

「DSことはじめ」

九州大学 大学院システム情報科学研究所
情報知能工学部門

備瀬竜馬, Diego Thomas, 正井克俊

データサイエンスに必要なスキル

データ解析スキル
検定、クラスタリング、回帰、
主成分分析、・・・



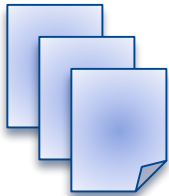
データを集めて後、実際にこれらの分析を行うには？



プログラミングスキル
解析のための具体的なデータ加工処理を
コンピュータに指示

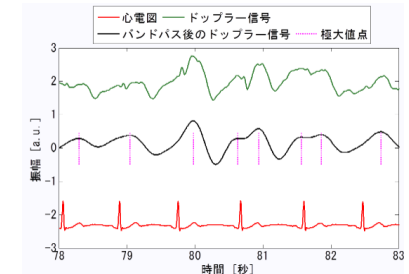
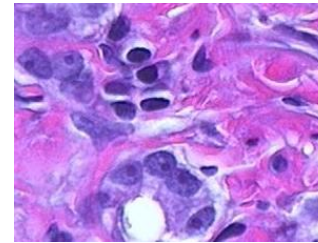
データ解析

数値・カテゴリデータ 解析



数値データ
身長・体重
血圧
カテゴリデータ
アンケート
男女
喫煙の有無

メディアデータ解析 画像 シグナル



細胞の大きさ、心拍間隔等
画像データからは直接抽出
できない場合が多く、事前
処理が必要



本講義：

まず、数値・カテゴリデータを対象とした解析手法について学び、
画像を例にメディアデータを対象とした解析手法を学ぶ

Excel じゃダメ？

- 「まずは Excel で見てみる」は大事
 - データの集計，簡単な可視化
 - Excel で完結するのであればそれでヨシ
- もっと高度な分析がしたい！
 - 統計分析，機械学習，
- エクセルじゃ開けない！

本日の講義内容

- データサイエンス技術のプログラム実践前の準備
- プログラムの基本的な内容を勉強する
 - Google Colabの準備
 - コンピュータに命令する：コンソール上のコマンド
 - コンピュータにまとめて命令：プログラムコード
 - 演習問題

プログラムの準備 Python

プログラムとは

- コンピュータに対する命令の記述
- 命令は何語？
 - C
 - Java
 - R
 - Python

Python の利用

Google Colaboratoryを利用
以下の web site へ

<https://colab.research.google.com/notebooks/welcome.ipynb?hl=ja#scrollTo=-Rh3-Vt9Nev9>

Google Colaboratory

← → ↻ https://colab.research.google.com/notebooks/welcome.ipynb?hl=ja#scrollTo=xitplqMNk_Hc ☆

Colaboratory へようこそ

共有 ログイン

+ コード + テキスト ドライブにコピー

接続 編集

目次 コードスニペット ファイル

Colaboratory の概要

はじめに

その他のリソース

機械学習の例: Seedbank

セクション

Colaboratory へようこそ

Colaboratory は、完全にクラウドで実行される Jupyter ノートブック環境です。設定不要で、無料でご利用になれます。

Colaboratory を使用すると、コードの記述と実行、解析の保存や共有、強力なコンピューティングリソースへのアクセスなどをブラウザからすべて無料で行えます。

Colaboratory の概要

この 3 分間の動画では、Colaboratory の主な機能の概要をご紹介します。

Get started with Google Colaboratory (Coding T...

Intro to Google Colab

Coding TensorFlow

自分のGoogle
アカウントで
ログイン

ファイルを
セーブできる
ようになる

Jupyter notebookを使う

Jupyter notebook: Pythonの開発環境の一つ

ファイルを
クリック

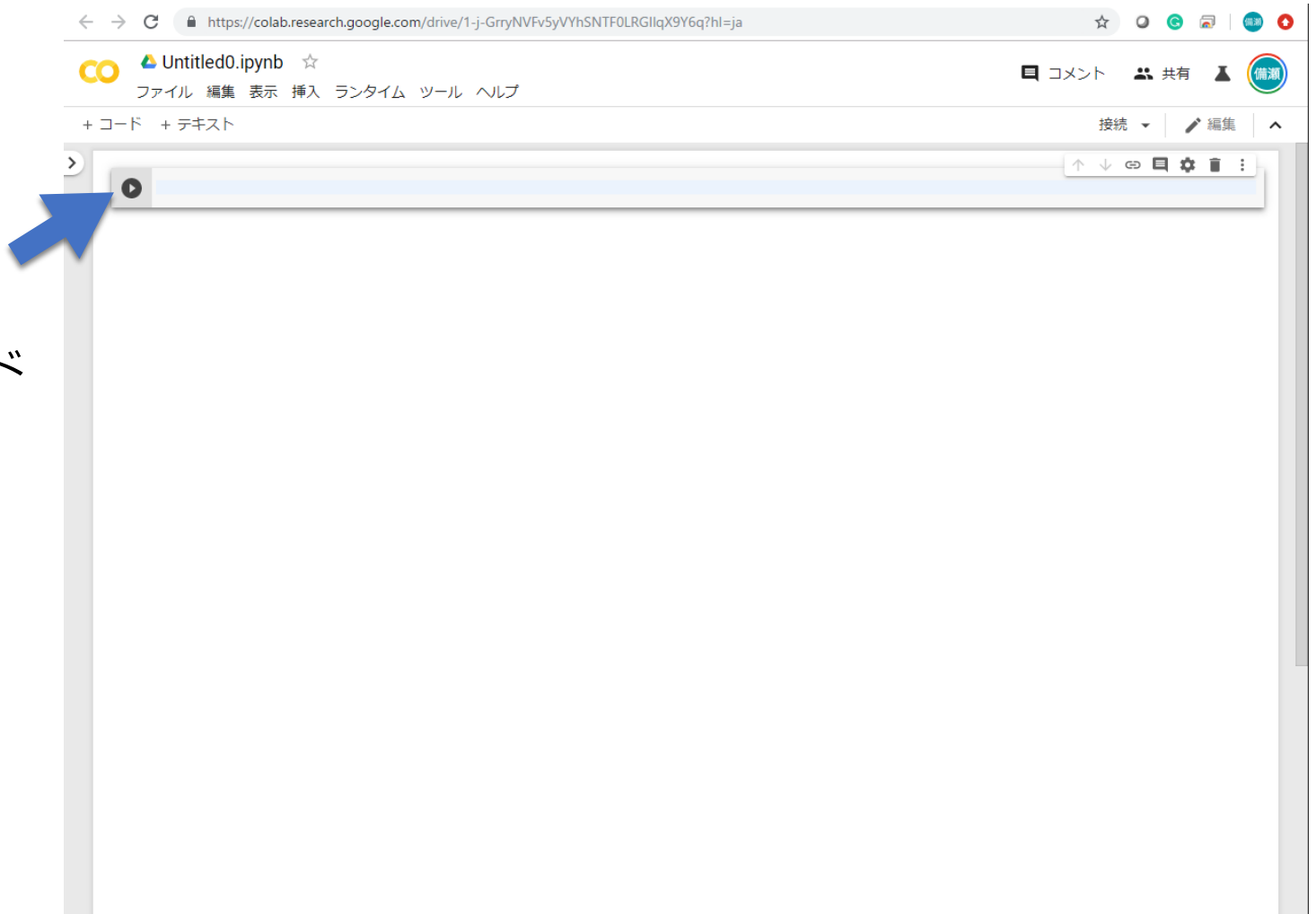
Python3の新しい
ノートブック
をクリック

※) 数字はPython
のバージョン

The screenshot shows the Google Colaboratory web interface. The browser address bar displays `https://colab.research.google.com/notebooks/welcome.ipynb?hl=ja`. The page title is "Colaboratory へようこそ". The "File" menu is open, showing options like "Python 3 の新しいノートブック" (highlighted with a blue box and a blue arrow), "Python 2 の新しいノートブック", "ノートブックを開く...", "ノートブックをアップロード...", "名前を変更...", "ゴミ箱に移動", "ドライブにコピーを保存...", "コピーを GitHub Gist として保存...", "GitHub にコピーを保存...", "保存" (with keyboard shortcuts Ctrl+S and Ctrl+M S), "変更内容を保存して固定", "変更履歴", ".ipynb をダウンロード", ".py をダウンロード", "ドライブプレビューを更新", and "印刷" (with keyboard shortcut Ctrl+P). The main content area displays a welcome message and a video titled "Intro to Google Colab" featuring a man in a blue shirt. The video player has a play button and a "Get started with Google Colaboratory (Coding T..." title. The bottom of the page shows the "Coding TensorFlow" logo.

Cell

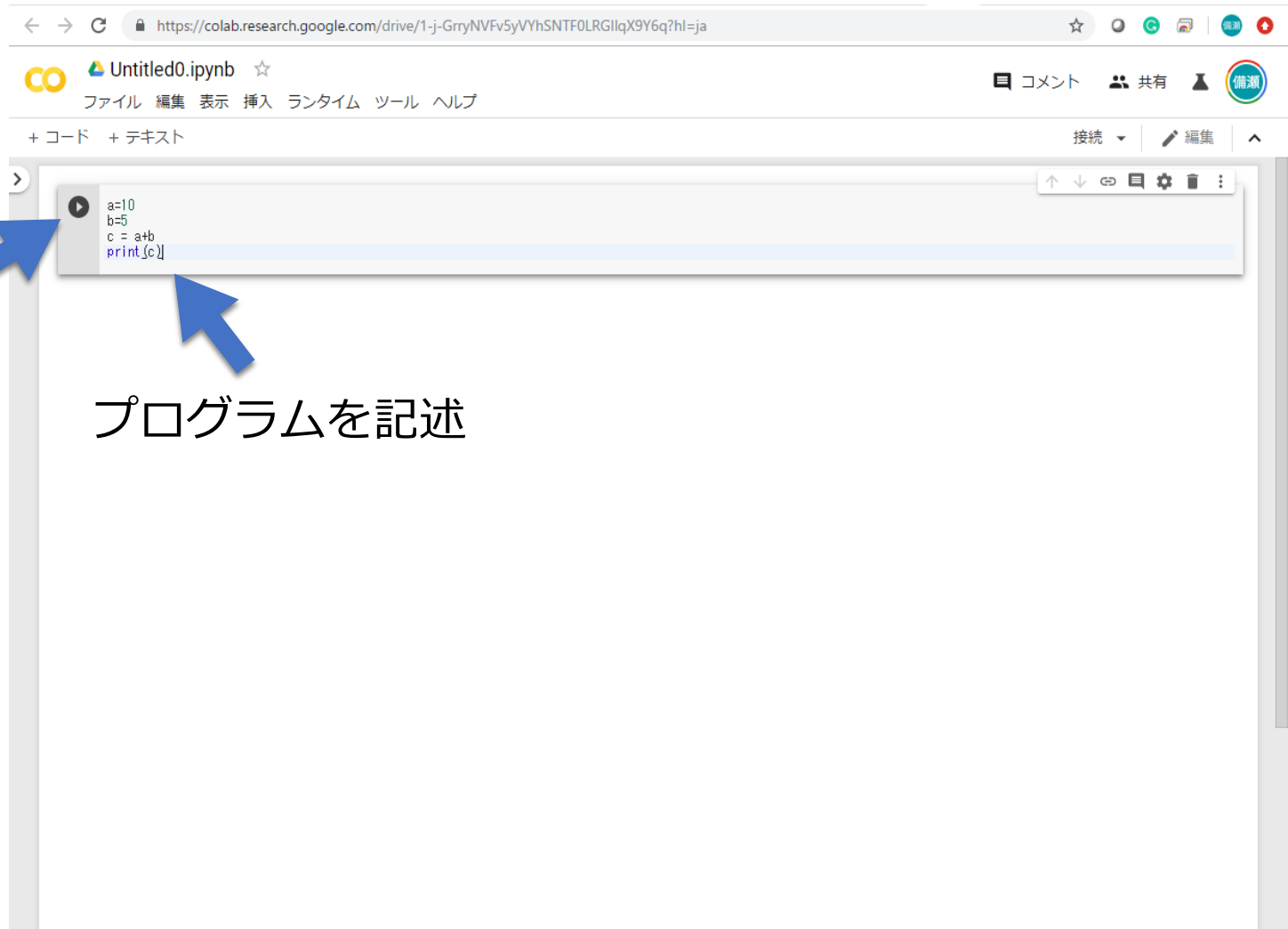
Cell
プログラムの
入力フィールド
Cellの単位で
実行される



Cellへの入力

▶ ボタンを
クリックで
実行

プログラムを記述



Cellの実行

実行結果が
表示される



Cellの追加

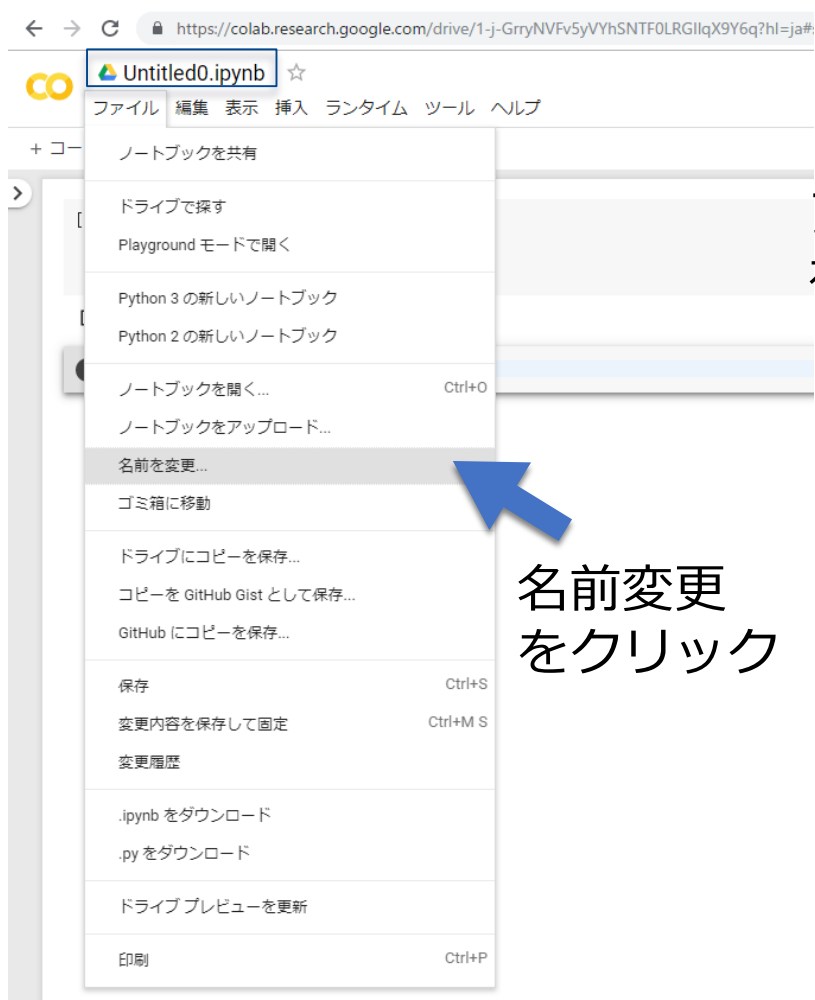
+コードを
クリック



新たなCellが
追加される

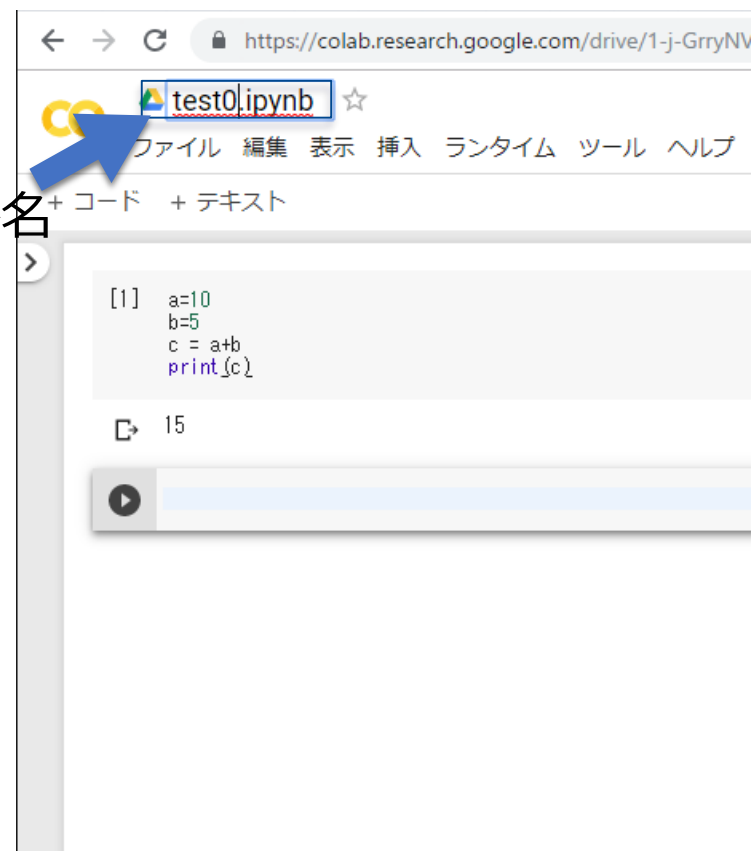


ファイル名を変更

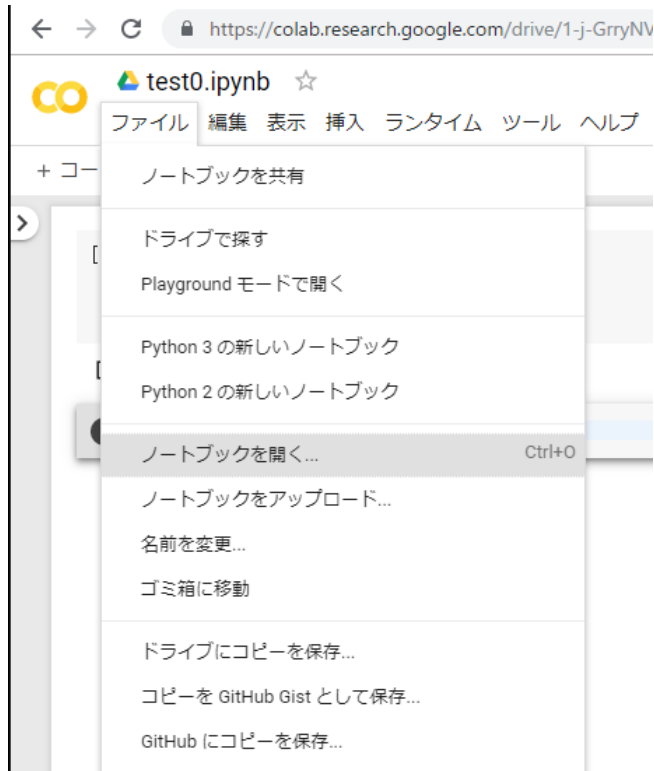


ファイル名
を編集

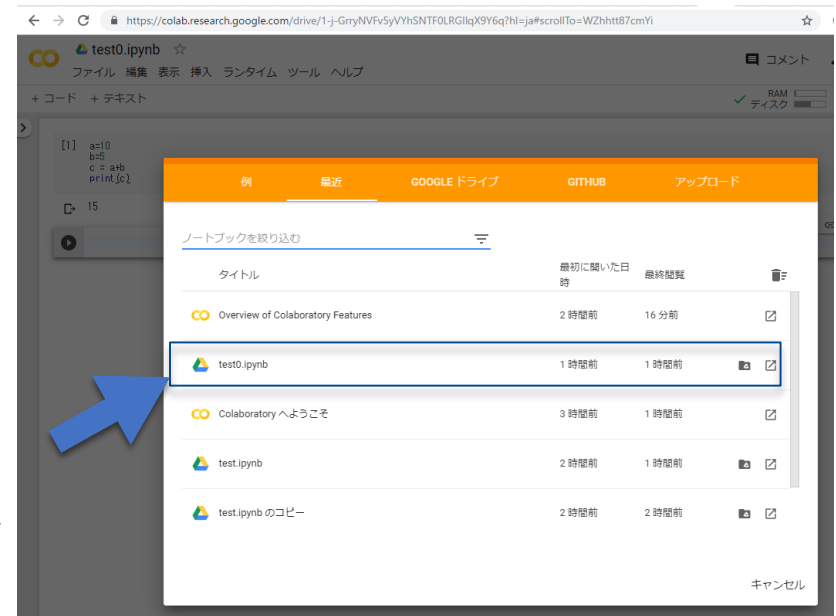
名前変更
をクリック



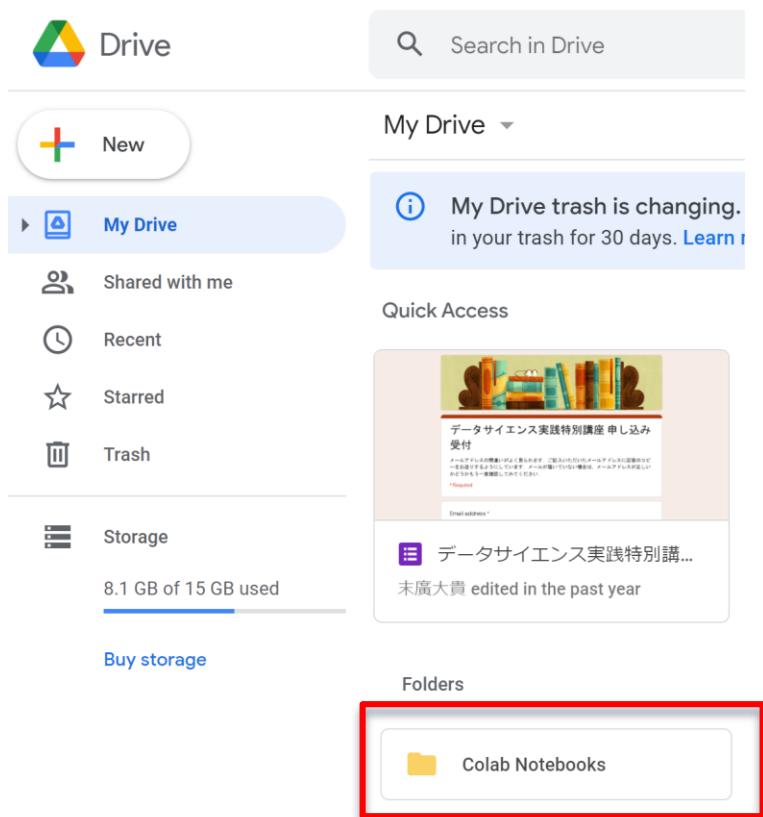
ファイルを開く



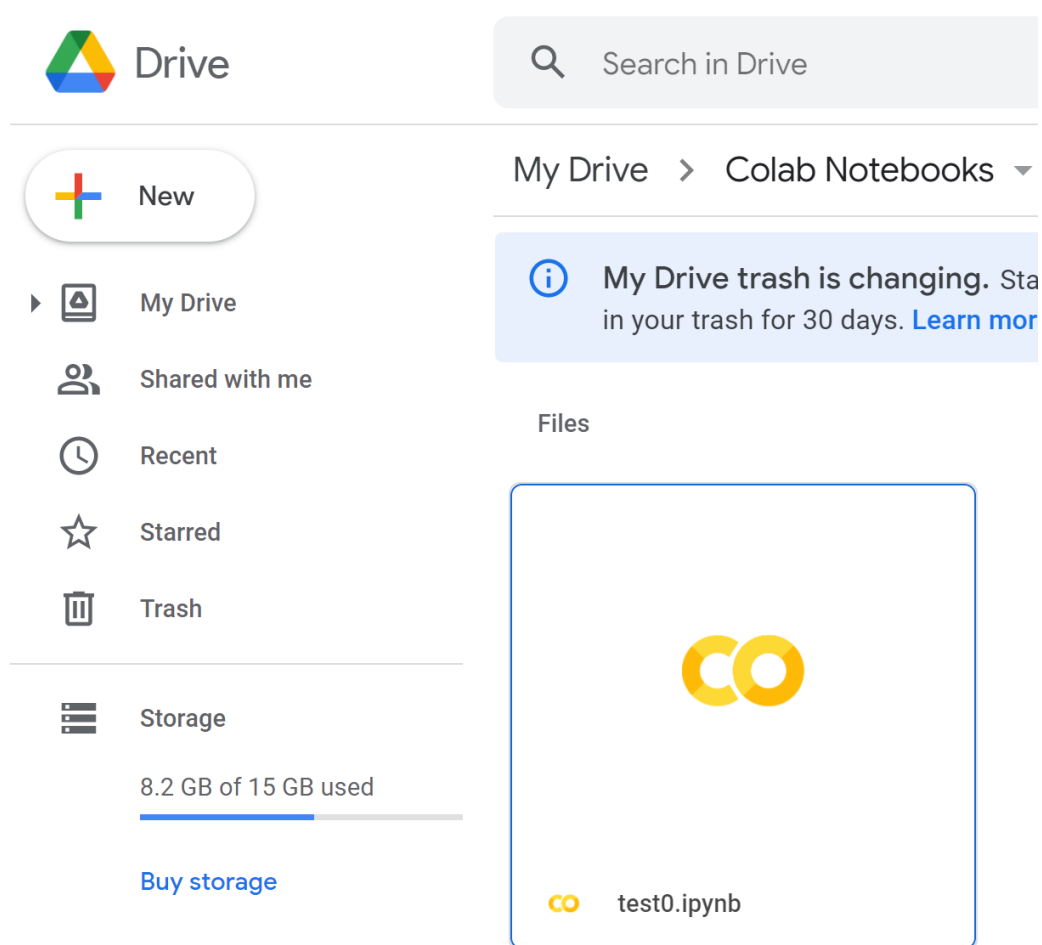
さっきの
ファイルが
選択できる



ファイルの保存先 : Google Drive

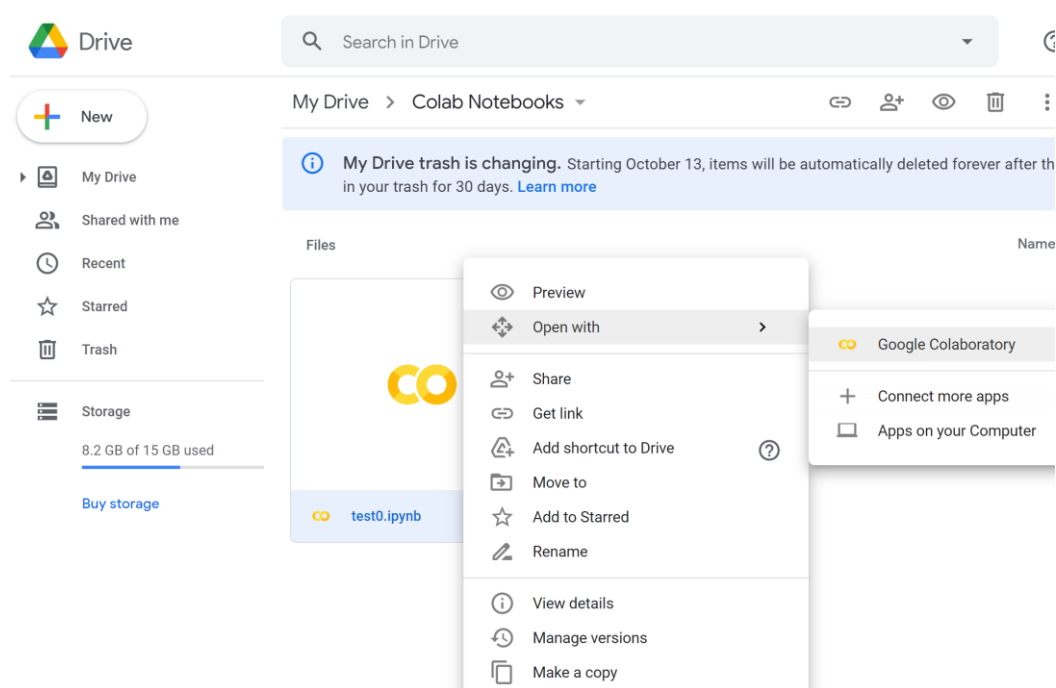


自動的に同じアカウントの
Google Drive の下に
フォルダが作成されている



先ほどのファイルが保存されている

ファイルの保存先 : Google Drive



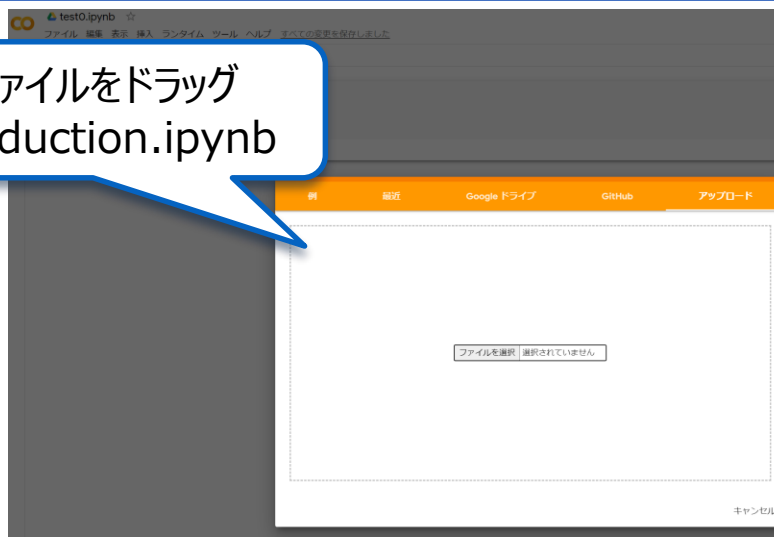
Google Driveから
ファイルを開ける

ファイルの上で右クリックして、
Open with -> Google Colaboratory をクリック

Fileのアップロード



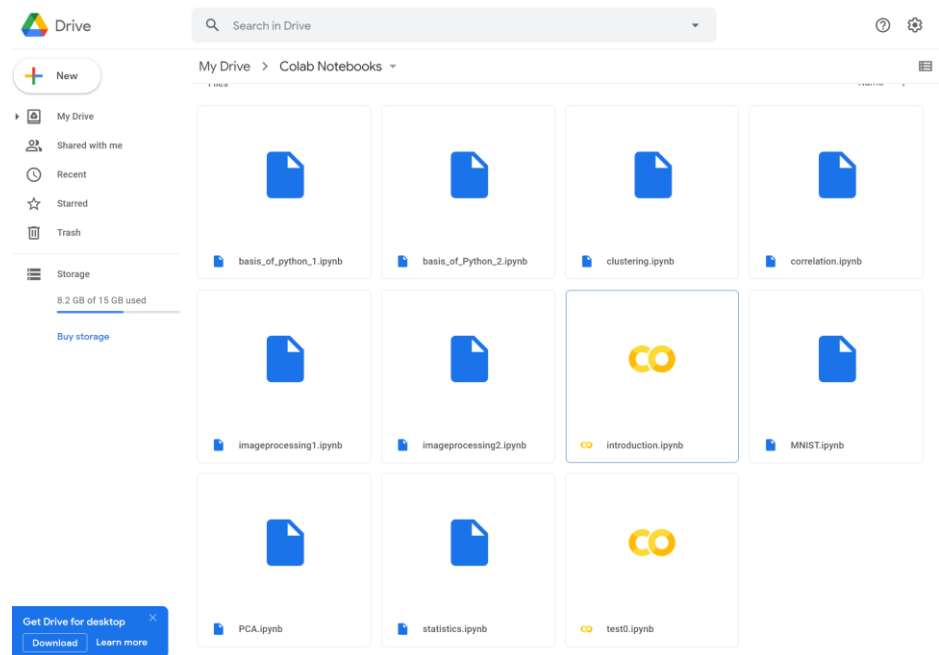
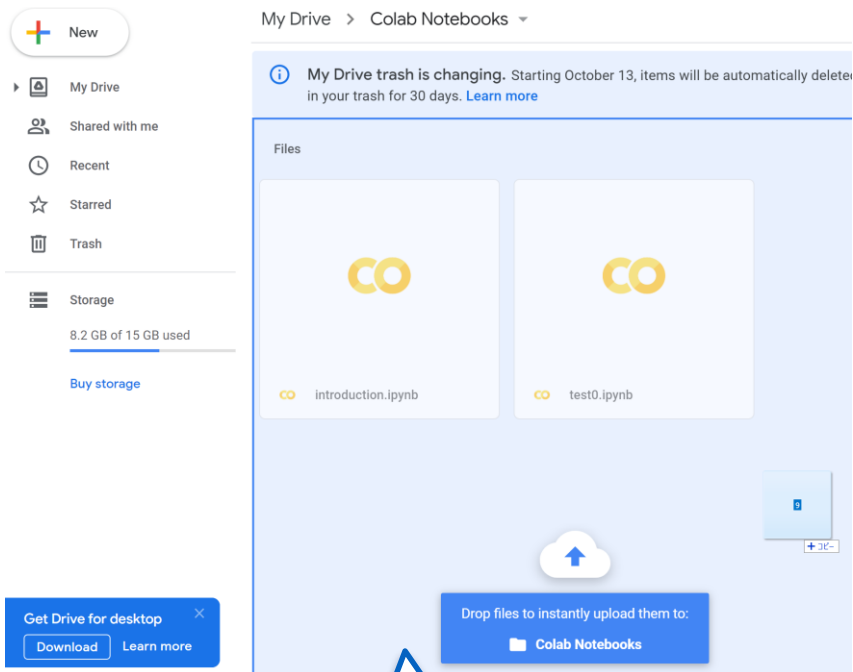
PCからファイルをドラッグ
例：introduction.ipynb



Introduction.ipynbが開く

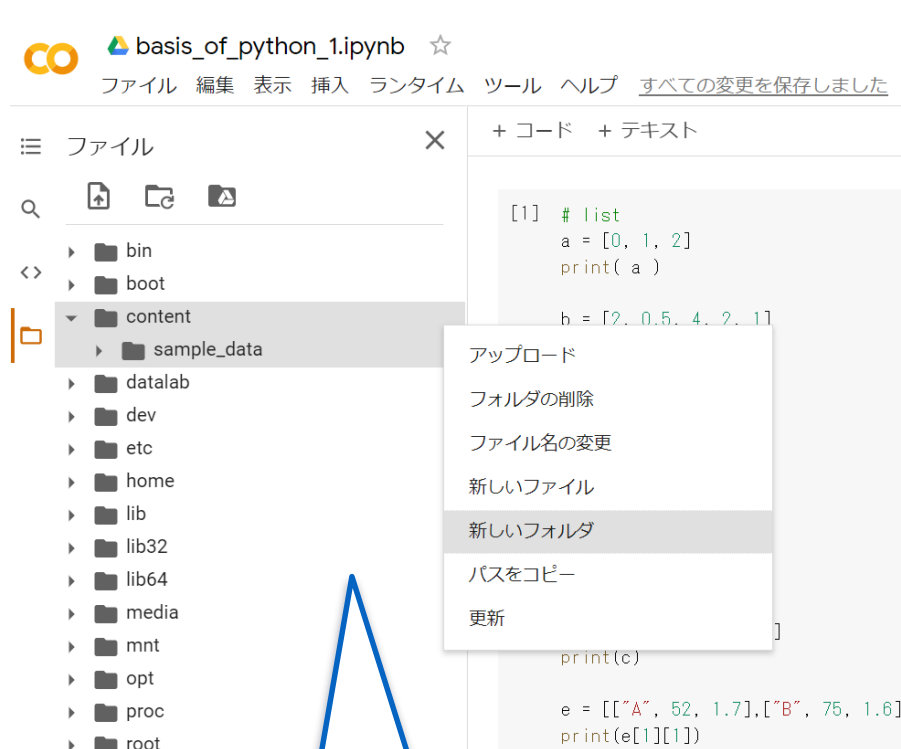


DriveへのFileアップロード



複数ファイルまとめてDrop

新しいフォルダの作成



basis_of_python_1.ipynb ☆

ファイル 編集 表示 挿入 ランタイム ツール ヘルプ すべての変更を保存しました

ファイル

- bin
- boot
- content
 - sample_data
- datalab
- dev
- etc
- home
- lib
- lib32
- lib64
- media
- mnt
- opt
- proc
- root

アップロード
フォルダの削除
ファイル名の変更
新しいファイル
新しいフォルダ
パスをコピー
更新

```
[1] # list
a = [0, 1, 2]
print(a)

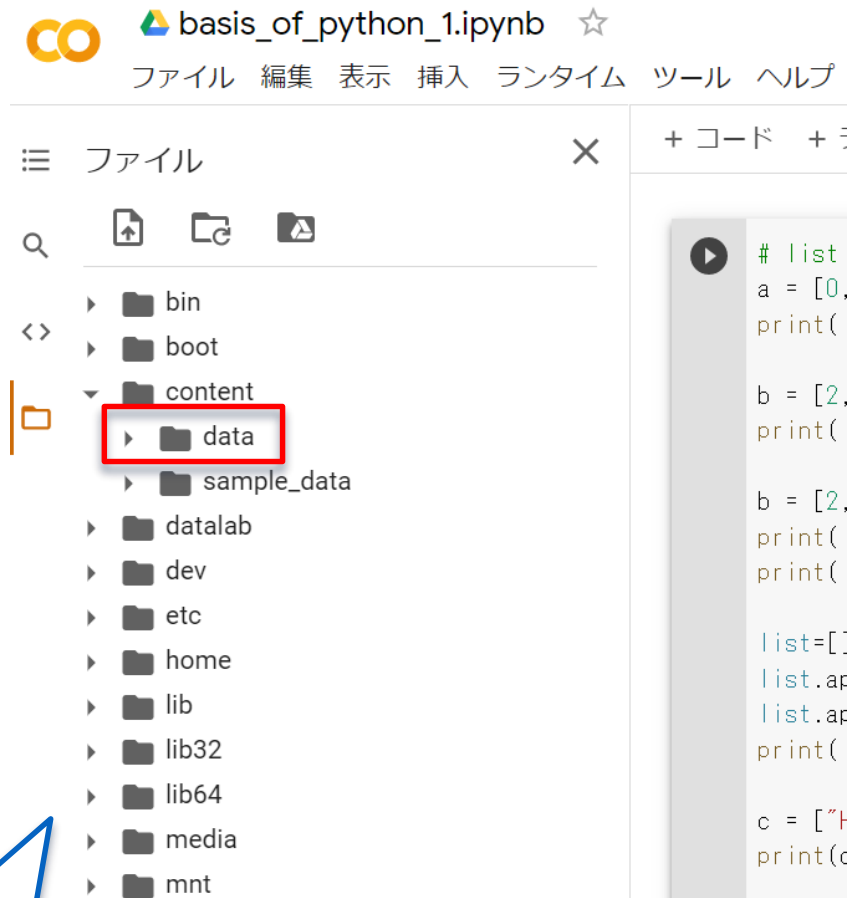
b = [2, 0.5, 4, 2, 1]

print(c)

e = [{"A", 52, 1.7}, {"B", 75, 1.6}]
print(e[1][1])
```

Contentの下に新しい
フォルダ'data'を作成

フォルダができた



basis_of_python_1.ipynb ☆

ファイル 編集 表示 挿入 ランタイム ツール ヘルプ

ファイル

- bin
- boot
- content
 - data
 - sample_data
- datalab
- dev
- etc
- home
- lib
- lib32
- lib64
- media
- mnt

```
# list
a = [0,
print(

b = [2,
print(

b = [2,
print(
print(

list=[
list.ap
list.ap
print(

c = [{"
print(c
```

データのアップロード

The image illustrates the process of uploading data to a Jupyter Notebook. It consists of three main parts:

- Left Panel (Jupyter Interface):** Shows the file browser on the left with the 'data' folder selected. A context menu is open, and the 'Upload' button is highlighted. The code editor on the right shows a list of files and a loop that reads and appends files from the 'data' folder.
- Center Panel (File Explorer):** Shows a Windows File Explorer window displaying a list of CSV files in a 'Data' folder. The 'Ctrl+A' keyboard shortcut is indicated, suggesting that all files are being selected.
- Right Panel (Jupyter Code Editor):** Shows the Jupyter code editor with a list of files and a loop that reads and appends files from the 'data' folder.

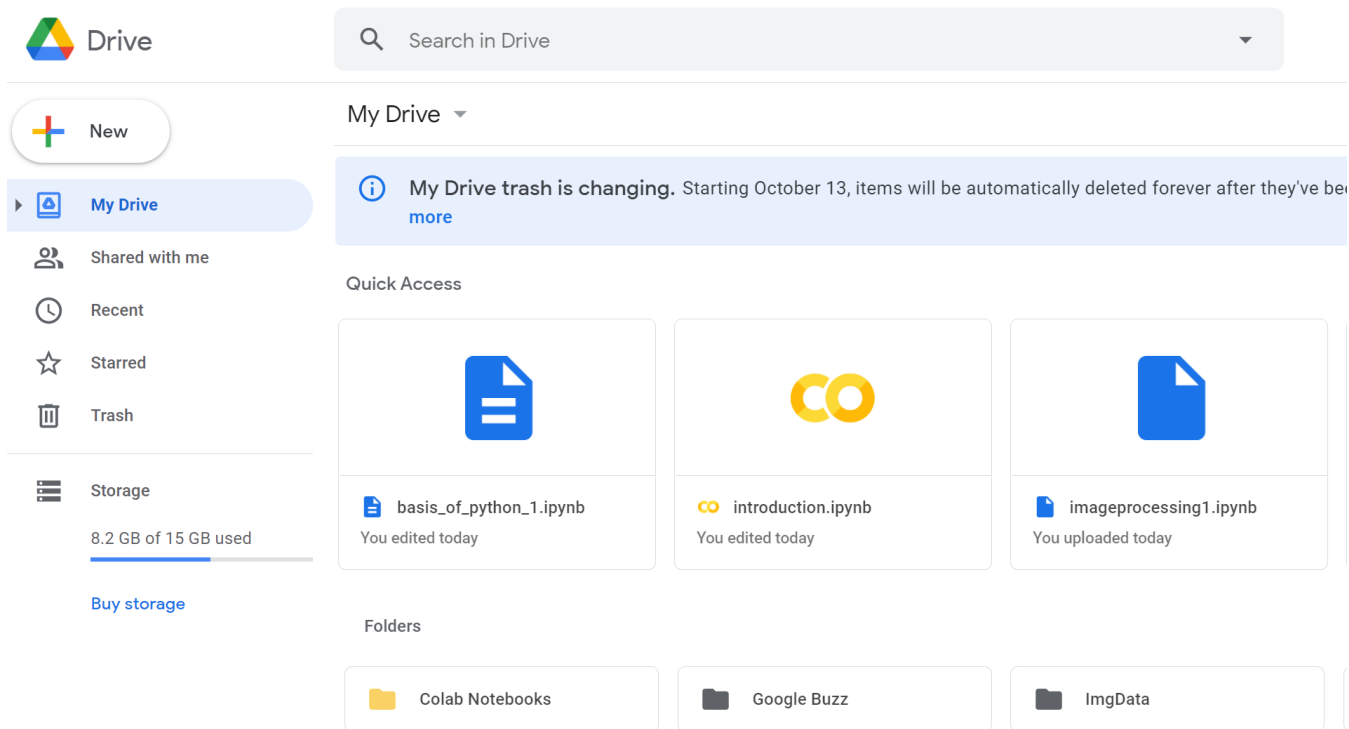
Blue callout boxes provide instructions for each step:

- Left Callout:** Data フォルダを選択してアップロードをクリック
- Center Callout:** 配布資料のDataフォルダのパスを指定して、全てのファイルを選択
- Right Callout:** これでプログラムからファイルを読めるようになった

※) ただし、接続が切れると初期化されるので、毎日アップする必要がある

Google Driveとの連携

準備 : Google Driveへフォルダごとアップ



フォルダごとドラッグすれば、
フォルダの中身も全てアップロード可能

Google Driveとの接続

Colaboratory へようこそ
ファイル 編集 表示 挿入 ランタイム ツール ヘルプ 変更を保存できませんで

ファイル

- bin
- boot
- content
 - code
 - introduction.ipynb
 - data
 - drive
 - datalab
 - web
 - run.sh
- dev
- etc
- home
- lib
- lib32
- lib64
- media
- mnt
- opt

Colaboratory とは

Colaboratory (略称: Colab) は、

- 環境構築が不要
- GPU への無料アクセス

Drive をマウントをクリック

はじめに

ご覧になっているこのドキュメントは静的なウェブページです。

Colaboratory へようこそ
ファイル 編集 表示 挿入 ランタイム ツール ヘルプ 変更を保存できませんでした

+ コード + テキスト | ドライブにコピー

ファイル

- bin
- boot
- content
 - code
 - introduction.ipynb
 - data
 - drive
 - datalab
 - web
 - run.sh
- dev
- etc
- home
- lib
- lib32
- lib64
- media

Colaboratory とは

Colaboratory (略称: Colab) は、ブラウザから Python

- 環境構築が不要
- GPU への無料アクセス
- 簡単に共有

Colab は、学生からデータサイエンティスト、AI リサー

ご覧ください。下のリンクからすぐに使ってみることもあ

```
from google.colab import drive
drive.mount('/content/drive')
```

Google ドライブをマウントするには、このセルを実行してください。

閉じる

はじめに

ご覧になっているこのドキュメントは静的なウェブページです。

セルが現れるので、実行

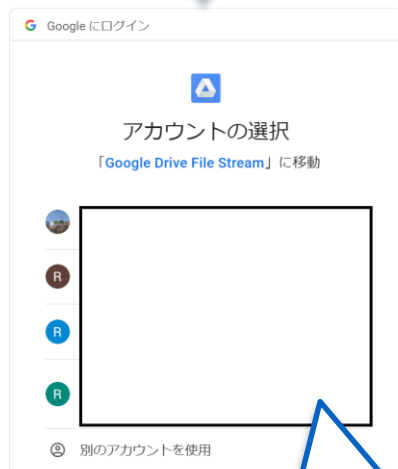
Google Driveとの接続

```
from google.colab import drive
drive.mount('/content/drive')
```

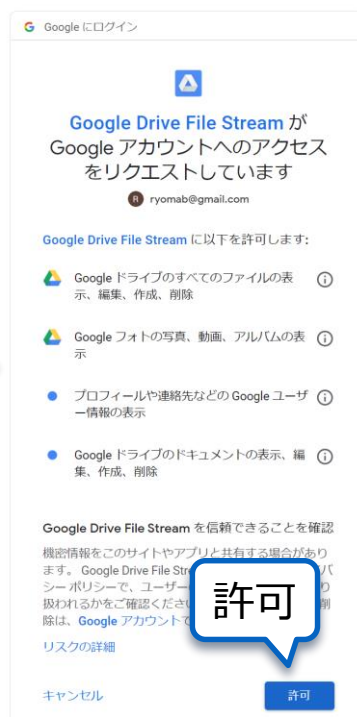
... Go to this URL in a browser: <https://accounts.google.com/o/oauth2/auth?c>

Enter your authorization code:

リンクをクリック



Google Driveの
アカウントを選択



許可



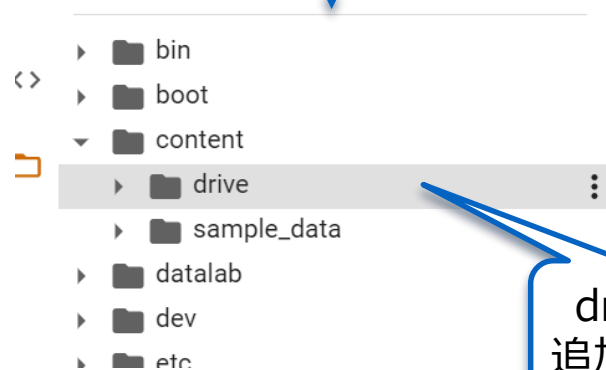
コピー

```
from google.colab import drive
drive.mount('/content/drive')
```

... Go to this URL in a browser: <https://accounts.google.com/o/oauth2/auth?c>

Enter your authorization code:

ペースト



driveが
追加された

コンピュータに指示を出す

Python で
あそんでみよう

大事な「おまじない」

- 今後, jupyter を起動したらまずはじめに必ずやること！
- `import numpy as np`
と打って enter キーを押す

```
In [2]: import numpy as np
```

```
In [3]:
```

コンピュータに指示を出す

文字の表示

お約束のHello World !

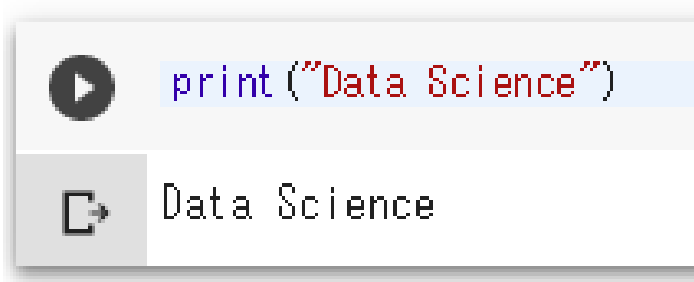
Hello World

- Cellで
Print(“Hello World”) と入力
- ▶キーを押す（実行）



練習 1 : 文字を変えてみよう！

- コンソールのところで
- 好きな言葉を“ ”に入れて表示させよう
print(“ ”) と入力
- return キーを押す（実行）



コンピュータに指示を出す

四則演算

たす, ひく, かける, わる

足し算

- +

- 例

```
In [1]: 1+2  
Out[1]: 3
```

引き算

- -

- 例

```
In [2]: 5-2
```

```
Out[2]: 3
```

掛け算

- * (アスタリスク) を使う

- 例

```
In [120]: 5*6
```

```
Out[120]: 30
```

```
In [121]: -3*2
```

```
Out[121]: -6
```

割り算

- / (スラッシュ) を使う

- 例

```
In [3]: 5/2  
Out[3]: 2.5
```

```
In [4]: 4/2  
Out[4]: 2
```

括弧

- $()$ 内を優先的に計算
- $(1-4)*5$

四則演算

- 足し算： $+$
 - $1+2$
- 引き算： $-$
 - $5-2$
- 掛け算： $*$ (アスタリスク)
 - $5*6$
- 割り算： $/$ (スラッシュ)
 - $5/2$
- 括弧： $()$ 内を優先的に計算
 - $(1-4)*5$

練習 2 : 四則演算

- 小学生の問題)

- A君が八百屋で100円のリンゴを5個と400円のイチゴを買いました。
- 合計にいくらになるかプログラムで計算させてみよう！

- BMI をプログラムで計算してみよう

ヒント : BMI は $\text{体重} \div (\text{身長} \times \text{身長})$


※体重は kg, 身長は m

変数

変数 (1)

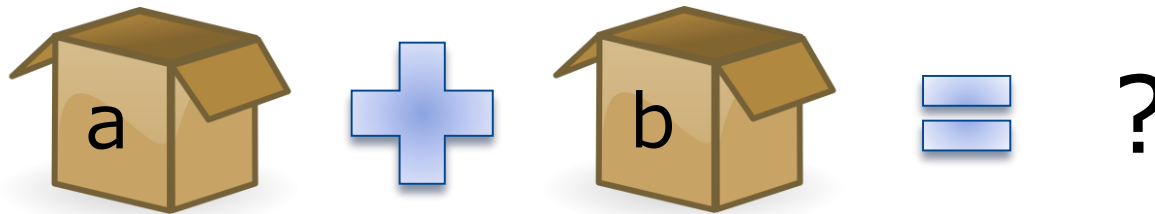
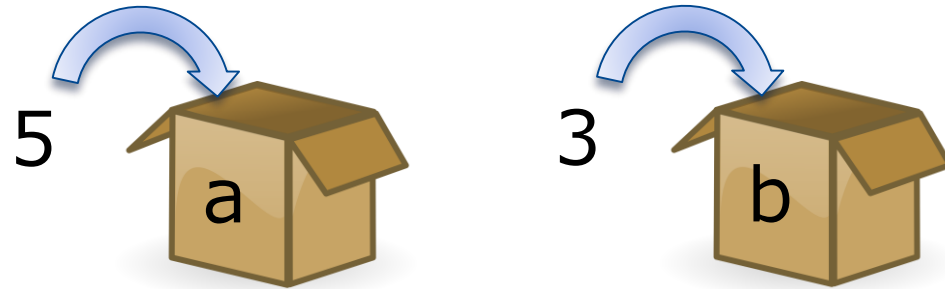
- 変数：好きな数を保存する「箱」



- =を使って保存
(左側が「箱の名前」, 右側が好きな数)
- a=5
- b="Monday"  実は文字列でも OK !

変数：演算（2）

- 変数同士の演算
- $a=5$
- $b=3$



```
In [293]: a + b  
Out[293]: 8
```

```
In [294]: a - b  
Out[294]: 2
```

変数 : 代入(2)

- 変数同士の演算
- $a=5, b=3$



$a = b$ とすると、
a の中身を捨てて、
b の中身をa に入れる

```
In [312]: a = b
```

```
In [313]: a  
Out[313]: 3
```

変数 (3)

- 変数の中身を変える

- $a=5$

 $a=20$

- $b=3$

 $b=10$

直接数字を書いて演算すると、毎回書き換える必要がある

```
In [301]: 5 + 3
```

```
Out[301]: 8
```

```
In [302]: 20 + 10
```

```
Out[302]: 30
```

変数で演算すると、同じ式でO.K.

```
In [293]: a + b
```

```
Out[293]: 8
```

```
In [299]: a + b
```

```
Out[299]: 30
```

変数 (4)

- データ 1

$a=5, b=4, c=1, d=6, e=29, f=12$

$(5+4)*(1+6)/29 + 12*5*4*(6-1)$

- データ 2

$a=8, b=3, c=2, d=7$

$(8+3)*(2+7)/20 + 1$

毎回すべて書き換える？

プログラムの大きなメリット！
同じ命令文が違うデータに
使える

式で記述すれば、中身を変えるだけでO.K.

- $(a+b)*(c+d)/e + f*a*b*(d-c)$

変数 (5) : 名前の付け方

- 例 :

```
name="John"
```

```
age=30
```

```
weight=56
```

```
height=1.68
```

```
BMI=weight/(height*height)
```

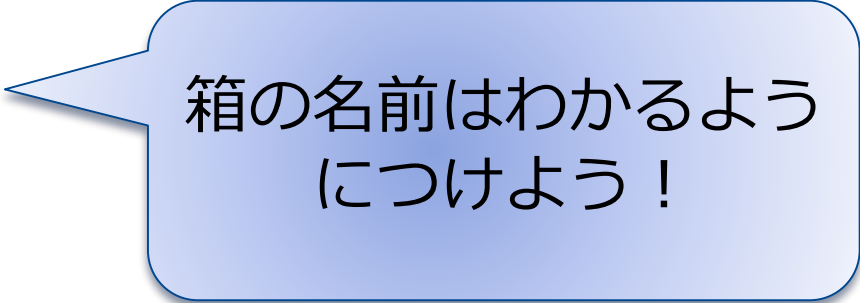
```
print( name )
```

```
print( age )
```

```
print( weight )
```

```
print( height )
```

```
print( BMI )
```



箱の名前はわかるように
につけよう！

変数 (6)

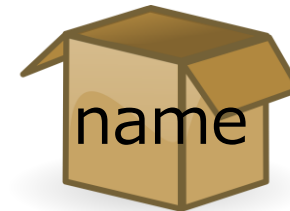
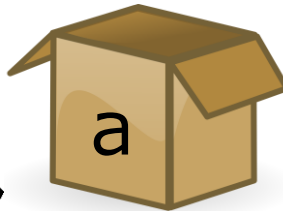
- 良くない例 :

a="John"

b=30

c=56

d=1.68



- えっと BMI の計算式は . . . $e = c / (d * d)$

アドバイス : 変数名はなんでもいいけど,
できる限り意味がわかりやすいように用意する

練習 3 : 変数

- 変数を使って、BMI をプログラムで計算してみよう

ヒント : BMI は $\text{体重} \div (\text{身長} \times \text{身長})$

weight = 自分の体重

height = 自分の身長

※体重は kg, 身長は m

コンピュータに指示を出す

関数

「材料」と「道具」(1)

材料 (変数)



道具 (関数)

切る



皮をむく



煮る



print や
+や- も
道具の一種



関数(1)

例)

- 平方根 ($\sqrt{\quad}$) を計算する関数 (道具)
 - sqrt (平方根は、英語で'square root'でその略)
- 道具を使うためのおまじない
 - import math

```
In [319]: math.sqrt(4)  
Out[319]: 2.0
```

```
In [320]: math.sqrt(9)  
Out[320]: 3.0
```

ライブラリ
道具はまとめて整理されている
調理する道具は台所、
トンカチ等の工具は工具箱
mathにsqrtがある
仕舞っている処から出して、
道具を使える状態にしている

関数(2)

- 他にもいろいろなライブラリ、関数がある
- import numpy
- numpy.round(3.8)
 - 小数点以下を四捨五入

```
In [328]: numpy.round(3.8)  
Out[328]: 4.0
```

- 毎回ライブラリ名をフルネームで書くのは面倒：短縮しちゃえ！
- import numpy as np

```
In [330]: np.round(3.8)  
Out[330]: 4.0
```

練習 4 : 関数の利用

- 関数を使って計算しよう！
 - 81の平方根を計算しよう
 - 256の平方根を計算しよう
 - 9.4を四捨五入しよう
 - 9.7を四捨五入しよう
- 自分で関数をWebで調べて使ってみよう！
 - 絶対値を求める関数を調べてみよう
 - -5の絶対値を関数を使って計算しよう

ヒント) Python 絶対値
でGoogleで検索してみよう

基本注意事項

※知ってる人は遠慮なく飛ばしてください

全角文字，日本語入力を避ける

- 日本語入力するとエラーが出る

a + b
全角スペース

```
In [314]: a + b
File "<ipython-input-314-c469e43fc799>", line 1
  a + b
    ^
SyntaxError: invalid syntax
```

全角文字，日本語入力を避ける（２）

- できる限り全角文字，日本語は使わない
 - 全角の空白（スペース）に注意！
（エラーになる＆見つけづらい）
 - ファイル名も半角英数字にしよう
OK：shincho_taiju.xlsx, height_weight.csv
NG：身長体重データ.csv
- 半角カタカナ（アイウ）とかもやめよう

コメントアウト

- メモ書き
- 書いたけど一時的に実行したくない場合
- 行頭に # で一行コメントアウト
- ''' または """" で囲んだ部分をコメントアウト

演習資料

Playing with the Body-mass index

演習1：BMI

- 変数を使ってBMIを計算するコードを書いてみよう

ヒント：BMI は $\text{体重} \div (\text{身長} \times \text{身長})$

※体重は kg, 身長は m

- 変数の値を変えて、それぞれ計算してみよう
 - A君：1.7m, 62kg
 - B君：1.6m, 75kg
 - C君：1.8m, 73kg

演習2：関数の呼び出し

- 求めたBMIを四捨五入してみよう
 - ヒント：round
- 求めたBMIを切り捨ててみよう
 - ヒント：「python 切り捨て」でGoogleで検索
- 求めたBMIを切り上げてみよう
- 変数の値を変えて、それぞれ計算してみよう
 - A君：1.7m, 62kg
 - B君：1.6m, 75kg
 - C君：1.8m, 73kg

演習：進みが早い人用

演習3：print文の応用

- 変数aに5を代入して、
- print文で「aの中身は5です。」と表示してみよう。
 - ヒント) 「print format python」で検索

演習4：print文の応用

- 演習3で作成したエディタファイルに以下のprint文を追加してまとめて実行してみよう。
 - 「A君のBMIはxxxです。」と表示
 - 「B君のBMIはxxxです。」と表示
 - 「C君のBMIはxxxです。」と表示
 - ※) xxxの部分はそれぞれ計算したBMIが入る