

2023.11.22版

Pythonのインストール手順を示します。
スライドはWindows11環境でのスクリーンショットです。ウェブブラウザによって挙動が多少異なりますのでご注意ください。Chrome(推奨)で動作確認しています。

Pythonのインストール (Windows版)

東京大学大学院農学生命科学研究科

アグリバイオインフォマティクス教育研究プログラム

<https://www.iu.a.u-tokyo.ac.jp/>

メール: info@iu.a.u-tokyo.ac.jp

Twitter: @Agribio_utokyo

Contents

- Python本体のインストール
- インストール確認
- Pythonの起動と終了(optional)
- 基本的な利用法(optional)
- パッケージのインストール(optional)
- Biopythonの基本的な利用法(optional)

Pythonのインストール

The screenshot shows the Python.org homepage. A red arrow labeled '1' points to the address bar containing 'https://www.python.org'. Another red arrow labeled '2' points to the 'Downloads' link in the navigation menu. The page features the Python logo, a search bar, and a navigation menu with links for About, Downloads, Documentation, Community, Success Stories, News, and Events. Below the navigation menu, there is a code editor showing Python code for list comprehensions and the enumerate function, and an article titled 'Compound Data Types'.

```
# Python 3: List comprehensions
>>> fruits = ['Banana', 'Apple', 'Lime']
>>> loud_fruits = [fruit.upper() for fruit in
fruits]
>>> print(loud_fruits)
['BANANA', 'APPLE', 'LIME']

# List and the enumerate function
>>> list(enumerate(fruits))
[(0, 'Banana'), (1, 'Apple'), (2, 'Lime')]
```

Compound Data Types

Lists (known as arrays in other languages) are one of the compound data types that Python understands. Lists can be indexed, sliced and manipulated with other built-in functions. [More about lists in Python 3](#)

1 2 3 4 5

Python is a programming language that lets you work quickly and integrate systems more effectively. >>> [Learn More](#)

Pythonのインストール2

①Pythonのサイト。②Downloads。③各自のOSに対応するものを選びたいが、例えばこのWindows11環境では④のように見えるので、④をクリック。Macは標準でインストール済みですが、バージョンが若干古い傾向にあります。気になる方はここから最新版をインストールしてもよいと思います。

The screenshot shows the Python.org website. The navigation bar includes 'Python', 'PSF', 'Docs', and 'PyPI'. The main navigation menu has 'About', 'Downloads', 'Documentation', 'Community', 'Success Stories', 'News', and 'Events'. The 'Downloads' menu is open, showing options: 'All releases', 'Source code', 'Windows', 'macOS', 'Other Platforms', 'License', and 'Alternative Implementations'. A red bracket labeled '3' groups 'Windows', 'macOS', and 'Other Platforms'. A red arrow labeled '4' points to the 'Python 3.12.0' button under the 'Download for Windows' section. The 'Download for Windows' section includes a note: 'Note that Python 3.9+ cannot be used on Windows 7 or earlier.' and a link to 'View the full list of downloads.' The footer text reads: 'Python is a programming language that lets you work quickly and integrate systems more effectively. >>> [Learn More](#)'.

<https://www.python.org/ftp/python/3.12.0/python-3.12.0-amd64.exe>

Pythonのインストール3

⑤exeファイルを任意の場所に、⑥保存。

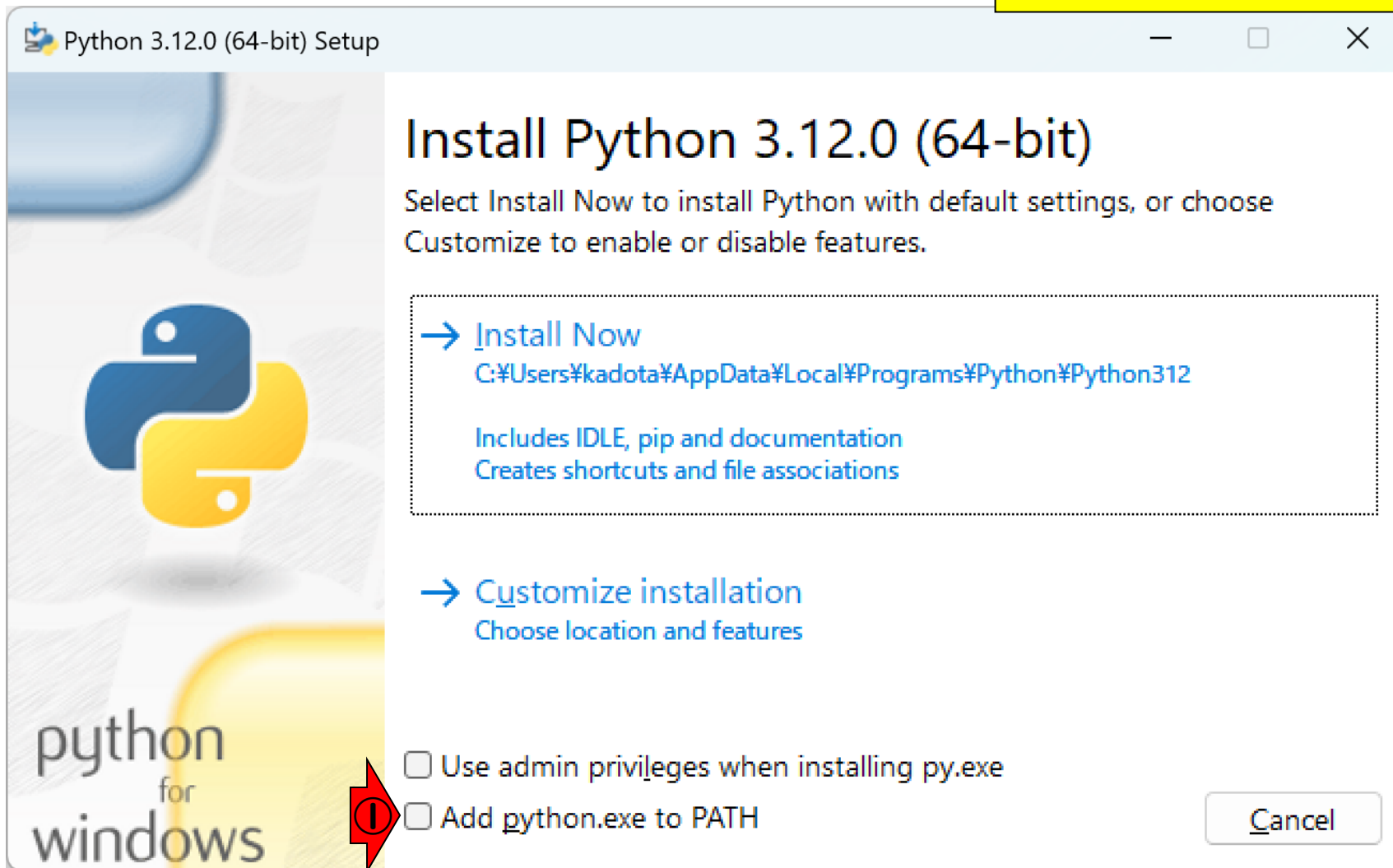
The screenshot shows a web browser window with the URL [python.org](https://www.python.org). The page content includes navigation links for Python, PSF, Docs, PyPI, Jobs, and Community. A file save dialog box is open over the browser, titled "名前を付けて保存". The dialog shows the current location as "デスクトップ". The file name field contains "python-3.12.0-amd64.exe" and the file type is set to "Application (*.exe)". A red arrow labeled "⑤" points to the file name field. Another red arrow labeled "⑥" points to the "保存(S)" button. The "キャンセル" button is also visible. In the background, a terminal window shows Python code execution:

```
# Python
>>> 1 / 2
0.5
>>> 2 **
8
>>> 17 /
5.6666666
>>> 17 //
5
```

At the bottom of the browser window, the address bar shows the URL: <https://www.python.org/ftp/python/3.12.0/python-3.12.0-amd64.exe>

Pythonのインストール4

ダウンロードしたexeファイルをダブルクリックした状態。デフォルトでは①にチェックが入っていないので…



Pythonのインストール5

ダウンロードしたexeファイルをダブルクリックした状態。デフォルトでは①にチェックが入っていないので、①チェックを入れて、②Install Now。

Python 3.12.0 (64-bit) Setup

Install Python 3.12.0 (64-bit)

Select Install Now to install Python with default settings, or choose Customize to enable or disable features.

→ Install Now

C:\Users\kadota\AppData\Local\Programs\Python\Python312

Includes IDLE, pip and documentation
Creates shortcuts and file associations

→ Customize installation

Choose location and features

Use admin privileges when installing py.exe

Add python.exe to PATH

Cancel

python
for
windows

Pythonのインストール6

ダウンロードしたexeファイルをダブルクリックした状態。デフォルトでは①にチェックが入っていないので、①チェックを入れて、②Install Now。インストール進行中…

Python 3.12.0 (64-bit) Setup

Setup Progress



python
for
windows

Installing:

Python 3.12.0 Standard Library (64-bit)




Cancel

Pythonのインストール7

ダウンロードしたexeファイルをダブルクリックした状態。デフォルトでは①にチェックが入っていないので、①チェックを入れて、②Install Now。インストール進行中。③無事完了したら、④Close。

Python 3.12.0 (64-bit) Setup

Setup was successful 

New to Python? Start with the [online tutorial](#) and [documentation](#). At your terminal, type "py" to launch Python, or search for Python in your Start menu.

See [what's new](#) in this release, or find more info about [using Python on Windows](#).



Disable path length limit

Changes your machine configuration to allow programs, including Python, to bypass the 260 character "MAX_PATH" limitation.



Close

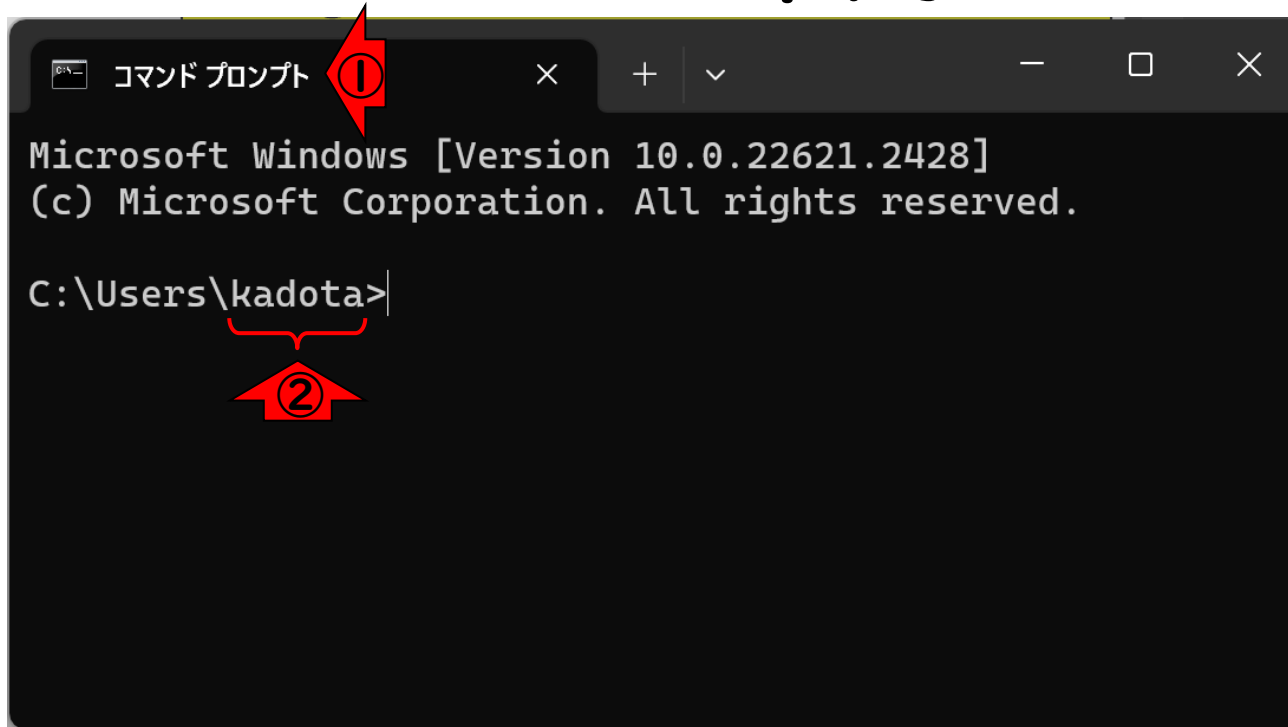
python
for
windows

Contents

- Python本体のインストール
- インストール確認
- Pythonの起動と終了(optional)
- 基本的な利用法(optional)
- パッケージのインストール(optional)
- Biopythonの基本的な利用法(optional)

インストール確認1

①コマンドプロンプトを起動。②の部分はヒトそれぞれです。



```
Microsoft Windows [Version 10.0.22621.2428]
(c) Microsoft Corporation. All rights reserved.

C:\Users\kadota>
```

インストール確認2

③ 「python -V」と打って実行。Vは大文字。④で見えているように、実行結果が「Python 3.12.0」のような感じでバージョン番号（この場合は3.12.0）まで見えていればOK。もし④が「Python」のみならインストールに失敗しています。

```
コマンド プロンプト
Microsoft Windows [Version 10.0.22621.2428]
(c) Microsoft Corporation. All rights reserved.

C:\Users\kadota>python -V
Python 3.12.0

C:\Users\kadota>
```

インストール確認3

③ 「python -V」と打って実行。Vは大文字。④で見えているように、実行結果が「Python 3.12.0」のような感じでバージョン番号（この場合は3.12.0）まで見えていればOK。もし④が「Python」のみならインストールに失敗しています。インストールに成功していれば、⑤「python -h」と打つとこのように見えます。インストールに失敗していれば、「Python」のみが表示されます。

コマンド プロンプト


×

+

▼

```
Microsoft Windows [Version 10.0.22621.2428]
(c) Microsoft Corporation. All rights reserved.
```

```
C:\Users\kadota>python -V
Python 3.12.0
```

```
C:\Users\kadota>python -h 
usage: python [option] ... [-c cmd | -m mod | file | ...]
Options (and corresponding environment variables):
-b          : issue warnings about str(bytes_instance), str(bytearray_instance)
              and comparing bytes/bytearray with str. (-bb: issue errors)
-B          : don't write .pyc files on import; also PYTHONDONTWRITEBYTECODE=x
-c cmd     : program passed in as string (terminates option list)
-d          : turn on parser debugging output (for experts only, only works on
              debug builds); also PYTHONDEBUG=x
-E          : ignore PYTHON* environment variables (such as PYTHONPATH)
-h          : print this help message and exit (also -? or --help)
-i          : inspect interactively after running script; forces a prompt even
              if stdin does not appear to be a terminal; also PYTHONINSPECT=x
-I          : isolate Python from the user's environment (implies -E and -s)
-m mod     : run library module as a script (terminates option list)
```

インストール確認4

インストールに失敗している場合の実際の表示結果。①「python -V」の場合も、②「python -h」の場合も、「Python」のみが表示されます。

```
コマンド プロンプト
C:\Users\kadota>
```

```
コマンド プロンプト
C:\Users\kadota>python -V
Python
C:\Users\kadota>
```

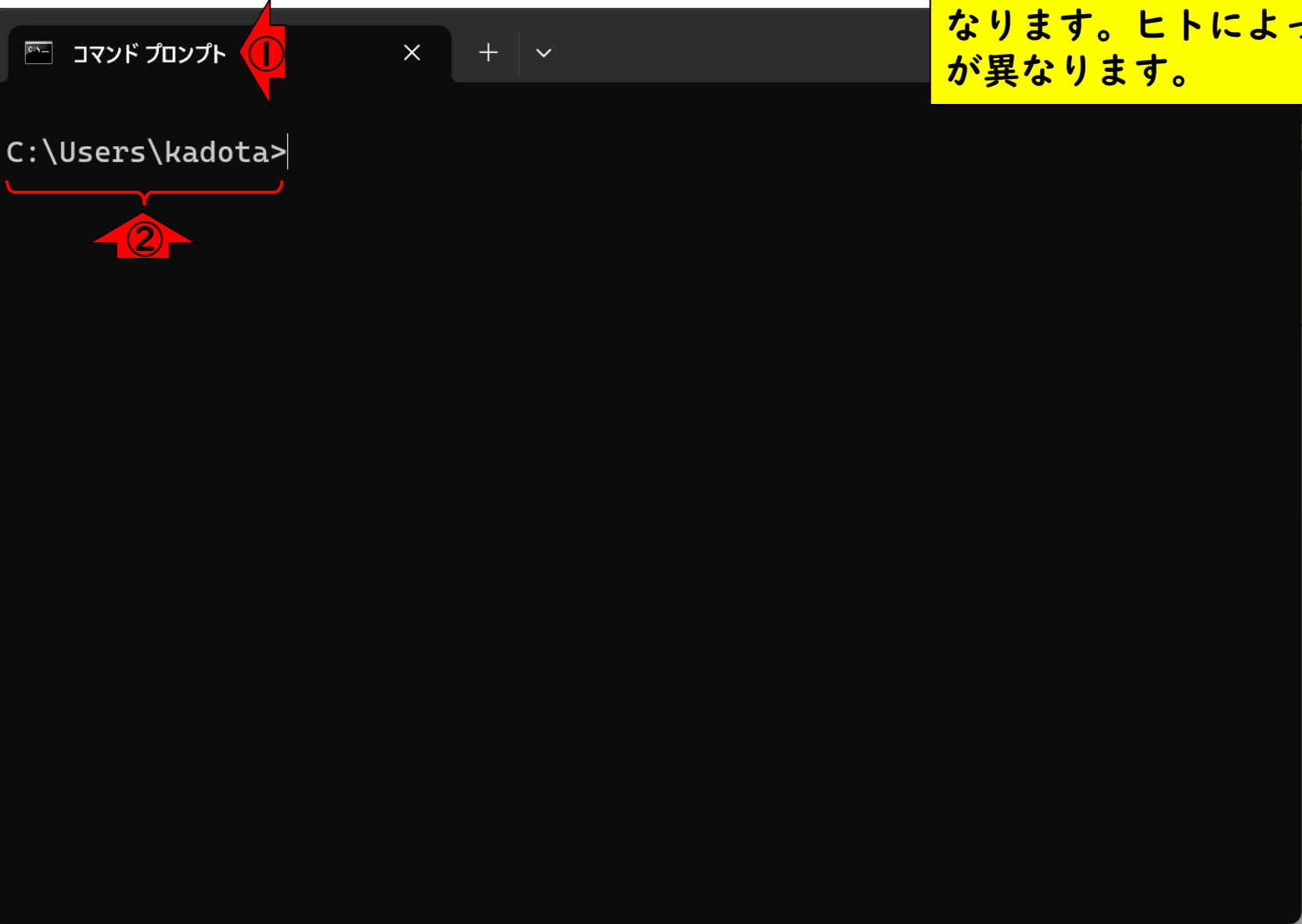
```
コマンド プロンプト
C:\Users\kadota>python -V
Python
C:\Users\kadota>python -h
Python
C:\Users\kadota>
```

Contents

- Python本体のインストール
- インストール確認
- Pythonの起動と終了(optional)
- 基本的な利用法(optional)
- パッケージのインストール(optional)
- Biopythonの基本的な利用法(optional)

Pythonの起動と終了！

①コマンドプロンプトを利用してPythonを起動するやり方を示します。
②は、最初から表示されている部分になります。ヒトによって、若干見え方が異なります。



Pythonの起動と終了2

①コマンドプロンプトを利用してPythonを起動するやり方を示します。
②は、最初から表示されている部分になります。ヒトによって、若干見え方が異なります。起動は、③「python」と打ってリターン。

コマンド プロンプト



×

+

▼

C:\Users\kadota>python|



Pythonの起動と終了3

こんな感じになればOK。Pythonの場合は、④「>>>」のように「大なりマーク(>)」が3つ連続で見えていれば、Pythonのコマンドを実行できます。

```
コマンド プロンプト - python × + ▾  
C:\Users\kadota>python  
Python 3.12.0 (tags/v3.12.0:0fb18b0, Oct 2 2023, 13:03:39) [MSC v.1935  
64 bit (AMD64)] on win32  
Type "help", "copyright", "credits" or "license" for more information.  
>>> |
```



Pythonの起動と終了4

こんな感じになればOK。Pythonの場合は、④「>>>」のように「大なりマーク(>)」が3つ連続で見えていれば、Pythonのコマンドを実行できます。Pythonの終了は、⑤「quit()」と打ちこんでリターン。⑥のようにコマンドプロンプト起動直後のようなものが見えていればOKです。

```
コマンド プロンプト
C:\Users\kadota>python
Python 3.12.0 (tags/v3.12.0:0fb18b0, Oct  2 2023, 13:
64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> quit()
C:\Users\kadota>
```

⑤

⑥

Contents

- Python本体のインストール
- インストール確認
- Pythonの起動と終了(optional)
- **基本的な利用法(optional)**
- パッケージのインストール(optional)
- Biopythonの基本的な利用法(optional)

基本的な利用法I

①コマンドプロンプト起動後の状態。
②は、最初から表示されている部分になります。ヒトによって、若干見え方が異なります。

コマンド プロンプト



C:\Users\kadota>



基本的な利用法2

①コマンドプロンプト起動後の状態。
②は、最初から表示されている部分になります。ヒトによって、若干見え方が異なります。コマンドプロンプト上では、③「| + |」を実行しても2という結果が得られないことを確認しているだけです。

コマンド プロンプト

×

+

▽

C:\Users\kadota>1 + 1

③

'1' は、内部コマンドまたは外部コマンド、操作可能なプログラムまたはバッチ ファイルとして認識されていません。

C:\Users\kadota>

基本的な利用法3

①コマンドプロンプト起動後の状態。
②は、最初から表示されている部分になります。ヒトによって、若干見え方が異なります。コマンドプロンプト上では、③「1 + 1」を実行しても2という結果が得られないことを確認しているだけです。④「python」と打って、Pythonを起動した状態。

コマンド プロンプト - python

×

+

▼

```
C:\Users\kadota>1 + 1
```

'1' は、内部コマンドまたは外部コマンド、
操作可能なプログラムまたはバッチ ファイルとして認識さ

```
C:\Users\kadota>python
```



```
Python 3.12.0 (tags/v3.12.0:0fb18b0, Oct 2 2023, 13:03:39) [MSC v.1935  
64 bit (AMD64)] on win32
```

```
Type "help", "copyright", "credits" or "license" for more information.
```

```
>>> |
```

基本的な利用法4

① コマンドプロンプト起動後の状態。
② は、最初から表示されている部分になります。ヒトによって、若干見え方が異なります。コマンドプロンプト上では、③ 「1 + 1」を実行しても2という結果が得られないことを確認しているだけです。④ 「python」と打って、Pythonを起動した状態。⑤ 「1 + 1」を再度実行すると、正しく2という答えが表示されていることがわかります。同じコマンドプロンプトでも、Python起動の有無がよくわかる例といえます。

```
コマンド プロンプト - python × + ∨  
C:\Users\kadota>1 + 1  
'1' は、内部コマンドまたは外部コマンド、  
操作可能なプログラムまたはバッチ ファイルとして認識さ  
C:\Users\kadota>python  
Python 3.12.0 (tags/v3.12.0:0fb18b0, Oct  2 2023, 13:  
64 bit (AMD64)] on win32  
Type "help" "copyright", "credits" or "license" for  
>>> 1 + 1 ⑤  
2  
>>> |
```


基本的な利用法5

```
コマンド プロンプト - python × + ∨  
C:\Users\kadota>1 + 1  
'1' は、内部コマンドまたは外部コマンド、  
操作可能なプログラムまたはバッチ ファイルとして認識さ  
C:\Users\kadota>python  
Python 3.12.0 (tags/v3.12.0:0fb18b0, Oct  2 2023, 13:  
64 bit (AMD64)] on win32  
Type "help", "copyright", "credits" or "license" for  
>>> 1 + 1  
2  
>>> aa = 5  
>>> aa  
5  
>>> |
```



① コマンドプロンプト起動後の状態。
② は、最初から表示されている部分になります。ヒトによって、若干見え方が異なります。コマンドプロンプト上では、③ 「1 + 1」を実行しても2という結果が得られないことを確認しているだけです。④ 「python」と打って、Pythonを起動した状態。⑤ 「1 + 1」を再度実行すると、正しく2という答えが表示されていることがわかります。同じコマンドプロンプトでも、Python起動の有無がよくわかる例といえます。⑥ 「aa = 5」のように打ち込んで、任意の変数(この場合はaa)に値を代入して利用することもできます。もちろん、次の行で「aa」と打ち込むと、その中身である5が正しく表示されます。

基本的な利用法6

```
コマンド プロンプト
C:\Users\kadota>1 + 1
'1' は、内部コマンドまたは外部コマンド、
操作可能なプログラムまたはバッチ ファイルとして認識さ
C:\Users\kadota>python
Python 3.12.0 (tags/v3.12.0:0fb18b0, Oct  2 2023, 13:
64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for
>>> 1 + 1
2
>>> aa = 5
>>> aa
5
>>> quit()
C:\Users\kadota>
```



① コマンドプロンプト起動後の状態。
② は、最初から表示されている部分になります。ヒトによって、若干見え方が異なります。コマンドプロンプト上では、③ 「1 + 1」を実行しても2という結果が得られないことを確認しているだけです。④ 「python」と打って、Pythonを起動した状態。⑤ 「1 + 1」を再度実行すると、正しく2という答えが表示されていることがわかります。同じコマンドプロンプトでも、Python起動の有無がよくわかる例といえます。⑥ 「aa = 5」のように打ち込んで、任意の変数(この場合はaa)に値を代入して利用することもできます。もちろん、次の行で「aa」と打ち込むと、その中身である5が正しく表示されます。⑦ 「quit()」でPythonを終了。

Contents

- Python本体のインストール
- インストール確認
- Pythonの起動と終了(optional)
- 基本的な利用法(optional)
- パッケージのインストール(optional)
- Biopythonの基本的な利用法(optional)

パッケージのインストール

Rとパッケージの関係同様、Pythonも様々なパッケージをインストールして利用します。ここでは、生命科学分野で有名な①Biopythonのインストールを行います。②Downloadをクリック。

Biopython · Biopython

biopython.org



Python Tools for
Computational
Molecular Biology

Documentation

Download

Mailing lists

News

Biopython Contributors

Scriptcentral

Source Code

GitHub project

Biopython version 1.81

© 2023. All rights

[Edit this page on GitHub](#)

Biopython



See also our [News feed](#) and [Twitter](#).

Introduction

Biopython is a set of freely available tools for biological computation written in [Python](#) by an international team of developers.

It is a distributed collaborative effort to develop Python libraries and applications which address the needs of current and future work in bioinformatics. The source code is made available under the [Biopython License](#), which is extremely liberal and compatible with almost every license in the world.

We are a member project of the [Open Bioinformatics Foundation \(OBF\)](#), who take care of our domain name and hosting for our mailing list etc. The OBF used to host our development repository, issue tracker and website but these are now on [GitHub](#).

This page will help you download and install Biopython, and start using the libraries and tools.

パッケージのインストール2

Rとパッケージの関係同様、Pythonも様々なパッケージをインストールして利用します。ここでは、生命科学分野で有名な①Biopythonのインストールを行います。②Downloadをクリック。移動後の状態。③Installation Instructionが見えるように、半ページ分ほど下に移動。

Download · Biopython

biopython.org/wiki/Download



Python Tools for
Computational
Molecular Biology

Documentation

Download

Mailing lists

News

Biopython Contributors

Scriptcentral

Source Code

GitHub project

Biopython version 1.81

© 2023. All rights

Download

Current Release - 1.81 - 12 February 2023

See also [What's new](#).

Files

Biopython 1.81

- [biopython-1.81.tar.gz](#) 17Mb – Source Tarball
- [biopython-1.81.zip](#) 19Mb – Source Zip File
- [Pre-compiled wheel files on PyPI](#)
- [Tutorial-1.81.pdf](#) – Documentation



Installation Instructions

All supported versions of Python include the Python package management tool `pip`, which allows an easy installation from the command line on all platforms. Try:

パッケージのインストール3

こんな感じ。インストールの基本形は、
コマンドプロンプトで④を実行する
だけです。つまり...

Download · Biopython

biopython.org/wiki/Download



Python Tools for
Computational
Molecular Biology

Documentation

Download

Mailing lists

News

Biopython Contributors

Scriptcentral

Source Code

GitHub project

Biopython version 1.81

© 2023. All rights

③

Installation Instructions

All supported versions of Python include the Python package management tool `pip`, which allows an easy installation from the command line on all platforms. Try:

```
pip install biopython
```

④

For updating an older version of Biopython try:

```
pip install biopython --upgrade
```

This will remove older versions of Biopython and NumPy before it installs the recent versions.

Should you wish to uninstall Biopython:

```
pip uninstall biopython
```

If pip is not already installed you may need to update your Python, but first try:

```
python -m ensurepip
```

パッケージのインストール4

こんな感じ。インストールの基本形は、
コマンドプロンプトで④を実行する
だけです。つまり、④のコマンドをコ
ピーして…

Download · Biopython

biopython.org/wiki/Download



Python Tools for
Computational
Molecular Biology

Documentation

Download

Mailing lists

News

Biopython Contributors

Scriptcentral

Source Code

GitHub project

Biopython version 1.81

© 2023. All rights

Installation Instructions

All supported versions of Python include the Python package management tool `pip`, which allows an easy installation from the command line on all platforms. Try:

```
pip install biopython
```



For updating an older version of Biopython try:

```
pip install biopython --upgrade
```

This will remove older versions of Biopython and NumPy before it installs the recent versions.

Should you wish to uninstall Biopython:

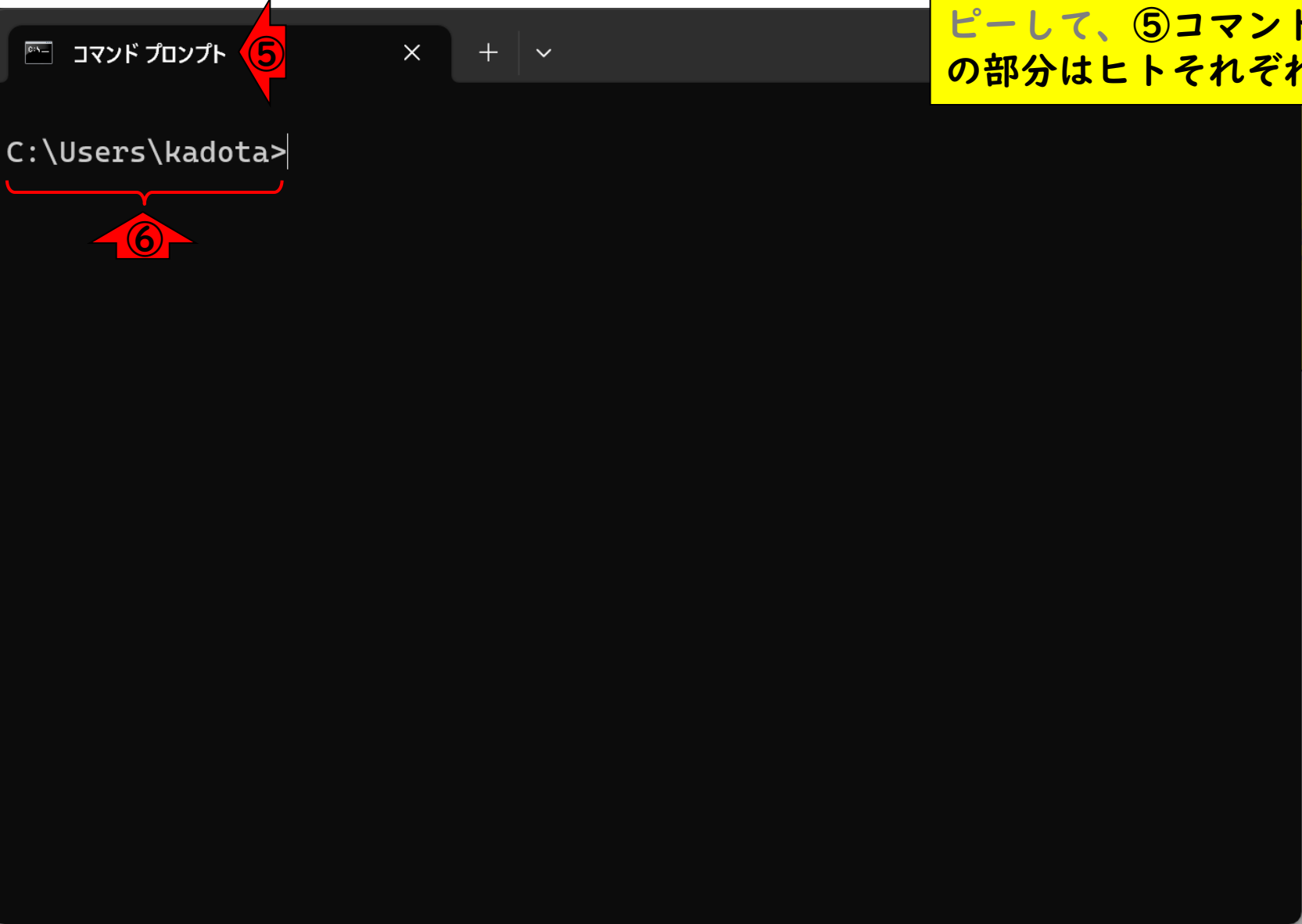
```
pip uninstall biopython
```

If pip is not already installed you may need to update your Python, but first try:

```
python -m ensurepip
```

パッケージのインストール5

こんな感じ。インストールの基本形は、コマンドプロンプトで④を実行するだけです。つまり、④のコマンドをコピーして、⑤コマンドプロンプト(⑥の部分はヒトそれぞれ)上で…



パッケージのインストール6

こんな感じ。インストールの基本形は、コマンドプロンプトで④を実行するだけです。つまり、④のコマンドをコピーして、⑤コマンドプロンプト(⑥の部分はヒトそれぞれ)上で、⑦ペーストし、リターン。

```
コマンド プロンプト × + ▾  
C:\Users\kadota>pip install biopython|
```



パッケージのインストール7

こんな感じ。インストールの基本形は、コマンドプロンプトで④を実行するだけです。つまり、④のコマンドをコピーして、⑤コマンドプロンプト(⑥の部分はヒトそれぞれ)上で、⑦ペーストし、リターン。インストールスタート。⑧biopythonだけでなく、numpyもインストールされます。

コマンド プロンプト - pip install bi

```
C:\Users\kadota>pip install biopython
Collecting biopython
  Obtaining dependency information for biopython from https://www.pyth
honhosted.org/packages/53/dc/95a8a6021d3b49f3ab977486b20e8e588e887c9e01
6a2174a1c026c4ae7f/biopython-1.81-cp312-cp312-win_amd64.whl.metadata
  Downloading biopython-1.81-cp312-cp312-win_amd64.whl.metadata (13 kB)
Collecting numpy (from biopython)
  Obtaining dependency information for numpy from https://files.pytho
osted.org/packages/28/75/3b679b41713bb60e2e8f6e2f87be72c971c9e718b1c17b
8f8749240ddca8/numpy-1.26.2-cp312-cp312-win_amd64.whl.metadata
  Downloading numpy-1.26.2-cp312-cp312-win_amd64.whl.metadata (61 kB)
  ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 61.2/61.2 kB 1.1 MB/s eta 0:00:00
Downloading biopython-1.81-cp312-cp312-win_amd64.whl (2.7 MB)
  ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 2.7/2.7 MB 11.6 MB/s eta 0:00:00
Downloading numpy-1.26.2-cp312-cp312-win_amd64.whl (15.5 MB)
  ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 15.5/15.5 MB 21.8 MB/s eta 0:00:00
Installing collected packages: numpy, biopython
```

⑧

パッケージのインストール8

インストール終了後の状態。①の行の Successfully installed という記載内容からも、無事インストール成功していることがわかります。②コマンドプロンプトが入力待ち状態になっているので、一旦終了してもOKです。

```
コマンド プロンプト
Collecting biopython
  Obtaining dependency information for biopython from https://files.pyth
honhosted.org/packages/53/dc/95a8a6021d3b49f3ab977486b20e8e588e887c9e01
6a2174a1c026c4ae7f/biopython-1.81-cp312-cp312-win_amd64.whl.metadata
  Downloading biopython-1.81-cp312-cp312-win_amd64.whl.metadata (13 kB)
Collecting numpy (from biopython)
  Obtaining dependency information for numpy from https://files.pytho
nhosted.org/packages/28/75/3b679b41713bb60e2e8f6e2f87be72c971c9e718b1c17b
8f8749240ddca8/numpy-1.26.2-cp312-cp312-win_amd64.whl.metadata
  Downloading numpy-1.26.2-cp312-cp312-win_amd64.whl.metadata (61 kB)
  _____ 61.2/61.2 kB 1.1 MB/s eta 0:00:00
Downloading biopython-1.81-cp312-cp312-win_amd64.whl (2.7 MB)
  _____ 2.7/2.7 MB 11.6 MB/s eta 0:00:00
Downloading numpy-1.26.2-cp312-cp312-win_amd64.whl (15.5 MB)
  _____ 15.5/15.5 MB 21.8 MB/s eta 0:00:00
Installing collected packages: numpy, biopython
Successfully installed biopython-1.81 numpy-1.26.2
[notice] A new release of pip is available: 23.2.1 -> 23.3.1
[notice] To update, run: python.exe -m pip install --upgrade pip
C:\Users\kadota>
```



Contents

- Python本体のインストール
- インストール確認
- Pythonの起動と終了(optional)
- 基本的な利用法(optional)
- パッケージのインストール(optional)
- Biopythonの基本的な利用法(optional)

Biopythonの基本的な利用法 I

①Biopythonの基本的な利用法を紹介します。②Documentationをクリック。

The screenshot shows the Biopython website homepage in a browser. The browser's address bar shows 'biopython.org'. The page features a dark sidebar on the left with the Biopython logo and navigation links. The main content area has a 'Biopython' heading, a 'News feed and Twitter' link, and an 'Introduction' section. A red lightning bolt icon with a circled '1' points to the 'Biopython' heading, and another red lightning bolt icon with a circled '2' points to the 'Documentation' link in the sidebar.

Python Tools for Computational Molecular Biology

Documentation ②

Download

Mailing lists

News

Biopython Contributors

Scriptcentral

Source Code

GitHub project

Biopython version 1.81
© 2023. All rights

[Edit this page on GitHub](#)

Biopython ①

See also our [News feed](#) and [Twitter](#).

Introduction

Biopython is a set of freely available tools for biological computation written in [Python](#) by an international team of developers.

It is a distributed collaborative effort to develop Python libraries and applications which address the needs of current and future work in bioinformatics. The source code is made available under the [Biopython License](#), which is extremely liberal and compatible with almost every license in the world.

We are a member project of the [Open Bioinformatics Foundation \(OBF\)](#), who take care of our domain name and hosting for our mailing list etc. The OBF used to host our development repository, issue tracker and website but these are now on [GitHub](#).

This page will help you download and install Biopython, and start using the libraries and tools.

Biopythonの基本的な利用法2

①Biopythonの基本的な利用法を紹介します。②Documentationをクリック。ググれば様々な情報が得られますが、例えば文字列を塩基配列やアミノ酸配列(Sequence)として取り扱ってくれる③Seq関数の利用例を見てみます。③をクリック。

Documentation · Biopython
biopython.org/wiki/Documentation



Python Tools for
Computational
Molecular Biology

Documentation

Download

Mailing lists

News

Biopython Contributors

Scriptcentral

Source Code

GitHub project

Biopython version 1.81

© 2023. All rights

Documentation

New to Biopython? Check out the [Getting Started](#) page, or follow one of the links below.

- The **Biopython Tutorial and Cookbook** contains the bulk of Biopython documentation. It provides information to get you started with Biopython, in addition to specific documentation on a number of modules

[HTML](#), [PDF](#)

- API documentation for Biopython modules is generated directly from source code comments Sphinx autodoc:

[Biopython API \(latest release\)](#), and [Biopython API \(in-development\)](#).

- [Wiki documentation](#)

- ③ [Seq](#) and [SeqRecord](#) objects
 - [Bio.SeqIO](#) - sequence input/output
 - [Bio.AlignIO](#) - alignment input/output
 - [Bio.PopGen](#) - population genetics
 - [Bio.PDB](#) - structural bioinformatics

Biopythonの基本的な利用法3

④ 「Biopythonでは配列を`Seq`オブジェクトとして取り扱う」的なことが書かれています。⑤が見えるように、半ページ分ほど下に移動。

Handling sequences with the Seq

biopython.org/wiki/Seq



Python Tools for
Computational
Molecular Biology

Documentation

Download

Mailing lists

News

Biopython Contributors

Scriptcentral

Source Code

GitHub project

Biopython version 1.81

© 2023. All rights

[Edit this page on GitHub](#)

- [Wiki Documentation](#)

Handling sequences with the Seq class.

In Biopython, sequences are usually held as `Seq` objects, which add various biological methods on top of string like behaviour.



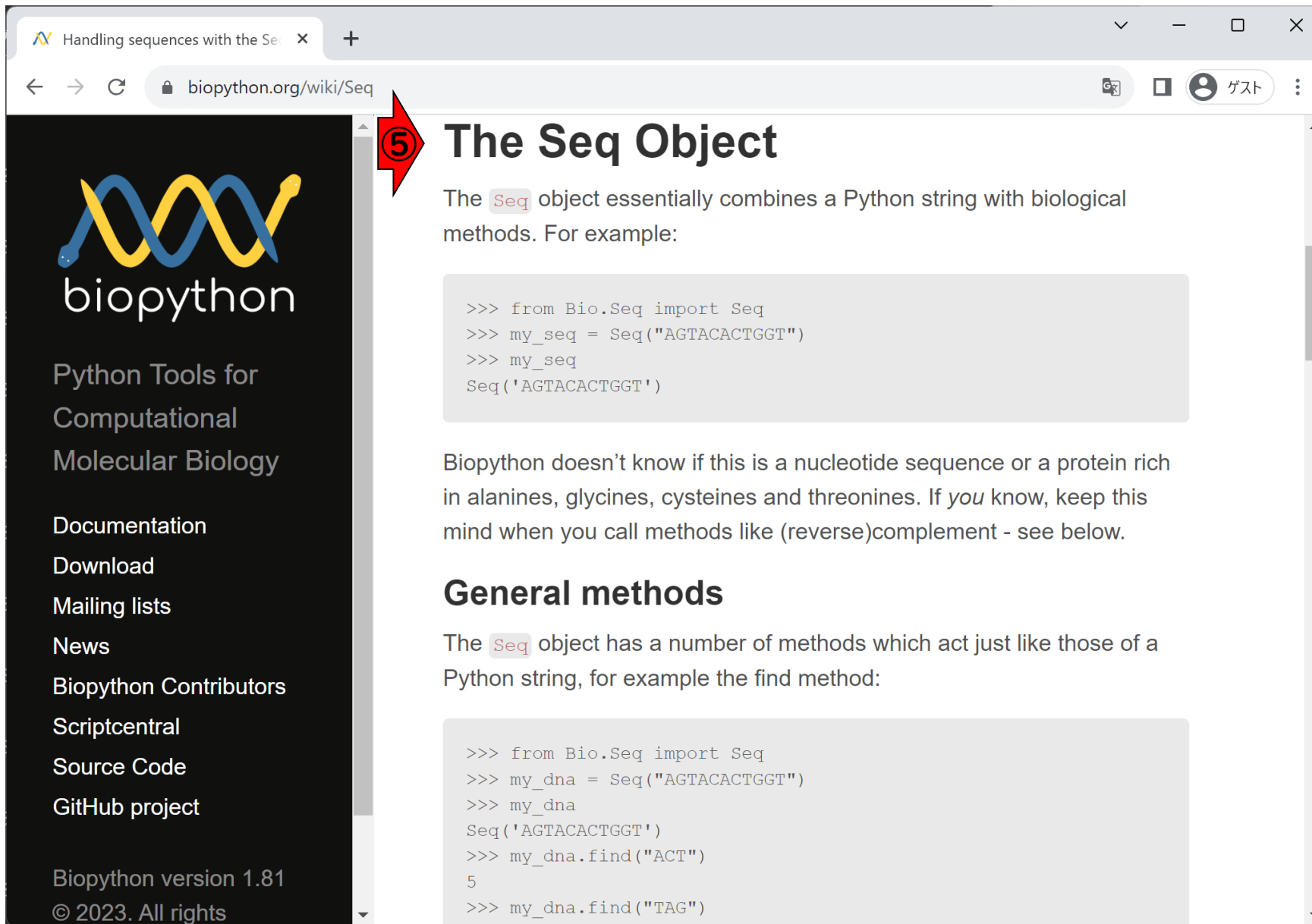
This page describes the Biopython `Seq` object, defined in the `Bio.Seq` module (together with related objects like the `MutableSeq`, plus some general purpose sequence functions). In addition to this wiki page, there is a whole chapter in the [Tutorial \(PDF\)](#) on the `Seq` object - plus its [API documentation](#) (which you can read online, or from within Python with the help command).

If you need to store additional information like a sequence identifier or name, or even more details like a description or annotation, then we use a `SeqRecord` object instead. These are the sequence records used by the `SeqIO` module for reading and writing sequence files.



The Seq Object

Biopythonの基本的な利用法4



Handling sequences with the Seq

biopython.org/wiki/Seq

The Seq Object

The `Seq` object essentially combines a Python string with biological methods. For example:

```
>>> from Bio.Seq import Seq
>>> my_seq = Seq("AGTACACTGGT")
>>> my_seq
Seq('AGTACACTGGT')
```

Biopython doesn't know if this is a nucleotide sequence or a protein rich in alanines, glycines, cysteines and threonines. If *you* know, keep this mind when you call methods like (reverse)complement - see below.

General methods

The `Seq` object has a number of methods which act just like those of a Python string, for example the find method:

```
>>> from Bio.Seq import Seq
>>> my_dna = Seq("AGTACACTGGT")
>>> my_dna
Seq('AGTACACTGGT')
>>> my_dna.find("ACT")
5
>>> my_dna.find("TAG")
```

biopython

Python Tools for
Computational
Molecular Biology

Documentation
Download
Mailing lists
News
Biopython Contributors
Scriptcentral
Source Code
GitHub project

Biopython version 1.81
© 2023. All rights

Biopythonの基本的な利用法5

こんな感じ。⑥のあたりを眺めて、「⑥の3行は、コマンドプロンプトでpythonと打ち込んでPython環境にしてから打ち込むコマンド。そして最初の>>>は実際には打ち込まない。」と解釈します。

Handling sequences with the Seq × +
← → ↻ biopython.org/wiki/Seq



Python Tools for
Computational
Molecular Biology

Documentation

Download

Mailing lists

News

Biopython Contributors

Scriptcentral

Source Code

GitHub project

Biopython version 1.81

© 2023. All rights

The Seq Object

The `Seq` object essentially combines a Python string with biological methods. For example:



```
>>> from Bio.Seq import Seq
>>> my_seq = Seq("AGTACACTGGT")
>>> my_seq
Seq('AGTACACTGGT')
```

Biopython doesn't know if this is a nucleotide sequence or a protein rich in alanines, glycines, cysteines and threonines. If *you* know, keep this mind when you call methods like `(reverse)complement` - see below.

General methods

The `Seq` object has a number of methods which act just like those of a Python string, for example the `find` method:

```
>>> from Bio.Seq import Seq
>>> my_dna = Seq("AGTACACTGGT")
>>> my_dna
Seq('AGTACACTGGT')
>>> my_dna.find("ACT")
5
>>> my_dna.find("TAG")
```

Biopythonの基本的な利用法6

まずは、BiopythonのSeq関数を利用できるようにロードすべく、①反転部分をコピー。

Handling sequences with the Seq

biopython.org/wiki/Seq



Python Tools for
Computational
Molecular Biology

Documentation

Download

Mailing lists

News

Biopython Contributors

Scriptcentral

Source Code

GitHub project

Biopython version 1.81

© 2023. All rights

The Seq Object

The `seq` object essentially combines a Python string with biological methods. For example:

```
>>> from Bio.Seq import Seq
>>> my_seq = Seq("AGTACACTGGT")
>>> my_seq
Seq('AGTACACTGGT')
```

Biopython doesn't know if this is a nucleotide sequence or a protein rich in alanines, glycines, cysteines and threonines. If *you* know, keep this mind when you call methods like (reverse)complement - see below.

General methods

The `seq` object has a number of methods which act just like those of a Python string, for example the find method:

```
>>> from Bio.Seq import Seq
>>> my_dna = Seq("AGTACACTGGT")
>>> my_dna
Seq('AGTACACTGGT')
>>> my_dna.find("ACT")
5
>>> my_dna.find("TAG")
```

Biopythonの基本的な利用法7

まずは、BiopythonのSeq関数を利用できるようにロードすべく、①反転部分をコピー。コマンドプロンプトで②pythonと打ち込んでPython環境(つまり>>>が見える状態)にした状態で…

コマンド プロンプト - python

②

```
C:\Users\kadota>python
Python 3.12.0 (tags/v3.12.0:0fb18b0, Oct  2 2023, 13:03:39) [MSC v.1935 6
4 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> |
```

Biopythonの基本的な利用法8

まずは、BiopythonのSeq関数を利用できるようにロードすべく、①反転部分をコピー。コマンドプロンプトで②pythonと打ち込んでPython環境(つまり>>>が見える状態)にした状態で、③ペーストしてリターン。特にエラーが出ていないことから、BiopythonのSeq関数が利用できるようになりました。

```
コマンド プロンプト - python
C:\Users\kadota>python
Python 3.12.0 (tags/v3.12.0:0fb18b0, Oct  2 2023, 1
4 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> from Bio.Seq import Seq ③
>>> |
```

Biopythonの基本的な利用法9

①は、AGTACACTGGTという文字列を、Seq関数で読み込むことで、`Seq`オブジェクトにした状態でmy_seqに格納せよというコマンドです。②は、この段階ではまだBiopythonは塩基配列なのかアミノ酸配列なのかはわからない的なことを書いています。直感的にはAGTACACTGGTは塩基配列ですが、アミノ酸配列の可能性もあるからです(Aはアラニン、Cはシステイン、Gはグリシン、Tはスレオニン)。

Handling sequences with the Seq × +
biopython.org/wiki/Seq



Python Tools for
Computational
Molecular Biology

Documentation

Download

Mailing lists

News

Biopython Contributors

Scriptcentral

Source Code

GitHub project

Biopython version 1.81

© 2023. All rights

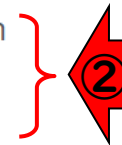
The Seq Object

The `Seq` object essentially combines a Python string with methods. For example:

```
>>> from Bio.Seq import Seq
>>> my_seq = Seq("AGTACACTGGT")
>>> my_seq
Seq('AGTACACTGGT')
```



Biopython doesn't know if this is a nucleotide sequence or a protein rich in alanines, glycines, cysteines and threonines. If *you* know, keep this mind when you call methods like (reverse)complement - see below.



General methods

The `Seq` object has a number of methods which act just like those of a Python string, for example the find method:

```
>>> from Bio.Seq import Seq
>>> my_dna = Seq("AGTACACTGGT")
>>> my_dna
Seq('AGTACACTGGT')
>>> my_dna.find("ACT")
5
>>> my_dna.find("TAG")
```

Biopythonの基本的な…10

```
コマンド プロンプト - python
C:\Users\kadota>python
Python 3.12.0 (tags/v3.12.0:0fb18b0, Oct  2 2023, 1
4 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" fo
>>> from Bio.Seq import Seq
>>> my_seq = Seq("AGTACACTGGT")
>>> my_seq
Seq('AGTACACTGGT')
>>> |
```

①は、AGTACACTGGTという文字列を、Seq関数で読み込むことで、`Seq`オブジェクトにした状態でmy_seqに格納せよというコマンドです。②は、この段階ではまだBiopythonは塩基配列なのかアミノ酸配列なのかはわからない的なことを書いています。直感的にはAGTACACTGGTは塩基配列ですが、アミノ酸配列の可能性もあるからです(Aはアラニン、Cはシステイン、Gはグリシン、Tはスレオニン)。さきほどの①とそのすぐ下のコマンドを実行した結果。赤枠内が実際にコピペ実行したものになります。

Biopythonの基本的な…II

ここまでで、①灰色部分全体と、②実際に打ち込んだ赤枠内との関係がわかりました。このノリで、③以降を解説を見ながら順を追って理解していくとよいと思います。

Handling sequences with the Seq × +
← → ↻ biopython.org/wiki/Seq



Python Tools for
Computational
Molecular Biology

Documentation

Download

Mailing lists

News

Biopython Contributors

Scriptcentral

Source Code

GitHub project

Biopython version 1.81

© 2023. All rights

The Seq Object

The `Seq` object essentially combines a Python string with biological methods. For example:

```
>>> from Bio.Seq import Seq
>>> my_seq = Seq("AGTACACTGGT")
>>> my_seq
Seq('AGTACACTGGT')
```

Biopython doesn't know if this is a nucleotide sequence or a protein rich in alanines, glycines, cysteines and threonines. If *you* know, keep this in mind when you call methods like (reverse)complement - see below.

General methods

The `Seq` object has a number of methods which act just like those of a Python string, for example the find method:

```
>>> from Bio.Seq import Seq
>>> my_dna = Seq("AGTACACTGGT")
>>> my_dna
Seq('AGTACACTGGT')
>>> my_dna.find("ACT")
5
>>> my_dna.find("TAG")
```