

13章 変数のスコープを理解しよう

13章 変数のスコープを理解しよう

| 変数の有効範囲を知り、関数の内外で変数を正しく使う方法を学びます

本章の目標

- スコープ（変数の有効範囲）について知り、関数の内外で変数を正しく使えるようになること

13章 なぜスコープを学ぶのか

変数の有効範囲を知っておかないとエラーにつながります

- 中身が入っていない
- 想定とは異なる中身が入っている

関数の中で変数を使うときは、使える範囲（有効範囲）に注意が必要です

| 変数を使える範囲（有効範囲）のことです

- 変数の値を取得したり、変更したりできる範囲
- 代表的なスコープは以下の2つ
 - ローカルスコープ：関数の中
 - グローバルスコープ：関数の外

13章 Pythonのスコープ (LEGB)

| Pythonのスコープは4つに分かれます

スコープ	説明
Local (ローカル)	関数の中
Enclosing (エンクロージング)	外側の関数
Global (グローバル)	関数の外
Built-in (ビルトイン)	組み込み

本章では代表的なローカルスコープとグローバルスコープを学びます

13章 ローカルスコープ

```
user_name = "侍花子"
```

グローバルスコープ

```
def show_user_name():
    user_name = "侍太郎"
    print(user_name)
```

ローカルスコープ

```
show_user_name()
print(user_name)
```

グローバルスコープ

| ローカルスコープで定義された変数をローカル変数といいます

- ローカル変数はその関数の中でしか使えない
- 「関数の中で定義した変数はその関数の中でしか使えない」

13章 ローカル変数の正しい使い方

| ローカルスコープの範囲内でローカル変数を使う

```
def show_user_name():
    # ローカル変数を定義する
    user_name = "侍太郎"

    # ローカルスコープの範囲内でローカル変数を使う
    print(user_name)

show_user_name()
```

13章 ローカル変数の正しい使い方

✓
0
秒

```
def show_user_name():
    # ローカル変数を定義する
    user_name = "侍太郎"

    # ローカルスコープの範囲内でローカル変数を使う
    print(user_name)

show_user_name()
```

侍太郎

13章 ローカル変数の間違った使い方

| ローカルスコープの範囲外でローカル変数を使う

```
def show_user_name():
    # ローカル変数を定義する
    user_name = "侍太郎"
    print(user_name)

show_user_name()

# ローカルスコープの範囲外でローカル変数を使う（エラー）
print(user_name)
```

13章 ローカル変数の間違った使い方



```
def show_user_name():
    # ローカル変数を定義する
    user_name = "侍太郎"

    # ローカルスコープの範囲内でローカル変数を使う
    print(user_name)

show_user_name()

# ローカルスコープの範囲外でローカル変数を使う(エラーが発生する)
print(user_name)
```

侍太郎

```
NameError          Traceback (most recent call last)
<ipython-input-2-1b255860fdae> in <module>
      9
     10 # ローカルスコープの範囲外でローカル変数を使う(エラーが発生する)
--> 11 print(user_name)
```

NameError: name 'user_name' is not defined

SEARCH STACK OVERFLOW

13章 ローカル変数の間違った使い方

「未定義の変数user_name」 というエラーが表示されます

- 変数user_nameは関数の中で定義したローカル変数
- ローカルスコープの範囲外（関数の外）では未定義として扱われる

13章 グローバルスコープ

```
user_name = "侍花子"
```

グローバルスコープ

```
def show_user_name():
    user_name = "侍太郎"
    print(user_name)
```

ローカルスコープ

```
show_user_name()
print(user_name)
```

グローバルスコープ

| グローバルスコープで定義された変数をグローバル変数といいます

- グローバル変数はプログラム全体でアクセス可能
- 関数の中でも扱える

13章 グローバル変数の使い方

グローバルスコープの範囲内でグローバル変数を使う

```
# グローバル変数を定義する
user_name = "侍花子"

def show_user_name():
    # ローカル変数を定義する
    user_name = "侍太郎"
    print(user_name)

show_user_name()

# グローバルスコープの範囲内でグローバル変数を使う
print(user_name)
```

13章 グローバル変数の使い方

✓
0 秒

```
▶ # グローバル変数を定義する  
user_name = "侍花子"  
  
def show_user_name():  
    # ローカル変数を定義する  
    user_name = "侍太郎"  
  
    # ローカルスコープの範囲内でローカル変数を使う  
    print(user_name)  
  
show_user_name()  
  
# グローバルスコープの範囲内でグローバル変数を使う  
print(user_name)
```

侍太郎
侍花子

13章 グローバル変数とローカル変数は別物

| 同じ変数名でも、関数の中で値が更新されていません

- グローバル変数として定義した `user_name`
- ローカル変数として定義した `user_name`

これらは別物として扱われます

13章 関数の中でグローバル変数を使う

| ローカル変数がない場合、グローバル変数が使われます

```
# グローバル変数を定義する
user_name = "侍花子"

def show_user_name():
    # ローカル変数を定義しない場合
    # user_name = "侍太郎"
    print(user_name)  # グローバル変数が使われる

show_user_name()
print(user_name)
```

13章 関数の中でグローバル変数を使う



```
# グローバル変数を定義する
user_name = "侍花子"

def show_user_name():
    # ローカル変数を定義する
    # user_name = "侍太郎"

    # グローバルスコープの範囲外でグローバル変数を使う(エラーは発生しない)
    print(user_name)

show_user_name()

# グローバルスコープの範囲内でグローバル変数を使う
print(user_name)
```

侍花子
侍花子

| 本章では以下の内容を学習しました

スコープとは

- 変数を使える範囲（有効範囲）のこと
- ローカルスコープ（関数の中）とグローバルスコープ（関数の外）の2つに分けられる

| 本章では以下の内容を学習しました

ローカル変数とグローバル変数

- ローカルスコープで定義された変数をローカル変数という
- グローバルスコープで定義された変数をグローバル変数という
- 同じ変数名でもローカル変数とグローバル変数は別物として扱われる
- 関数外でローカル変数を使った場合、エラーが発生する