

ビッグデータプログラミング の基礎(python)

社会はプログラムに満ちている

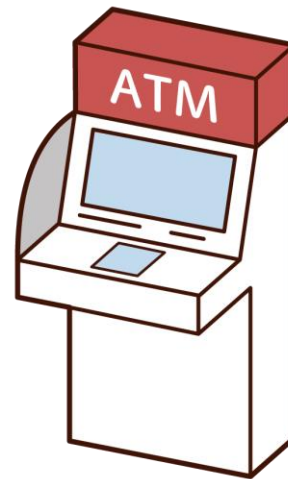
1. ありふれた日常に、潜んでいるプログラミング
2. 決して最新テクノロジーということでもない



リモコン制御



インターネットショッピング



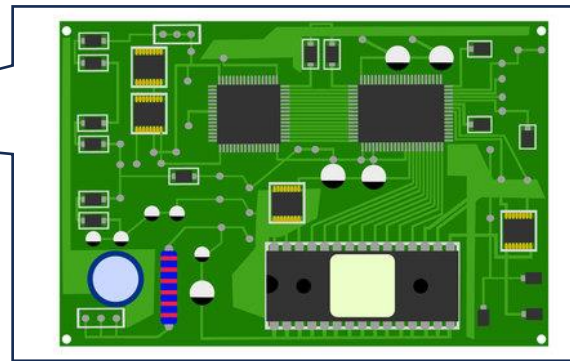
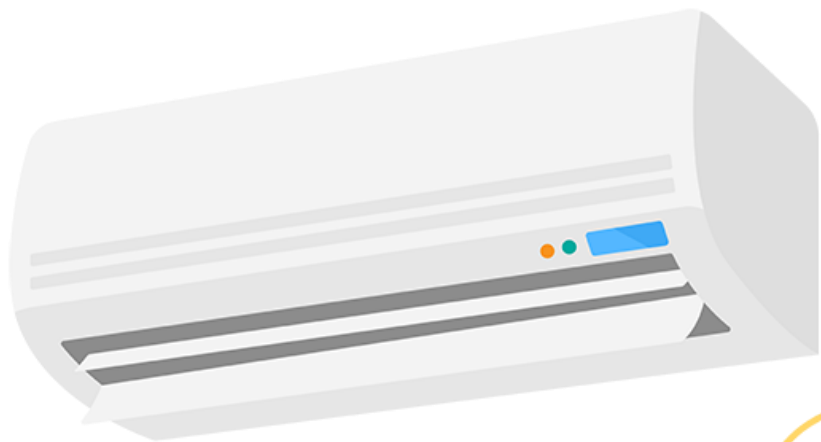
プログラムを一言でいうと。

コンピュータを動かす命令群

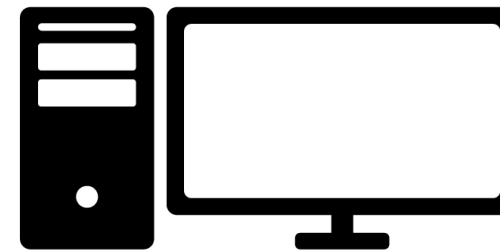
コンピュータとは

プログラムから与えられた命令をハードウェアで実行する

リモコンからの命令（信号）を受け取って何をしたらいいかを解釈・実行するコンピュータが入っている

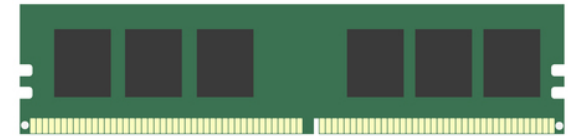
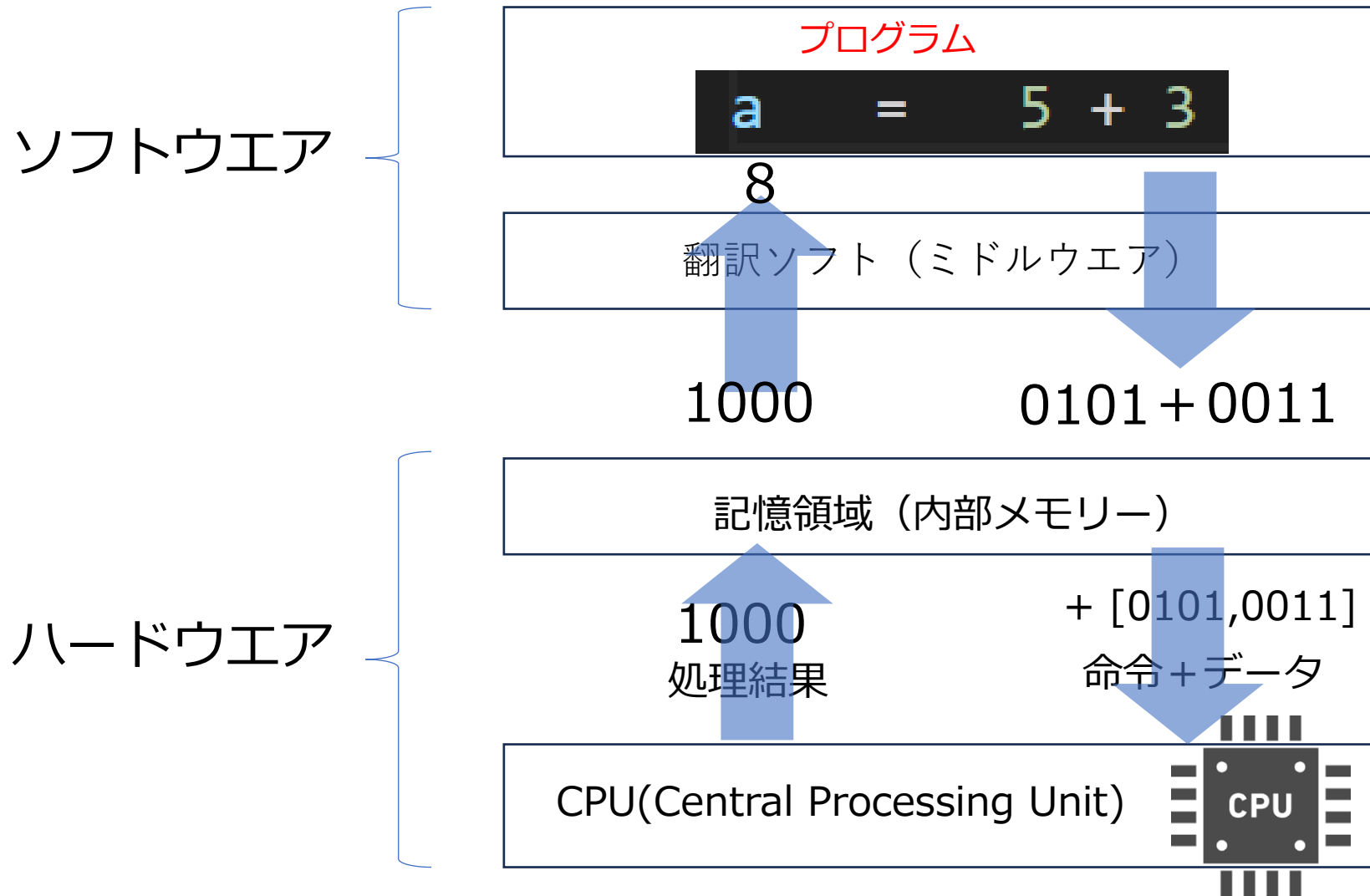


PCの中身もこれ



コンピュータはプログラムによって何をすればいいかを理解する

1. コンピュータは2進数しか理解しない
2. プログラムはそのままではコンピュータは理解できない
3. プログラムを2進数に翻訳する必要がある



プログラムを記述するための言語

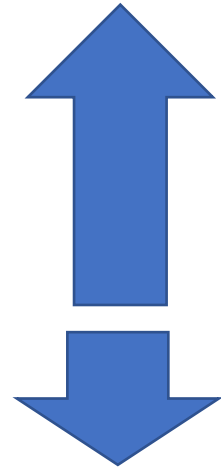
1. プログラミング言語と呼ばれる
2. from, import, for, in のようにところどころ英語っぽい

```
from janome.tokenizer import Tokenizer
t = Tokenizer()
for token in t.tokenize('すももももももものうち'):
    print(token)
```

すもも	名詞,一般,* ,* ,* ,* ,すもも,スモモ,スモモ
も	助詞,係助詞,* ,* ,* ,* ,も,モ,モ
もも	名詞,一般,* ,* ,* ,* ,もも,モモ,モモ
も	助詞,係助詞,* ,* ,* ,* ,も,モ,モ
もも	名詞,一般,* ,* ,* ,* ,もも,モモ,モモ
の	助詞,連体化,* ,* ,* ,* ,の,ノ,ノ
うち	名詞,非自立,副詞可能,* ,* ,* ,うち,ウチ,ウチ

プログラミング言語は進化する

機械に優しい=高速処理に向いている（書くのは大変！）



1960年代：アセンブラ

1970年代：COBOL, FORTRAN

1980年代：C, C++

1990年代：java, PHP, python

人間にやさしい=大規模システムに向いている
（バグが比較的少ない：でも遅い！）

機械にやさしい時代のプログラミング



プログラミング言語のなわばり

C, C++



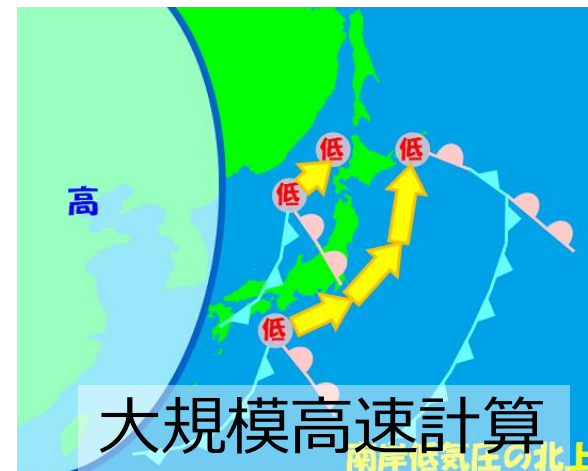
リアルタイム処理

COBOL



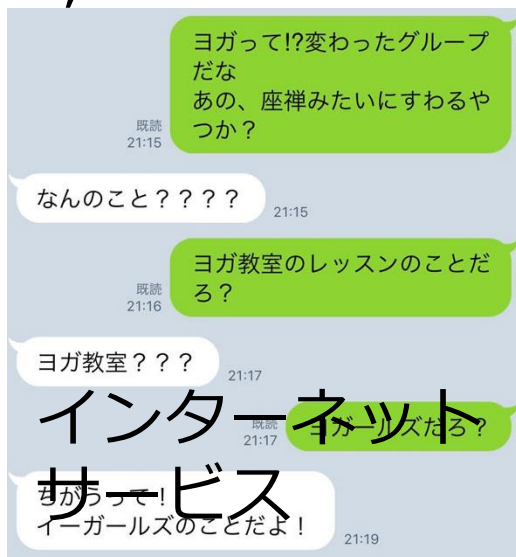
信頼性

FORTRAN



大規模高速計算

Java, PHP



python



脚光を浴びる pythonTM

名前の由来は、イギリスのテレビ局 BBC が製作したコメディ番組『空飛ぶモンティ・パイソン』である。Pythonという英単語が意味する爬虫類のニシキヘビがPython言語のマスコットやアイコンとして使われている。

グルー言語（糊付け）と呼ばれる
Importできるモジュール群がダントツ豊富

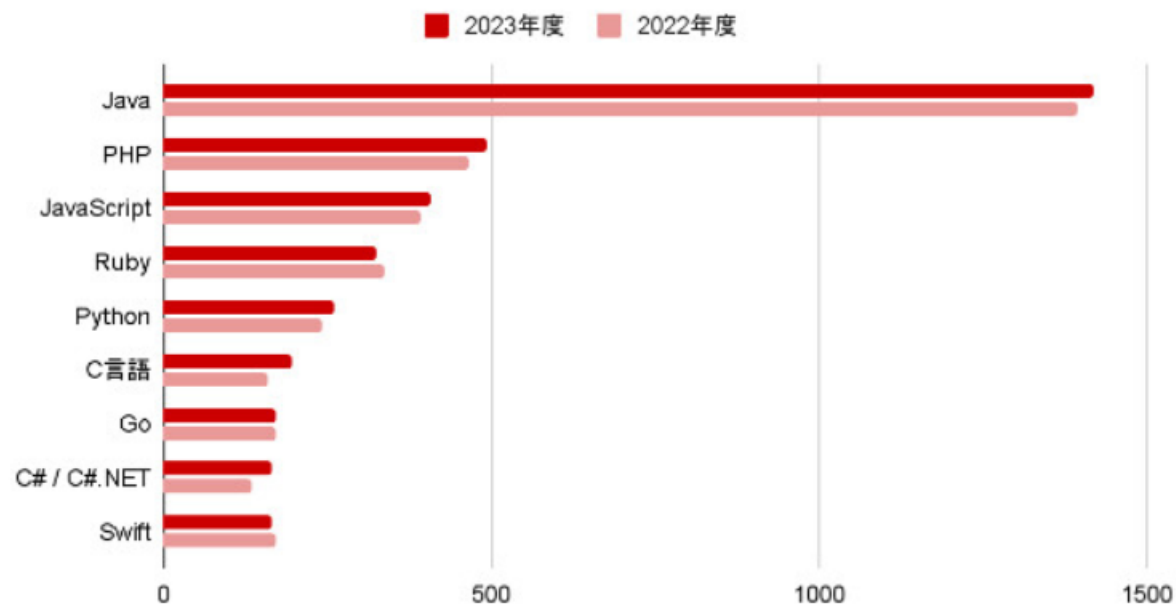
プログラミング言語別求人数ランキング

<https://career.levtech.jp/guide/knowhow/article/918/>




プログラミング言語別の求人数は以下の通りです。

1位	Java	1418件（BtoB：1219件、BtoC：199件）
2位	PHP	494件（ソーシャルゲーム：37件、ソーシャルゲーム以外：457件）
3位	JavaScript	406件（フロント：326件、サーバ：80件）
4位	Ruby	326件
5位	Python	261件

プログラミング言語別求人数



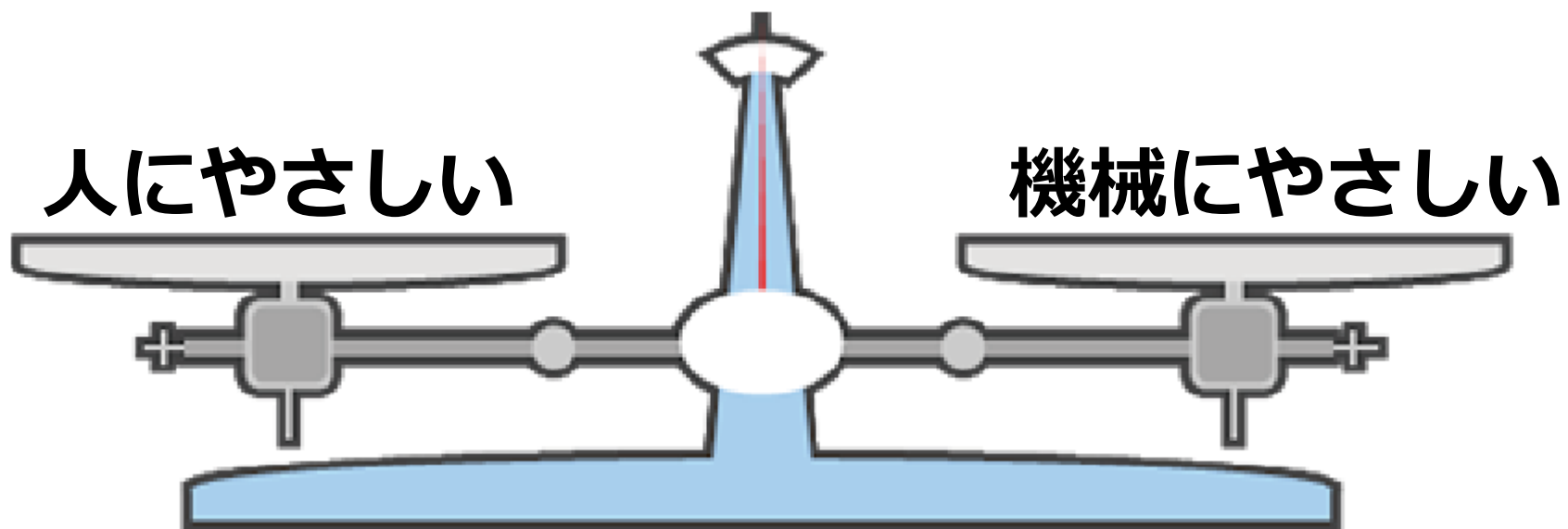
2024年 | 案件が多い「プログラミング言語」ランキング

順位	言語	案件比率	月収
 1	JavaScript	21.2%	87万円
 2	PHP	13.0%	81万円
 3	Python	12.9%	104万円
4	TypeScript	11.0%	103万円
5	Ruby	10.7%	94万円
6	Java	8.9%	79万円

<https://magazine.sokudan.work/post/hbyTpL8z>

Python のブレイクスルーは

Python 3.0 2008～



Pythonでプログラミングしてみる

コンピュータはコマンドラインから始まった

Windowsの前身：MS-DOS

PCを起動するといきなりコマンドラインになった



人が直感的に操作できるようにマウスを発明

➡ GUI(Graphic User Interface)

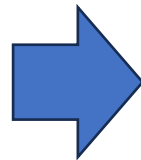
今でも、ソフトウェア開発はコマンドラインがベース。
Microsoft officeだって開発時はコマンドライン つまり、
IT系を目指すにはコマンドライン習得が必須



コマンドラインからプログラムを書く

数学とよく似ている

```
>>> a=1+5
>>> a
6
>>> b=a*2
>>> b
12
>>> c=b**2
>>> c
144
```



a=2+5の間違い
だったとしたら



最初から入力しなお
しとなる

1行でも間違えると振出しに戻る！

もっとプログラミングを快適にできないか

1. プログラミング環境！
2. 修正箇所だけ直して再実行

colaboratory

Google Colaboratory



Jupyter



colaboratory

Colaboratoryを起動する

Microsoft Edgeを立ち上げ、'colaboratory' で検索し、Google Colaboratoryのページを開く

 Microsoft Bing





 検索

 チャット

学校

画像

動画

地図

ニュース

ショッピング

 さらに表示

約 285,000 件の結果

 Google Colab
<https://colab.research.google.com/notebooks/welcome...>

Google Colaboratory

ウェブ Colab（正式名称「**Colaboratory**」）では、ブラウザ上で Python を記述、実行できます。以下の機能を使用できます。環境構築が不要 GPU に料金なしでアクセス 簡単に共有

データの読み込み

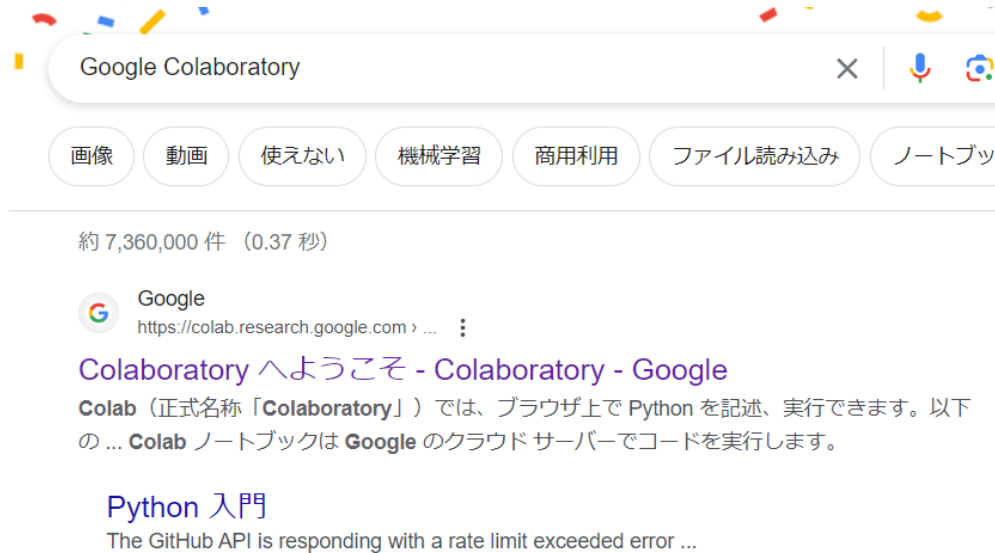
In order to use Colaboratory with GCS, you'll need to create a Google Cloud ...

グラフ

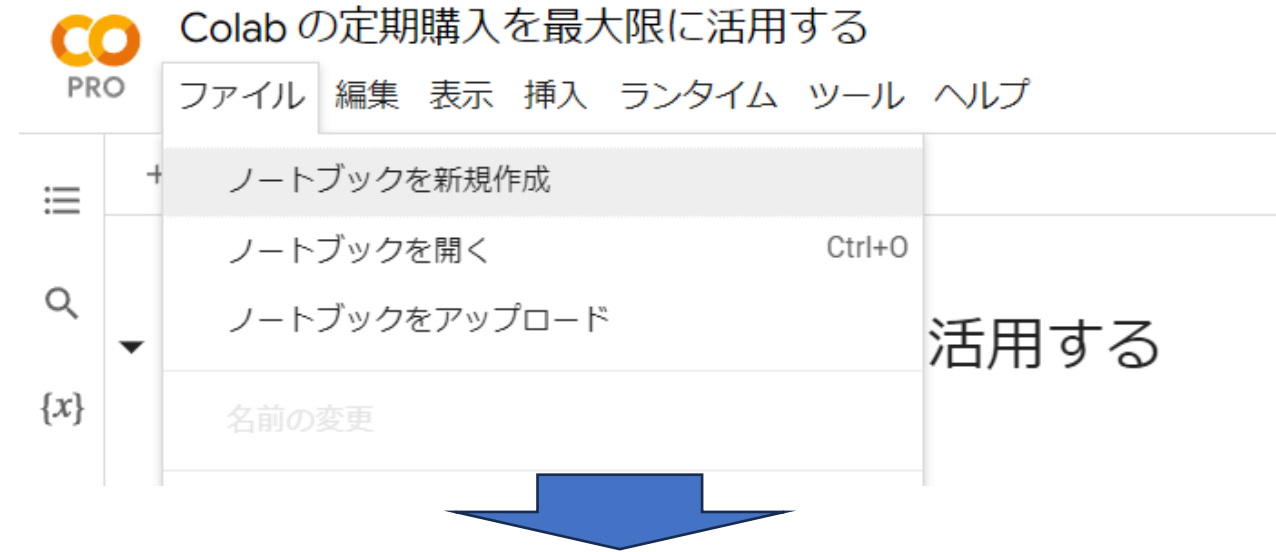
Google Colab ... Sign in

Google colaboratoryの基本操作

1. Google colaboratoryで検索



2. Colabの新規作成を選択



活用する

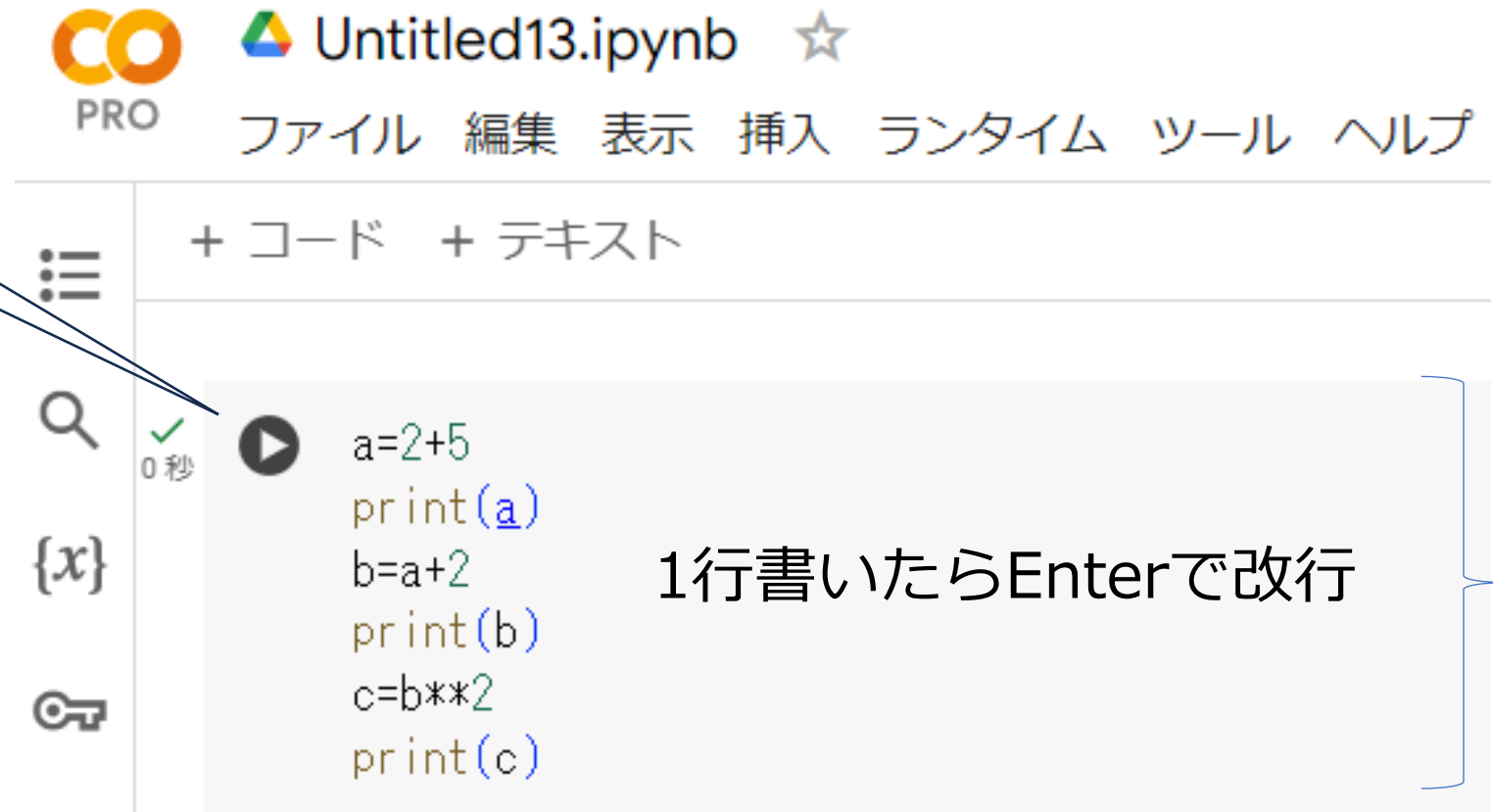
3. 以下でpythonのプログラミングが可能



プログラムを書いて実行する

セルの中にプログラムを書く

プログラムの実行
(Shift Enterを同時に
押してもOK)



The screenshot shows the JupyterLab interface. At the top, there's a header bar with the 'CO PRO' logo, the file name 'Untitled13.ipynb', and a star icon. Below this is a menu bar with options: 'ファイル' (File), '編集' (Edit), '表示' (View), '挿入' (Insert), 'ランタイム' (Runtime), 'ツール' (Tools), and 'ヘルプ' (Help). The main area is divided into a left sidebar and a central workspace. The sidebar contains icons for a menu, search, a variable '{x}', and a key. The central workspace has a toolbar with '+ コード' (Add Code) and '+ テキスト' (Add Text). Below the toolbar is a code cell. The cell has a play button icon, a green checkmark, and '0 秒' (0 seconds). The code inside the cell is:

```
a=2+5
print(a)
b=a+2
print(b)
c=b**2
print(c)
```

1行書いたらEnterで改行

セル

セル単位でプログラムをひとまとまり書くことができる

新しいセルを作る

CO PRO Untitled13.ipynb ☆

ファイル 編集 表示 挿入 ランタイム ツール ヘルプ

+ コード + テキスト

[3] a=2+5
print(a)
b=a*2
print(b)
c=b**2
print(c)

7
14
196

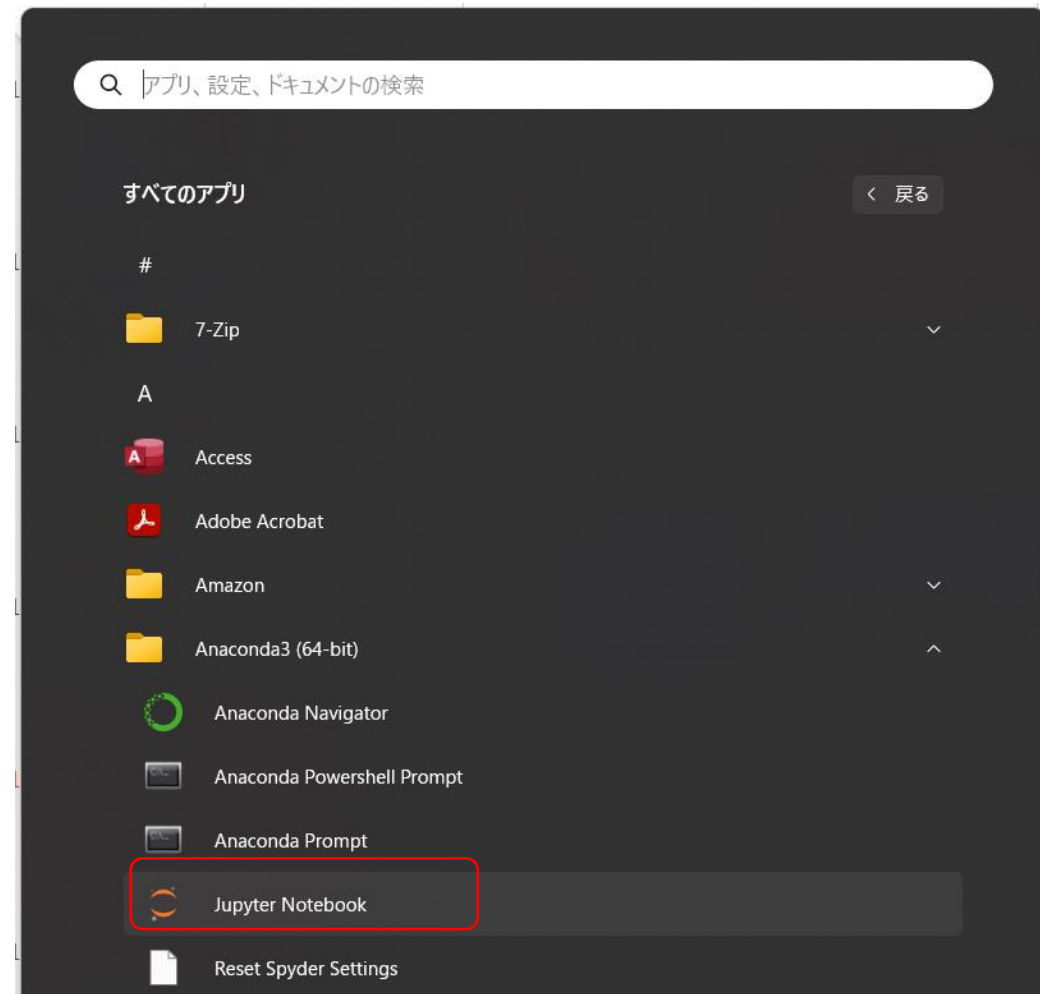
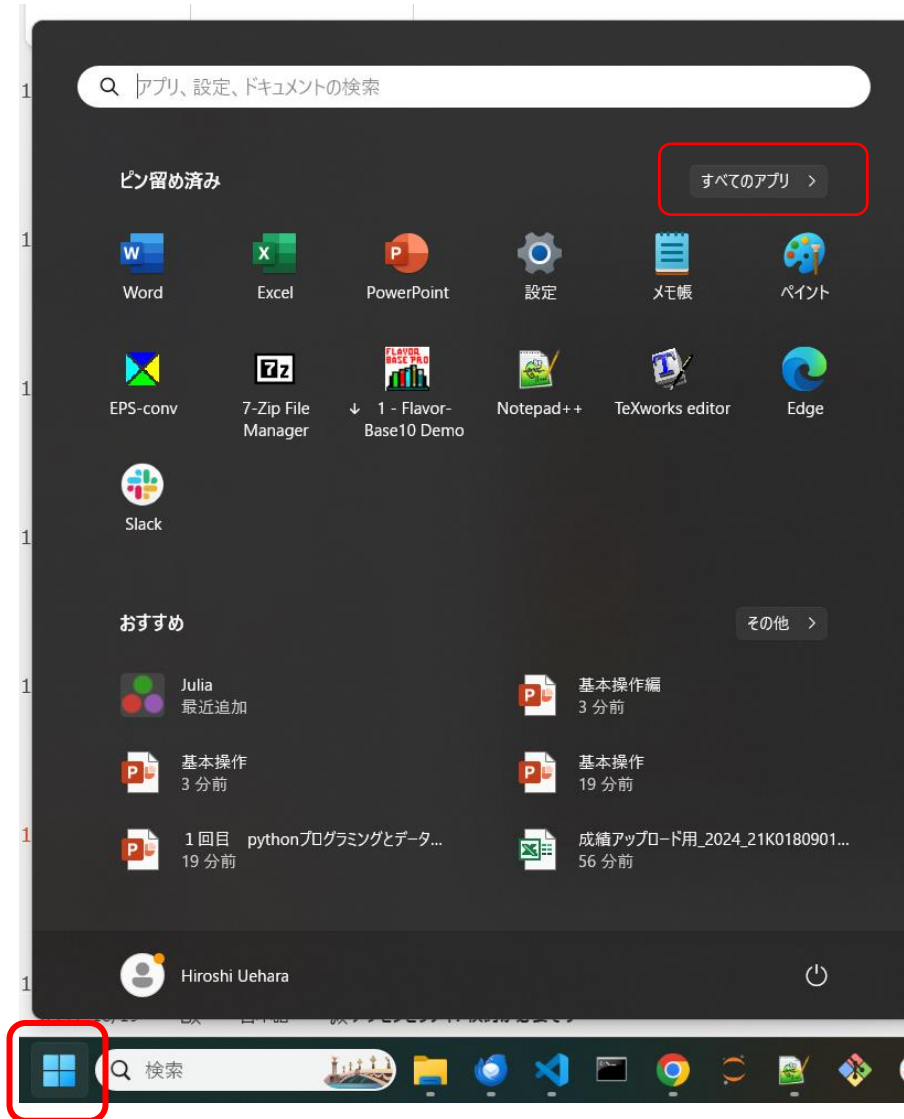
a=2*5
print(a)
d=a+b+c
print(d)

10
220

- 変数aは書き換わる
- b,cは上のセルのプログラムの結果のまま

Jupyter

Jupyterの起動



Files

Running

Select items to perform actions on them.

/

<input type="checkbox"/> Name	La	Size
<input type="checkbox"/> aws		
<input type="checkbox"/> Contacts		
<input type="checkbox"/> Desktop		
<input type="checkbox"/> Documents		
<input type="checkbox"/> Downloads		
<input type="checkbox"/> Dropbox		
<input type="checkbox"/> ...		

New

Upload

Refresh

Notebook

Terminal

Console

New File

New Folder

yesterday

4 days ago

1 hour ago

last month

last week

jupyter Untitled Last Checkpoint: 1 minute ago

File Edit View Run Kernel Settings Help

Trusted

+

✂

📄

📄

▶

■

↺

▶▶

Code

▼

JupyterLab

No Kernel

[]:

📄

↑

↓

📄

🔍

🗑

Select Kernel

Select kernel for: "Untitled.ipynb"

Python 3 (ipykernel)

☐ Always start the preferred kernel

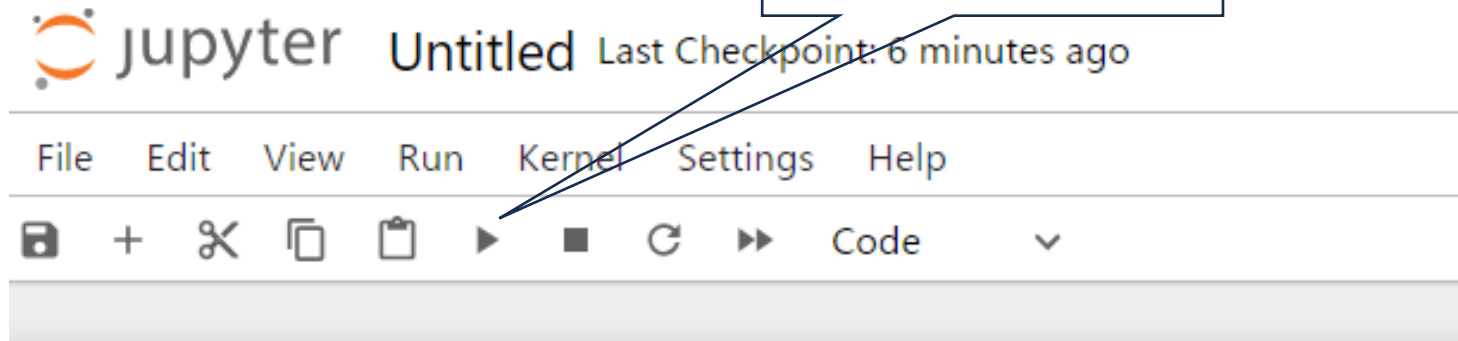
No Kernel

Select

プログラムを書いて実行する

セルの中にプログラムを書く

プログラムの実行
(Shift Enterを同時に押してもOK)



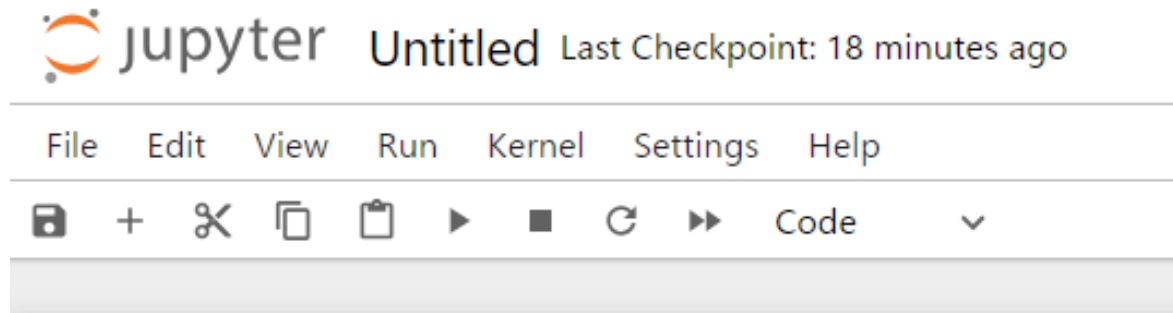
```
[5]: a=2+5  
      print(a)  
      b=a*2  
      print(b)  
      c=b**2  
      print(c)
```

1行書いたらEnterで改行

セル

```
7  
14  
196
```

セル単位でプログラムをひとまとまり書くことができる



```
[5]: a=2+5  
      print(a)  
      b=a*2  
      print(b)  
      c=b**2  
      print(c)
```

```
7  
14  
196
```

```
[8]: a=2*5  
      print(a)  
      d=a+b+c  
      print(d)
```

```
10  
220
```

- 変数aは書き換わる
- b,cは上のセルのプログラムの結果のまま

プログラミング言語には文法がある

プログラムの基本構文

1. アルファベットで始まる任意の記号
2. 右辺の結果の値を保管する

変数 = 何らかの値（演算処理を含む）

変数、値どちらでもOK

例

上から下に順番
に実行される

a = 5	aには5が入る
a=2+5	aが7に置き換わる
b=a*2	7 × 2 を意味する

プログラミングとは

1. プログラムには構文（文法）がある（プログラミング言語によって異なる）
2. プログラムを習得するとはいろいろな構文を理解して自由に書けるようになること
3. プログラム言語なので言語の習得と似ている
4. ただし、人の言語よりは簡単

問題

以下のプログラムを実行すると何が起きるか？

$$a = d + 5$$

変数は定義しないと使えない！

(変数は**最初に**必ず等式の左辺に置くこと)

$$d = 2**3$$
$$a = d + 5$$

プログラミングを勉強する 2つの考え方

0 からオリジナルにプログラムを作成する

将来プロフェッショナルデータサイエンティストを目指す

既存のプログラムを修正していろいろなデータ分析に応用する

様々な企業でのデータ活用業務をこなす

