

Python 入門

真偽値と条件分岐

プログラムの基本の1つである条件分岐について扱います。

「**ある条件を満たすときだけ**処理を行いたい」というときに使うのが if 文です。

if 条件式:
処理（条件式が真のときに実行される）

```
N = int(input())

if N < 10: # 条件式「N が 10 未満」
    print("N は 10 未満です")
print("終了")
```

5

N は 10 未満です
終了

15

終了

比較演算子

演算子	意味
<code>x == y</code>	<code>x</code> と <code>y</code> は等しい
<code>x != y</code>	<code>x</code> と <code>y</code> は等しくない
<code>x < y</code>	<code>x</code> は <code>y</code> より小さい
<code>x > y</code>	<code>x</code> は <code>y</code> より大きい
<code>x <= y</code>	<code>x</code> は <code>y</code> 以下
<code>x >= y</code>	<code>x</code> は <code>y</code> 以上

論理演算子

演算子	意味
<code>not 条件式</code>	<code>条件式</code> が成り立た ない
<code>条件式1 and 条件式2</code>	<code>条件式1</code> かつ <code>条件式2</code> が成り立つ
<code>条件式1 or 条件式2</code>	<code>条件式1</code> と <code>条件式2</code> の 少なくとも一方 が成り立つ

```
N = int(input())

if 0 < N and N < 10:
    print("N は 1 桁です")

if N % 2 == 0 or N % 3 == 0:
    print("N は 2 または 3 の倍数です")
```

条件式が真の場合と偽の場合、両方の処理をしたいとき

```
if 条件式:  
    処理1
```

```
if not 条件式:  
    処理2
```

でもいいですが。。

```
if 条件式:  
    処理1
```

```
else:  
    処理2
```

のように **else** を使うのが楽です

処理がたくさんあるとき

elif を使いましょう。

```
if 条件式1:  
    処理1  
  
elif 条件式2:  
    処理2  
  
elif 条件式3:  
    処理3  
  
else:  
    処理4
```

条件式1を満たすか？ → 条件式2を満たすか？ → 条件式3を満たすか？ ...という順に上から条件が調べられていきます。

例

```
N = int(input())

if N < 10:
    print("N は 1 桁です")

elif N < 100:
    print("N は 2 桁です")

elif N < 1000:
    print("N は 3 桁です")

else:
    print("N はとても大きいです")
```

bool 型変数

真であるか 偽 であるかを値として持つ型として bool 型があります。

```
A = True
B = False

if A:
    print("Hello!") # 実行される

if B:
    print("Goodbye!") # 実行されない
```

演習

これまでの内容で APG4bPython の演習問題

- EX 6
- EX 7

を解くことができます。実際に手を動かしてやってみましょう！