

演算子とは？

演算子とは、足し算、引き算などの四則演算や2つの値の
大小を比較するときに使う記号のことです。

算術演算子

算術演算子からみていきましょう。

算術演算子とは、足し算、引き算、掛け算、割り算などを
するための演算子です。

早速、ソースコードを見ていきましょう。

```
x = 10
```

```
y = 2
```

```
print(x + y)
```

```
print(x - y)
```

```
print(x * y)
```

```
print(x / y)
```

```
print(x % y)
```

xという変数に「10」、yという変数に「2」を代入。

このxとyの変数を使って、演算子の役割をみていきましょう。

足し算と引き算はそのままプラスとマイナスです。

掛け算はアスタリスク(*)です。

割り算は、スラッシュ(/)です。

最後に剰余です。剰余とは、割り算の余りのことです。

剰余は、パーセント(%)です。

足し算の結果は「12」、引き算は「8」、掛け算は「20」、割り算は「5」、剰余は「0」になるはずです。

実行して、表示させてみましょう。

実行結果：

12

8

20

5.0
0

予想通りの結果となりました。

関係演算子

次は関係演算子です。

関係演算子とは、2つの値の関係が正しいか正しくないか判断させる演算子のことです。

例えば、「20」という数字は、「10」より大きいです。これを比較することができます。

関係演算子は、次のレッスンで学ぶ条件分岐のif文などでよく使います。

「 $10 > 0$ 」という意味は、「10」は「0」より大きいという意味です。

この関係演算子は、正しいです。

2つの関係が正しいときには「True」が返ってきます。

実際に、ソースコードを見ていきましょう。

```
x = 10
```

```
y = 2
```

```
print(x > y)
```

xに10、yに2を代入します。

$x > y$ が正しいか試してみます。

実行してみます。

実行結果：

True

Trueが^{ひょうじ}表示されました。

「 $x < y$ 」としたらどうなるでしょう？

つまり、「10」は「2」より^{ちいさい}小さいという^{いみ}意味です。

これは^{あやまり}誤りです。

```
x = 10
```

```
y = 2
```

```
print(x < y)
```

^{ただしく}正しくないときの「False」が^{かえっ}返ってくるはずです。

^{じっこう}実行してみましょう。

じっこうけっか
実行結果：

False

Falseがひょうじ表示されました。

つぎに、とうか等価についてみていきましょう。

とうか等価とは、2つのね値がひとしいということです。

とうか等価はイコールが2つです。

```
y = 2  
z = 10
```

```
print(x <= y)  
print(x >= z)
```

実行結果：
じっこうけっか

False

True

等価ではない場合は、エクスクラメーションマークにイコールです。
とうか

コードで試してみましょう。
ためし

```
x = 10
```

```
y = 2
```

```
print(x == y)
```

```
print(x != y)
```

「10」と「2」は同じではないので、正しくないときの
「Flase」が返ってくるはずです。

等価ではないので、こちらは「True」が返ってくるはずで
す。

実行してみましょう。

実行結果：

False

True

それぞれ「False」と「True」が返ってきました。

論理演算子

次は論理演算子についてみていきましょう。

論理演算子とは、複数の条件を判断させる演算子のことで

す。

日本語でいうと「かつ」とか「または」のことです。英語でいうとandとorです。

and条件はandを記述し、or条件はorを記述します。

「3」と「8」という数字でみていきましょう。

例えば、「8」という数字は、5以上かつ10以下です。

「3」という数字は、5以上かつ10以下ではないです。

コードで試してみましょう。

```
x = 8
```

```
y = 3
```

```
print(x >= 5 and x <= 10)
```

```
print(y >= 5 and y <= 10)
```

まずはand条件についてです。

「8」は5以上かつ10以下なので「True」、 「3」は5以上

かつ10以下ではないので「False」が返ってくるはずです。
実行してみましょう。

実行結果：

```
True
False
```

それぞれ「True」と「False」が返ってきました。

次にor条件についてです。

```
x = 8
```

```
y = 3
```

```
print(x == 3 or y == 3)
```

```
print(x == 1 or y == 1)
```

xは3と等しいか、また、yは3と等しいという条件です。

これはyが3なので、後ろの条件が一致します。

Trueが返ってくるはずです。

xは1と等しいか、yは1と等しいという条件も書いてみましょう。

こちらはどちらの条件にも一致しないので、Falseが返ってくるはずです。

実行してみましょう。

実行結果：

True

False

予想通りの結果が返ってきました。

代入演算子

次に、代入演算子です。

今まで変数に代入するときに使っていた「=」は代入演算子といえます。

また、代入する時に、足し算や引き算を同時にすることができます。

足し算、引き算などと組み合わせて代入する演算子のことを複合代入演算子といえます。

例えば、xという変数に「10」を足してから、xに代入する方法は、「x += 10」です。

例えば、zにyを足してからzに代入する方法は、「z += y」です。

コードで試してみましょう。

```
x = 8
```

```
y = 12
```

```
z = 20
```

```
x += 10
```

```
z += y
```

```
print(x)
```

```
print(z)
```

xに8、yに12、zに20を代入して、複合代入演算子を試してみましよう。

xは18となり、zは32になるはずです。

実行してみましよう。

実行結果：

18

32

予^よ想^そ通^うり^どの^お結^り果^けが^っ返^かっ^てき^かえ^えま^っした。