

Python の入門として **文字列 (string)** と **比較演算** をまとめて、シニア向け ICT/AI クラブでも使いやすいように、丁寧に分かりやすく整理してみますね。
ここはつまづきやすいポイントが多いので、例を多めにして理解しやすくしています。

Python入門：文字列 (string) と比較演算

1. 文字列とは？

Python では、文字の集まりを「文字列 (string)」と呼びます。
文字列は "ダブルクォーテーション" または 'シングルクォーテーション' で囲みます。

```
Python ^ [Copy]

name = "Taro"
message = 'こんにちは'
```

2. 文字列の基本操作

● 結合 (+)

```
Python ^ [Copy]

"Hello" + "World"
# → "HelloWorld"
```

● 繰り返し (*)

```
Python ^ [Copy]

"ha" * 3
# → "hahaha"
```

● 長さ (len)

```
Python ^ [Copy]

len("Python")
# → 6
```

3. 文字列の比較演算

Python では、文字列同士を比較できます。
使う演算子は数値と同じです。

演算子	意味
==	等しい
!=	等しくない
<	小さい (辞書順)
>	大きい (辞書順)
<=	以下
>=	以上

4. 文字列比較のポイント (重要)

✓ 比較は「辞書順 (アルファベット順)」で行われる

例：

```
Python ^ [Copy]

"apple" < "banana"
# → True
```

✓ 大文字と小文字は別物

ASCIIコード順で比較されるため、
大文字 < 小文字 という順番になります。

```
Python ^ [Copy]

"Apple" < "apple"
# → True
```

✓ 日本語も比較できる (Unicode順)

```
Python ^ [Copy]

"あ" < "い"
# → True
```

5. 実用例

● パスワードが一致するか確認

```
Python ^ [Copy]

password = "abcd1234"
input_pw = "abcd1234"

if password == input_pw:
    print("ログイン成功")
else:
    print("パスワードが違います")
```

● 名前の並び替え (辞書順)

```
Python ^ [Copy]

names = ["Suzuki", "Tanaka", "Abe"]
sorted(names)
# → ['Abe', 'Suzuki', 'Tanaka']
```

6. よくあるつまづきポイント

✗ "10" < "2" は True になる

理由：文字列として比較するため、
最初の文字 "1" と "2" を比べる。

```
Python ^ [Copy]

"10" < "2"
# → True
```

✓ 数値として比較したい場合

```
Python ^ [Copy]

int("10") < int("2")
# → False
```

7. まとめ

- 文字列は "" または '' で囲む
- 比較は 辞書順 (Unicode順)
- 大文字と小文字は別扱い
- 数値比較したいときは int() に変換

必要なら、

- シニア向けの練習問題
- クラブ用のスライド形式
- ワークシート形式
- 例題の追加
なども作れますよ。

次はどの部分を深めたいですか？

ページを開始しました。Python入門：文字列と比較演算。今すぐに編集することも、後で戻ってきて行うことも可能です。