

4. 標準ライブラリとモジュール

Standard Libraries and Modules

プログラミング・データサイエンス I

2021/5/13

1 今回の目標

今回の目標

- 関数
- 標準的ライブラリ
- モジュール

プログラムに必要なものを、ゼロから全て作るということはありません。多くのプログラムで必要となる機能は、ライブラリとして提供されます。今回は、標準的なライブラリについて説明します。

始めに、サンプルプログラムを取得してください

<https://github.com/first-programming-saga/StandardLibraries>

2 関数 : Functions

関数 : Functions

- 引数を与えると、それに基づく計算を行い、結果を返す
 - 数学関数など
 - 式で表せなくてもよい
- 引数を与えると、そのデータを加工して返す

初めに、「関数」について説明します。皆さんの知っている関数はどんなものがありますか。例えば二次関数 $f(x) = ax^2 + bx + c$ は、三つのパラメタ (a, b, c) が定まっていて、変数 x の値を入れると関数の値が計算するものです。関数に与える変数を引数 (arguments)

と言います。

これを一般的に言うと、数学でいうところの関数は、引数を与えると、それに対応した計算を行って、その値を返すものです。この計算は、先ほどの二次関数のような式で直接にかける必要はありません。例えば、 $x < 0$ ならば 0 を返し、 $x \geq 0$ ならば 1 を返すように、条件に応じて異なる値を返すものでも構いません。

更に、データの塊を与えると、その中を書き換えて返すものも関数になります。まだ、データの塊を説明していないので、これは後でやりましょう。

2.1 Python の関数 : Functions in Python

Python の関数 : Functions in Python

- 組み込み関数
特別な指定なしで利用できる
- モジュールを指定して利用する関数
沢山ある
- 自分で定義した関数

Python の関数は、3 つに分類されます。第一は「組み込み関数」と呼ばれるもので、言語とともに配布され、特別に使うことを指示しなくてもよいものです。プログラミングに限りませんが、関連する部品などを集めたものをモジュール (modules) と言います。第二の関数のグループは、モジュールを指定して利用するものです。第三は、プログラムを書く過程で、自ら定義する関数です。

3 数値計算に使える組み込み関数

Built-in functions for numerical calculations

数値計算に使える組み込み関数 : Built-in functions for numerical calculations

- 何も指定せずに使える数値処理関数
- 基本的な数値処理

数値計算に使える組み込み関数を表 1 に示します。動作例を見るために、`basicFunctions.ipynb` を開けて、実行してみましょう。

最初のセルにある `abs()` は、絶対値、つまり引数が正であればそのまま、負であれば符

関数	説明
<code>abs(x)</code>	x の絶対値 $ x $
<code>divmode(a,b)</code>	a を b で除した商と余りを組 (tuple) にして返す
<code>max(a,b,c,...)</code>	引数の中から、最大値を返す
<code>min(a,b,c,...)</code>	引数の中から、最小値を返す
<code>pow(x,y)</code>	x^y
<code>pow(x,y,z)</code>	$x^y \bmod z$
<code>round(x,k)</code>	x を k 桁に丸める

表 1 数値計算に使える組み込み関数。round は、切り上げと切り下げが等距離である場合には、偶数側になることに注意。

号を正にした値を返します。

4 番目セルにある `max()` は、引数（複数であることに注意）のうちから、最大値を返します。

6 番目の `round()` は、「まるめ」と呼ばれる関数です。通常の上捨五入とな違うことに注意が必要です。

課題 1 最小値を求める例を作成し、実行しなさい。

4 文字列操作関数

Built-in functions for string operations

文字列操作関数 : Built-in functions for string operations

- 何も指定せずに使える文字列関数
- 基本的な文字列処理
- 文字のコード

文字操作の基本的な関数を表 2 に示します。

コンピュータのなかでは、文字は、文字コードと対応が付いています。`chr()` 関数は整数を引数として与えると、それに対応する文字が帰ってきます。`ord()` 関数はその逆です。

`len()` 関数は、これからも何度も出てきます。ここでは、文字列の長さを返しますが、

関数	説明
<code>chr(k)</code>	文字 k に対する unicode
<code>ord(k)</code>	unicode k に対する文字
<code>len(k)</code>	文字列 k の長さ
<code>str(k)</code>	数値 k を文字列化

表 2 文字列操作関数

それ以外にも様々なオブジェクトの長さを図ることができる関数です。

ちょっと物足りませんね。オブジェクトとしての文字が持っている操作 (メソッドと言います) は、後で説明します。

動作例を見るために、`basicStringFunctions.ipynb` を開けて、実行しましょう。

はじめの 4 つは、文字コードと文字の関係を示す例です。ほかのコードや文字に書き換えて実行してみましょう。

5 番目のセルは文字列の長さを求めています。

6 番目から 8 番目は、数字の文字列と数値との型変換です。

課題 2 文字コードを指定することで、アルファベット大文字を表示しなさい

5 モジュール : Modules

モジュール : Modules

- 関連する関数や定数などをまとめたもの
 - python と一緒に配布されているもの
 - 後からインストールするもの
- `import` モジュール
- `from` モジュール `import` 関数
- `as` で別名を付けることも可能

関連する関数や定数などをまとめたものを、プログラミングの世界ではモジュール (modules) と言います。Python のモジュールには、言語と一緒に配布されるものと、初回で説明した `pip` を使って、追加するものがあります。

モジュールを指定することで、そのモジュールに入っている関数や定数などを使うこと

ができます。モジュールの指定の仕方は複数あります。実際の場面に沿って、説明していきます。

5.1 math モジュール

math モジュール

- 基本的な数学関数
 - 三角関数、指数関数、対数関数
 - 切り捨て、切り上げ、平方根

前に出てきた数学関数はものすごく簡単なものでした。それだけでは、様々な計算、特に科学技術計算はできません。こうした関数のうち、基本的なものは `math` という名前のモジュールにまとめられています。

ここには、三角関数、対数関数などのほか、切り捨て・切り上げ、最大公約数、平方根などがあります。

関数	説明
<code>ceil(x)</code>	$\lceil x \rceil$ x の小数以下を切り捨てる
<code>copysign(x,y)</code>	x と絶対値が等しく、 y と符号が等しい値
<code>fabs(x)</code>	$ x $ の float 型
<code>factorial(x)</code>	$x!$
<code>floor(x)</code>	$\lfloor x \rfloor$ x の小数以下を切り上げ
<code>fmod(x,y)</code>	$x \bmod y$ の float 型
<code>fsum(d)</code>	Iterable なデータ列 d の和
<code>gcd(x,y)</code>	x と y の最大公約数
<code>inf</code>	float 型の最大値
<code>nan</code>	float 型の数値でないものを表現

表 3 様々な関数。Iterable とは、リストのように、順序がついたデータの塊のこと。詳しくはデータ構造の回で説明

それでは、`mathFunctions.ipynb` を開いて実行しましょう。今回の例は、理解が難しいかも知れません。細かいことはよいので、何ができるかを見ていきましょう。

最初のセルで `math` というモジュールをインポート (`import`) しています。import す

関数	説明
exp(x)	e^x
log(x, b)	$\log_b x$
log(x)	$\ln x = \log_e x$
log2(x)	$\log_2 x$
log10(x)	$\log_{10} x$
sqrt(x)	\sqrt{x}
e	e
cos(t)	$\cos t$
sin(t)	$\sin t$
tan(t)	$\tan t$
acos(x)	$\arccos x$
asin(x)	$\arcsin x$
atan(x)	$\arctan x$
atan2(x, y)	原点から (x, y) へのベクトルの角度
degree(t)	ラジアンから角度へ
radian(t)	角度からラジアンへ

ると、そのモジュールに属している関数などが使えるようになります。

常用対数とは、数値が 10 の何乗なのかを調べる関数です。 $10^0 = 1$ 、 $10^1 = 10$ 、 $10^2 = 100$ などに対応して、 $\log_{10} 1 = 0$ 、 $\log_{10} 10 = 1$ 、 $\log_{10} 100 = 2$ となります。ここから分かることは、整数の桁数が対数でわかることです。このことを使って、2 番目のセルでは、101 という整数が 3 桁の数であることを割り出しています。

3 番目のセルでは、最大公約数を求めています。

課題 3 小数以下の切り下げ、切り上げの例を作成しなさい。

6 オブジェクトとメソッド : Objects and Methods

オブジェクトとメソッド : Objects and Methods

- python では、データやデータの塊をオブジェクトと呼ぶ
- オブジェクトには、操作方法（メソッド）が付随している

オブジェクトの話は、前回しましたね。python では、全てのデータやデータの塊がオブジェクトです。オブジェクトには、それを操作する関数がついています。操作する「方法」という意味でメソッド (method) と呼びます。普通の関数と違うのは、メソッドでは、オブジェクトが持っているデータの塊を変更することが含まれていることです。

6.1 文字列のメソッド : Methods for strings

文字列のメソッド : Methods for strings

- 大文字小文字変換
- 含まれる文字の数
- 文字列を発見
- 文字列を置き換え
- 余分な文字を取り去る
- 文字列差し込み
- 注意 : immutable であること

ここでは、文字列というオブジェクトに関連したメソッドをいくつか説明しましょう。`stringFunctions.ipynb` を開き、実行しましょう。

始めのほうは、文字列を全て大文字にする `upper()`、小文字にする `lower()` を実行しています。4 番目のセルで、このようなメソッドの実行が、もとの文字列を変更していないことを確認しています。文字列は、変更不能なオブジェクトだからです。

その後の、5 から 7 番目のセルでは、先頭だけを大文字にする `capitalize()`、各単語の先頭を大文字にする `title()` というメソッドを実行しています。

8 から 10 番目のセルでは、部分文字列の取り出しです。元の文字列が変更されているわけではありません。

11 から 15 番目のセルでは、文字列を検索する、置換するというメソッドです。置換の

例では、置換した文字列が帰ってくるので、それを `c` という文字列に入れています。くどいですが、元の文字列が変更されているわけではありません。

課題 4 文字列”Saga University”から二つの文字列”Saga”と”University”を切り出し、それぞれ別の変数に保存しなさい。

7 次回

次回は、条件に応じて処理を選択する方法と処理を繰り返す方法を扱います。教科書では5章「条件分岐、繰り返し、例外処理」の前半、`if` と `while` を扱っている部分です。