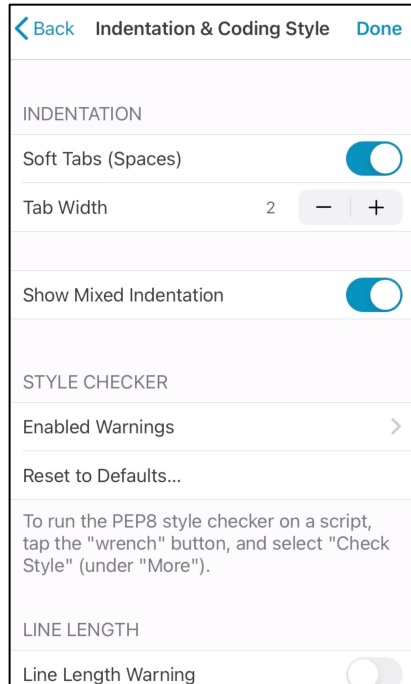
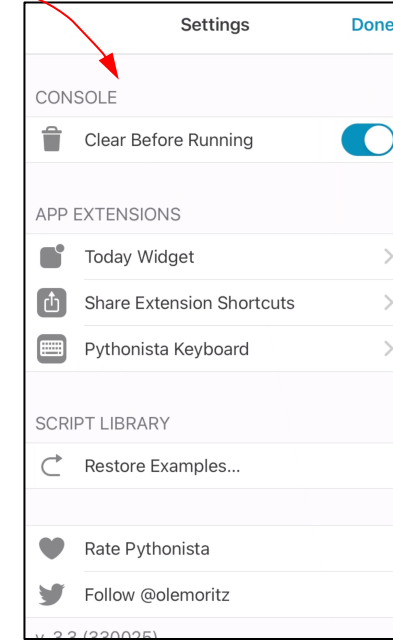
フォントサイズ



好きな配色を選択



①歯車クリック



情報を消す

```
> Console ▾ ⓘ ✕ Clear  
>>> x=100  
>>> y=200  
>>> z=(x,y,x+y,x*y)  
>>> print(z)  
(100, 200, 300, 20000)  
  
> z=(x,y,x+y,x*y)|
```

「i」をクリックすると

以前に使用したコード
が現れる

```
> Console ▾ ⓘ ✕ Clear  
Object Inspector ⓘ  
x 100  
y 200  
▶ z (100, 200, 300, 20000)  
▶ __annotations__ {}  
▶ __builtins__ <module 'builtins' (built-i...  
__cached__ None  
__doc__ None  
  
> z=(x,y,x+y,x*y) ▾ ▴
```

```
import cv2; import numpy as np
import scipy.signal as signal
import matplotlib.pyplot as plt
frame=0; dataR=[]; dataG=[];
dataB[]
vidcap=cv2.VideoCapture('/Users/
...hoge.mov')
success,img=vidcap.read()
while success:

    dataB.append(average(img[400:461
,400:461,0]))

    dataG.append(average(img[400:461
,400:461,1]))

    dataR.append(average(img[400:461
,400:461,2]))
    frames +=1; success,
img=vidcap.read()
x=np.arange(0, 1/240*frames,1/240)
plt.plot(x, dataB, color='blue')
plt.plot(x, dataG, color='green')
plt.plot(x, dataR, color='red')
```

文法を解析して自動で
インデント位置を調整

青	数値
緑	関数
紫	モジュール
赤	テキスト