

論理演算と文字列・入出力

Python プログラミング入門

複数条件の組み合わせ、文字列の加工、ユーザーとの対話— 実践的なプログラミングの基礎



論理演算子



文字列操作



入出力

この知識があると何ができる？

論理演算・文字列・入出力は、「使えるプログラム」を作るための必須スキルです



条件の組み合わせ判定

年齢とチケットの
両方を確認して入場可否
を自動判定



文字列の組み立て

名前・住所・価格など
を組み合わせ
メッセージを生成



ユーザーとの対話

入力を受け取って
計算し、結果を
見やすく表示



生成AIにコードを書かせる場合でも、これらを理解していれば「指示の精度」が格段に上がります

and 演算子(かつ)

すべての条件がTrueのときだけTrueを返す

動作イメージ

True and True → True

True and False → False

False and True → False

False and False → False

CODE

```
# 大人かつ免許を持っている？
age = 25
has_license = True

can_drive = (age >= 18) and has_license
print(can_drive)
→ True
```

両方の条件を満たして初めてTrue。片方でもFalseなら結果はFalse

or 演算子 (または)

いずれかの条件が True なら True を返す

動作イメージ

True or True → True

True or False → True

False or True → True

False or False → False

CODE

```
# 週末または祝日なら休み
is_saturday = False
is_sunday = True

is_day_off = is_saturday or is_sunday
print(is_day_off)
→ True
```

どれか1つでも条件を満たせば True。すべてが False のときだけ False

not 演算子(否定)

条件を反転させる — TrueをFalseに、FalseをTrueに

動作イメージ

not True → False

not False → True

CODE

```
# 雨でなければ外出する
is_raining = False
can_go_out = not is_raining
→ True
```

複雑な条件の組み合わせ

CODE

```
# クーポン適用条件
can_use = (is_member or is_first) and is_campaign and (price >= 3000)
→ True
```

文字列の結合と繰り返し

+ で文字列を結合

```
# 名前を結合する
last = "山田"
first = "太郎"
full = last + first

print(full)
→ 山田太郎
```

* で文字列を繰り返し

```
# 区切り線を作る
print("=" * 20)
→ =====

print("♪ " * 3)
→ ♪ ♪ ♪
```

文字列と数値の変換

`str()`・`int()`・`float()` で型を変換する

`str()`

数値 → 文字列

```
str(25)  
→ "25"
```

`int()`

文字列 → 整数

```
int("100")  
→ 100
```

`float()`

文字列 → 小数

```
float("3.14")  
→ 3.14
```

CODE

```
# 数値を文字列に結合するには str() が必要  
age = 25  
message = "私は" + str(age) + "歳です"  
→ 私は25歳です
```

format() メソッド

文字列に値を埋め込む、読みやすい方法

基本的な使い方

```
name = "田中"  
age = 30  
  
# {} に値が順番に入る  
"名前: {}, 年齢: {}".format(name, age)  
→ 名前: 田中, 年齢: 30
```

数値のフォーマット

```
# 小数点以下2桁  
"{:.2f}".format(3.14159)  
→ 3.14  
  
# 3桁区切り  
"{:,}円".format(1234567)  
→ 1,234,567円
```

+ で結合するよりも format() を使うと、見やすく柔軟なコードになる

print() と input()

画面への表示と、ユーザーからの入力

print() — 画面に表示

```
# 区切り文字を変更 (sep)
print("A", "B", "C", sep="-")
→ A-B-C

# 改行を制御 (end)
print("処理中", end="...")
print("完了!")
→ 処理中...完了!
```

input() — 入力を受け取る

```
# 文字列として受け取る
name = input("名前: ")
→ 太郎

# 数値として使うには変換が必要
age = int(input("年齢: "))
print(age + 1)
→ 26
```

注意: input() の結果は常に文字列型 (str)。数値として使うには int() や float() で変換が必要

実用シーン：お買い物計算

入力 → 計算 → 出力 を組み合わせた実践例

CODE

```
# ユーザーから入力を受け取る
price = int(input("商品の価格: "))
qty = int(input("購入個数: "))

# 計算する
subtotal = price * qty
tax = subtotal * 0.1
total = subtotal + tax
free_ship = subtotal >= 10000

# 結果を表示する
print("合計: {:.0f}円".format(total))
print("送料無料: {}".format(free_ship))
```

まとめ



論理演算子

and: すべてTrueのときTrue

or: いずれかTrueのときTrue

not: 条件を反転させる



文字列操作

+ で文字列を結合

* で文字列を繰り返す

format() で値を埋め込む



入出力

print() で画面に表示

input() で入力を受け取る

input()の結果は常に文字列



次のステップ → 条件分岐 (if文) でプログラムに判断させる方法を学ぶ