

Pythonプログラミング

追加補足版



■ 変数名を付けるときの注意点

変数の命名ルールは大きく分けて3項目があります (P.23 参照)。このルールが守られていない場合、プログラムを実行することができません。

- ・半角英数字 (a~z, A~Z, 0~9) とアンダースコア (_) のみを使う
- ・数字のみ、または先頭が数字の変数名は禁止
- ・予約語と同じ変数名は禁止

Pythonではあらかじめ用意された組み込み関数と同じ名前を変数に付けることは、上記のルールに当てはまらないため可能です。しかし、プログラムによってはエラーになったり、予期せぬ結果になったりする場合がありますので注意が必要です。

次のプログラムは、変数strに文字列Yamada Ichiroを代入し、print関数で変数strが記憶している値を表示します。

▶ ex_1_1

```
01 str = 'Yamada Ichiro'
02 print(str)
```

▶ 実行結果

```
01 Yamada Ichiro
```

次のプログラムも実行してみましょう。

▶ ex_1_2

```
01 str = 'Yamada Ichiro'
02 age = 24
03 text = str(age)
04 print(str, text)
```

▶ 実行結果

```
01 Traceback (most recent call last):
02   File "/Users/admin/Document/python/ex_1_2.py", line 3, in <module>
03     text = str(age)
04   TypeError: 'str' object is not callable
```

実行結果の4行目は、「TypeError: 'str'オブジェクトは呼び出しできません。」という意味で、文字列を関数のように呼び出そうとしたことを表します。

組み込み関数である str 関数は、数値や値を str 型に変換する働きがあります。しかし、あとから作った変数 str を優先して使うようになってしまい、変数 str を関数のように呼び出そうとしてエラーが発生したのです。

もう1つ別の例も見てみましょう。

▶ ex_1_3

```
01 name = 'Yamada Ichiro'  
02 type = '正社員'  
03 age = 25  
04 print(type(age))
```

▶ 実行結果

```
01 Traceback (most recent call last):  
02   File "/Users/admin/Document/python/ex_1_3.py", line 4, in <module>  
03     print(type(age))  
04   TypeError: 'str' object is not callable
```

このプログラムも同様のエラーが発生します。type 関数 (P.30) は変数や値の型を調べたいときに使いますが、あとから作った変数 type を優先するため、変数 type を関数のように呼び出そうとしてエラーが発生します。

このように、組み込み関数と同じ名前の変数を作ることはできますが、重複した名前の組み込み関数が呼び出せなくなります。また次のような問題も考えられます。

- ・プログラムを読んだほかの人が混乱しやすくなる
- ・エラーが発生した際に、原因を見つけるのが難しくなる場合がある

Python の組み込み関数の名前は次の公式ドキュメントを参照してください。特別な理由がないかぎり、下記にある組み込み関数と同じ名前の変数名は避けたほうがよいでしょう。

- ・公式ドキュメント (組み込み関数)

<https://docs.python.org/3/library/functions.html>

■ 型アノテーション

「この変数にはstr型の値を入れたい」ということを明示する方法として、**型アノテーション**と呼ばれるものがあります。アノテーションは「注釈」という意味です。

Pythonは動的に型を付けるため(P.30参照)、型を明示する必要はありません。しかし、複数人でプログラムを作る場合や、複雑なプログラムを作る場合など、変数に型の注釈を付けておくことで変数の意図をより明確に伝えられます。

型アノテーションは次のように、変数名と=(イコール)の間に、:(コロン)と型名を入れます。

・型アノテーションの書き方

```
変数名 : 型名 = 値
```

次のプログラムでは、変数name、変数ageで型アノテーションを行っています。

▶ ex_1_4

```
01 name: str = 'Yamada Ichiro'  
02 age: int = 25  
03 print(name)  
04 print(age)
```

▶ 実行結果

```
01 Yamada Ichiro  
02 25
```

このように記述しておくことで、「nameという名前の変数には、str型の値を入れてほしい」ということを、プログラムを見た人に伝えられます。

▶ COLUMN 変数名に接頭語または接尾語を使う

変数にどのような値を入れてほしいかを変数名で明示したい場合は、接頭語または接尾語を付けるとよいでしょう。その名のとおり、接頭語は変数名や関数名の頭に付ける文字列、接尾語は末尾に付ける文字列です。

・接頭語と接尾語の例

```
str_name, int_tellnumber, list_subject ..... 接頭語の場合  
name_str, tellnumber_int, subject_list ..... 接尾語の場合
```