

Python

Akihiro Minamino

August 14, 2020

1 変数

1.1 変数、名前、オブジェクト

Python 変数の重要なポイントは、変数はただの名前だということである。データをいれているオブジェクトに名前を付けるだけである。名前は値自体ではなく値の参照である。名前は、オブジェクトに貼るポストイットのようなものである。

1.2 数値

Python の数字の並びは、リテラル¹の整数と見なされる。

Python では、=記号の右辺の式がまず計算され、次に左辺の変数に代入が行われる。

1.3 基数

整数は、プレフィックスで基数を指定しない限り、10 進（基数 10）と見なされる。基数は、「桁上り」しなければならなくなるまで、何個の数字を使えるかを示す。

Python では、10 進以外に 3 種類の基数を使ってリテラル整数を表す。

- 0b は 2 進（基数 2）
- 0o は 8 進（基数 8）
- 0x は 16 進（基数 16）

インタプリタは、10 進整数として、整数を表示する。

```
1 >>> 10
2 10
3 >>> 0b10
```

¹リテラルとは、プログラムのソースコードにおいて使用される、数値や文字列を直接に記述した定数のことである。変数の対義語であり、変更されないことを前提とした値である。

```
4 2
5 >>> 0x10
6 16
```

1.4 型の変換

Python の整数以外のデータ型を整数に変換するには、`int()` 関数を使う。この関数は整数部だけを残し、小数部を切り捨てる。

`int()` は、数字でできた文字列を整数に変換する。しかし、小数点や指数部を含む文字列は処理しない。

```
1 >>> int('98.6')
2 ValueError: ...
3 >>> int('1.0e4')
4 ValueError: ...
```

1.5 文字列

文字列は、文字のシーケンスである。

他の言語と異なり、Python の文字列はイミュータブルである。つまり、文字列をその場で書き換えることができない。

Python 文字列は、シングルクォートかダブルクォートで文字を囲んで作る。どちらのクォートを使っても、Python はまったく同じように扱う。2 種類のクォート文字を使えるようにしている理由は、クォート文字を含む文字列を作りやすくするためである。ダブルクォートで文字列を作るときは、文字列内にシングルクォートを入れることができ、シングルクォートで文字列を作るときは、文字列内にダブルクォートを入れることができる。

```
1 >>> "Nay," said the naysayer."
2 "Nay," said the naysayer."
3 >>> 'The rare double quote in captivity: "."'
4 'The rare double quote in captivity: "."'
```