

第3回 Python基礎 (2)

制御構造

大妻女子大学 社会情報学部 WebプログラミングI

担当：余野

出席認証コード: 3343

授業資料: <https://x.gd/NoqkC> (大学のアカウントでgoogleにログインしていることが必要)

本日の流れ (約80分)

1. 前回の復習 (20分 - 課題含む)
2. 条件分岐 (if文) (15分)
3. 繰り返し (forループ) (15分)
4. 繰り返し (whileループ) (10分)
5. ネストした制御構造 (10分)
6. まとめと次回予告 (10分)

目標: `if` , `for` , `while` を使いこなし、プログラムの流れを自在に操る！

1. 前回の復習 (20分 - 課題含む)

復習ポイント

- 変数, データ型 (`int` , `float` , `str` , `list` , `bool`)
- 演算 (算術, 比較, 論理)
- 文字列・リスト操作 (基本)
- `input()` , `print()` , 型変換 (`int()` , `str()`)
- Streamlit: `st.write()` , `st.text_input()`

次のスライドで、これらの知識を使ったStreamlitの復習課題を行います。

1. (続き) 復習課題 (Streamlit)

- **環境:** GitHub Codespaces上で実施します。
- **ファイル:** `src/lecture03/app.py` で以下を試してみよう！
- **目的:** 前回学んだPythonの基本とStreamlitの入出力を組み合わせる練習。
- **課題例:**
 - かんたん計算機:** `st.text_input` で2つの数を入力 -> 和・差・積・商を計算して `st.write` で表示。
 - 挨拶メッセージ:** `st.text_input` で名前と年齢を入力 -> 「こんにちは、〇〇さん！あなたは△△歳ですね。」と `st.write` で表示。
 - BMI計算アプリ:** `st.text_input` で身長(m)と体重(kg)を入力 -> BMIを計算して `st.write` で表示。

復習課題環境: Codespaces と Streamlit 実行

1. Codespacesの起動:

- 以下のテンプレートリポジトリURLにアクセス:

```
https://github.com/kyouto-yono-ac/web_programming_2025
```

- 緑色の `Use this template` ボタンをクリックし、「Open in codespace」を選択。
- 少し待つと、ブラウザにVS Codeのようなエディタが表示されます。

2. 実装:

- `src/lecture03/app.py` を編集します

3. Streamlitアプリの実行:

- Codespacesエディタ下部などの **ターミナル** で以下のコマンドを入力して実行:

```
streamlit run src/lecture03/app.py
```

- 実行後、右下に表示されるポップアップの「ブラウザーで開く」ボタンをクリックするか、「ポート」タブに表示されるURLにアクセスしてアプリを確認します。

2. 条件分岐 (if文) (15分)

特定の条件に応じて処理を変えたい

```
score = 75

if score >= 80:
    print("優")
elif score >= 60: # else if の略
    print("良")
elif score >= 0:
    print("可")
else:
    print("無効な点数です")
```

- 比較演算子 (`==` , `!=` , `<` , `>` ...) と論理演算子 (`and` , `or` , `not`) で条件を作る。
- **インデント (字下げ)** がブロックを示す重要な役割を持つ！

3. 繰り返し処理 (forループ) (15分)

リストの各要素や、決まった回数だけ処理を繰り返したい

```
# リストの要素を順に処理
fruits = ["apple", "banana", "cherry"]
for fruit in fruits:
    print(f"I like {fruit}")

# 0から4まで繰り返す (計5回)
for i in range(5):
    print(f"現在の数値: {i}")

# range(開始, 終了[, ステップ])
for i in range(1, 6, 2): # 1から始まり、6未満まで、2ずつ増加 (1, 3, 5)
    print(i)
```

forループ: `break` と `continue`

- `break` : ループを完全に **中断** して抜ける。
- `continue` : 現在の回の処理を **スキップ** して、次の繰り返しに進む。

```
for i in range(10): # 0から9まで
    if i == 7:
        print("7になったので中断します")
        break # ループ全体を終了
    if i % 2 != 0: # 奇数だったら
        continue # print(i) をスキップして次の i へ
    print(i) # 偶数のみ出力される (0, 2, 4, 6)
```


4. 繰り返し処理 (whileループ) (10分)

特定の条件が満たされている "間"、処理を繰り返したい
(繰り返しの回数が事前に分からない場合に便利)

基本的な形:

```
while 条件式:  
    # 条件式がTrueの間、実行される処理  
    # ...  
    # 注意: ループ内で条件式がいつかFalseになるようにする!
```

例: カウンタ変数を使う

```
count = 0  
while count < 3:  
    print(f"現在のカウンタ: {count}")  
    count += 1 # カウンタを増やす (忘れると無限ループ!)  
print("ループ終了") # 0, 1, 2 が表示された後に出力
```

whileループ: `break` と 無限ループ注意

- `while` ループでも `break` で中断できる。
- 条件式が常に `True` になってしまうと **無限ループ** に陥る。
(プログラムが止まらなくなるので注意！ Colab等では停止ボタンで止められる。)

例: 特定の入力があるまで繰り返す

```
total = 0
while total <= 10:
    num_str = input("数値を入力 (合計が10を超えたら終了): ")
    num = int(num_str)
    total += num
    print(f"現在の合計: {total}")
    if total > 10:
        print("合計が10を超えました。")
        break # 条件を満たしたのでループを抜ける

print("プログラム終了")
```

5. ネストした制御構造 (10分)

制御構造の中に、さらに制御構造を入れることができる (入れ子)

- `if` の中に `for` や `while`
- `for` や `while` の中に `if`

例: 九九の表 (一部)

```
for i in range(1, 4): # 1から3の段 (外側のループ)
    print(f"--- {i}の段 ---")
    for j in range(1, 10): # 1から9をかける (内側のループ)
        print(f"{i} x {j} = {i*j}")
    print("") # 段ごとに改行
```

複雑な処理も組み合わせで実現できる！

6. まとめ

- **条件分岐 (if文):** `if`, `elif`, `else` で処理の流れを変える。
- **繰り返し (forループ):** リストや `range()` で決まった回数繰り返す。
- **繰り返し (whileループ):** 条件が `True` の間繰り返す。
- **ループ制御:** `break` (中断), `continue` (スキップ)
- **ネスト構造:** 制御構造の入れ子で複雑な処理を実現。

次回予告

第4回: Python基礎 (3): 関数

次回は、処理をまとめて再利用可能にする「関数」を学びます！

- 関数の定義 (`def`)
 - 引数と戻り値 (`return`)
 - スコープ (変数が見える範囲)
 - デフォルト引数、キーワード引数
- より効率的で読みやすいコードを書くための重要なステップです。

付録: 復習課題 Streamlit 解答例 (1/3)

課題1: かんたん計算機

```
import streamlit as st
st.header("課題1: かんたん計算機")
num1_str = st.text_input("数値1を入力してください")
num2_str = st.text_input("数値2を入力してください")
if num1_str and num2_str:
    num1 = float(num1_str)
    num2 = float(num2_str)
    # 計算
    sum_val = num1 + num2
    diff_val = num1 - num2
    prod_val = num1 * num2
    div_val = num1 / num2 # num2が0だとエラー
    # 結果表示 (文字列連結)
    st.write(str(num1) + " + " + str(num2) + " = " + str(sum_val))
    st.write(str(num1) + " - " + str(num2) + " = " + str(diff_val))
    st.write(str(num1) + " * " + str(num2) + " = " + str(prod_val))
    st.write(str(num1) + " / " + str(num2) + " = " + str(div_val))
```

付録: 復習課題 Streamlit 解答例 (2/3)

課題2: 挨拶メッセージ

```
import streamlit as st

st.header("課題2: 挨拶メッセージ")

name = st.text_input("お名前を入力してください")
age_str = st.text_input("年齢を入力してください")

# 入力があるか確認
if name and age_str:
    # 年齢を整数に変換 (エラーハンドリングなし)
    age = int(age_str)
    # 文字列連結で表示
    st.write("こんにちは、" + name + "さん！あなたは" + str(age) + "歳ですね。")
```

付録: 復習課題 Streamlit 解答例 (3/3)

課題3: BMI計算アプリ

```
import streamlit as st

st.header("課題3: BMI計算アプリ")

height_str = st.text_input("身長(m)を入力してください (例: 1.65)")
weight_str = st.text_input("体重(kg)を入力してください (例: 55.0)")

# 入力があるか確認
if height_str and weight_str:
    # 数値に変換 (エラーハンドリングなし)
    height = float(height_str)
    weight = float(weight_str)

    # BMI計算 (身長が0だとエラーになる)
    bmi = weight / (height ** 2)
    # 結果表示 (文字列連結、roundで丸め)
    st.write("あなたのBMIは " + str(round(bmi, 2)) + " です。")
```