

# データ型とは？

データ型とは、変数に入れるデータの種類のことです。

この動画では、数値型、文字列型、ブール型について説明します。

ちなみに、Pythonでは、変数にデータを入れるときに、データ型を指定する必要がありません。

Pythonが自動的にデータ型を判断してくれるからです。

このようなプログラミング言語のことを動的型付け言語といいます。

動的型付け言語には、Python、Ruby、Python、PHPなどがあります。

対して、変数にデータを入れる時にデータ型を指定する言語のことを、静的型付け言語といいます。

静的型付け言語には、C言語、Java、Kotlin、Goなどがあります。

## 整数 型

それではまず、最初に、数値型について説明します。

数値型には、整数の「int型」、小数点の「float型」があります。

Pythonのプログラムを書いて、みていきましょう。

```
num01 = 123
num02 = 1.23

print(num01)
print(num02)
```

num01には、整数の123を代入。

num02には、小数点の1.23を代入します。

実行してみましょう。

実行結果：

123

1.23

表示されました。

Pythonのデータ型は、`type`（タイプ）を使用して確認する事が出来ます。

`type`の丸括弧の中に`num01`を入れて、それを`print`でくくりましょう。

```
num01 = 123
```

```
num02 = 1.23
```

```
print(type(num01))
```

```
print(type(num02))
```

実行してみます。

実行結果：

```
int  
float
```

intとfloatが表示されました。

## 文字列型

次に、文字列型について説明します。

文字列型は「string型」ともいいます。

string\_aという変数を作って、そこへ文字列を代入してみま  
しょう。

```
string_a = "Hello,World!"
```

```
print(string_a)
print(type(string_a))
```

文字列はシングルクォーテーション、またはダブルクォーテーションでくくります。

今回はダブルクォーテーションで「Hello,World!」の文字列をくくります。

string\_aの中身を表示させ、あわせて、データの種類を表示させてみましょう。

「Hello,World!」と、データ型のStringが表示されるはずです。

実行してみましょう。

実行結果：

```
Hello,World!
str
```

予想通り「Hello,World!」と「String」が表示されました。

# ブール型

最後に、ブール型です。

ブール型は、Boolean型（ブーリアン型）ともいいます。

ブール型は、TrueまたFalseの2つのうち、どちらか1つを持つ型です。

TrueとFalseとはなんでしょう？

例えば、10と1では10の方が大きいです。

つまり、「 $10 > 1$ 」は正しいです。

これを変数に代入するとTrueが入ります。

逆に「 $10 < 1$ 」は誤りです。これを変数に代入すると、Falseが入ります。

コードを書いてみていきましょう。

```
a = 10
```

```
b = 1
```

```
bool01 = (a > b)
print(bool01)
print(type(bool01))
```

aという変数に10を代入。bという変数に1を代入しましょう。

a>bは正しいです。

結果をbool01に代入してみましょう。

Trueが代入されているはずです。

bool01のデータ型も確認してみましょう。

実行してみます。

実行結果：

```
True
bool
```

「True」と、ブール型である「boolean」が表示されました。

不等号を逆にすると誤りなのでFalseが代入されているはずです。

```
a = 10
b = 1

bool01 = (a < b)
print(bool01)
print(type(bool01))
```

実行してみましょう。

実行結果：



False

*bool*

Falseとブール型の「boolean」が表示されました。