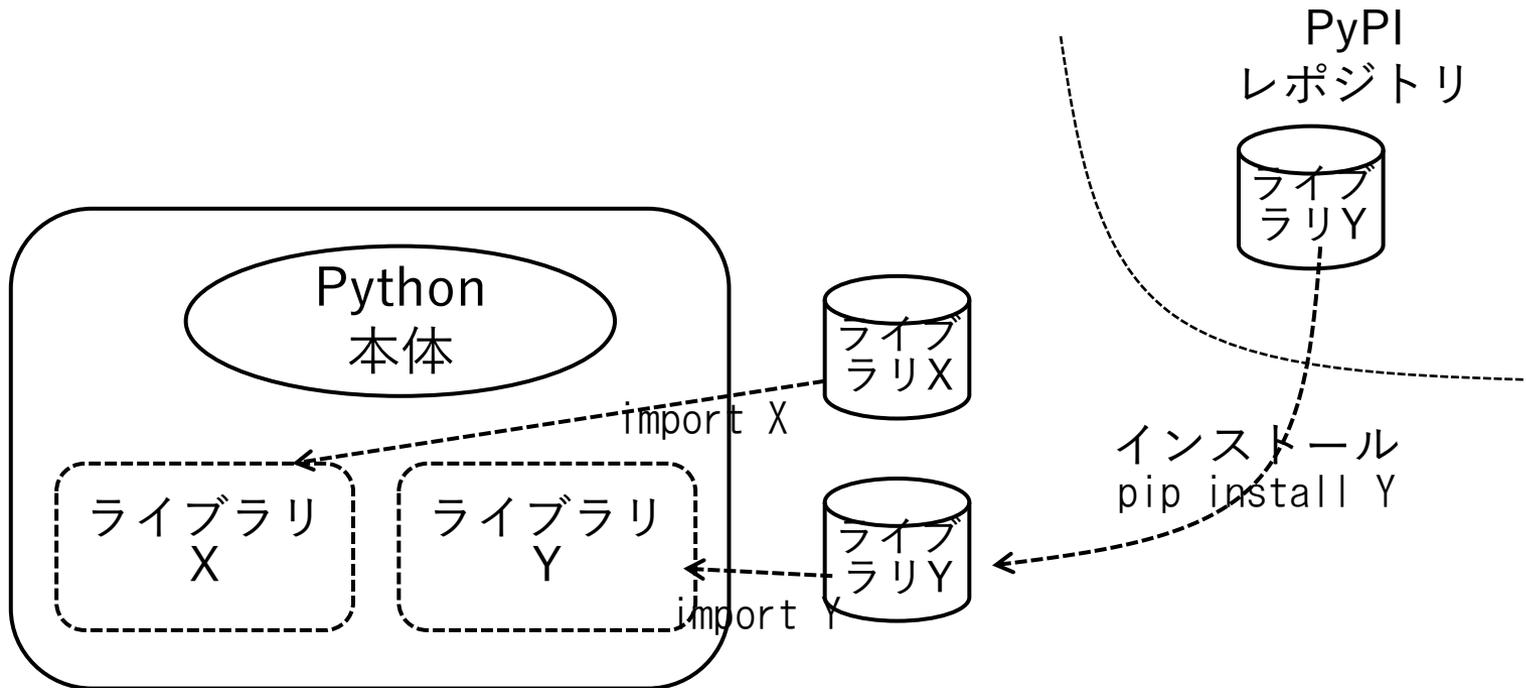


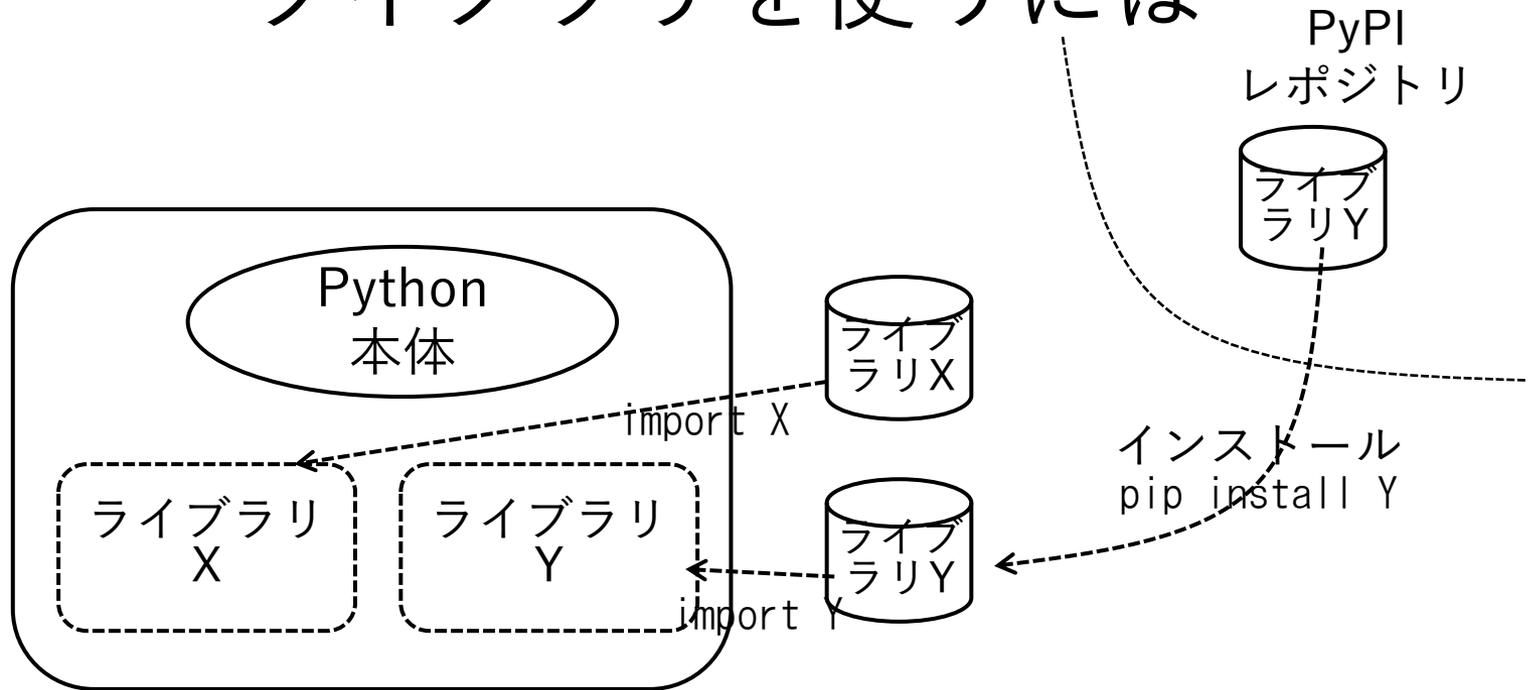
ライブラリとは



- 必要に応じて、機能を取り込む
- なぜ予め全部を入れておかないのか？
 - 無駄に大きくしない
 - 追加が容易 ⇒ 様々な人たちが作って登録 (PyPI)

<https://pypi.org/>

ライブラリを使うには



- 標準ライブラリはインストール時に付いてくる
⇒ そのまま次へ
- PyPIサイトからダウンロードするもの
⇒ pip install **パッケージ名**

• プログラム内で
import ライブラリ名

← コマンドレベルで行う
実習環境ではダウンロード済み

ライブラリ～どんなものが

標準ライブラリ

<https://docs.python.jp/3/library/index.html>

- 必要に応じてimport
- 例：
 - math 数学関数
math.log(...) math.sin(...)
 - random 乱数発生
random.uniform(...)

PyPIからインストール

<https://pypi.org>

- pipの後、import
- 例：
 - numpy
 - 数値計算・高速化
 - ベクトル・行列の
効率よい計算
 - pandas
 - データ解析を行うため
 - 表形式・時系列データ
 - matplotlib
 - グラフを生成表示

標準ライブラリの使用例

```
import math
for x in [1, 2, 3, 4]:
    print(math.log10(x))
```



```
0. 0
0. 3010299956639812
0. 47712125471966244
0. 6020599913279624
```

```
import random
for x in range(5):
    print(random.random())
```



```
0. 5485815956018213
0. 8381522944622031
0. 45988254932904127
0. 061798471715748704
0. 28332411370580723
```

```
for x in range(5):
    print(random.randint(0, 10))
```



```
1 2 6 0 7
```

試してみよう

PyPIから取り込むパッケージの使用例

あらかじめ

```
pip install <パッケージ名>
```

あとは、同様に

```
import <ライブラリ名>
```

```
import numpy as np  
a = np.array([[1, 2], [3, 4]])  
b = np.array([[5, 6], [7, 8]])  
print(a*b)      ⇒ 要素ごと積
```

↓

```
[[ 5 12]  
 [21 32]]
```

```
print(a.dot(b)) ⇒ 内積
```

↓

```
[[19 22]  
 [43 50]]
```

```
import pandas as pd  
a = pd.DataFrame(  
    [[165.0, 65.0], [170.0, 68.0]],  
    columns=['身長', '体重'],  
    index=['田中', '山田'])  
print(a)
```

↓

| | 身長 | 体重 | |
|----|-------|------|---------|
| 田中 | 165.0 | 65.0 | ← Excel |
| 山田 | 170.0 | 68.0 | 類似 |

の表

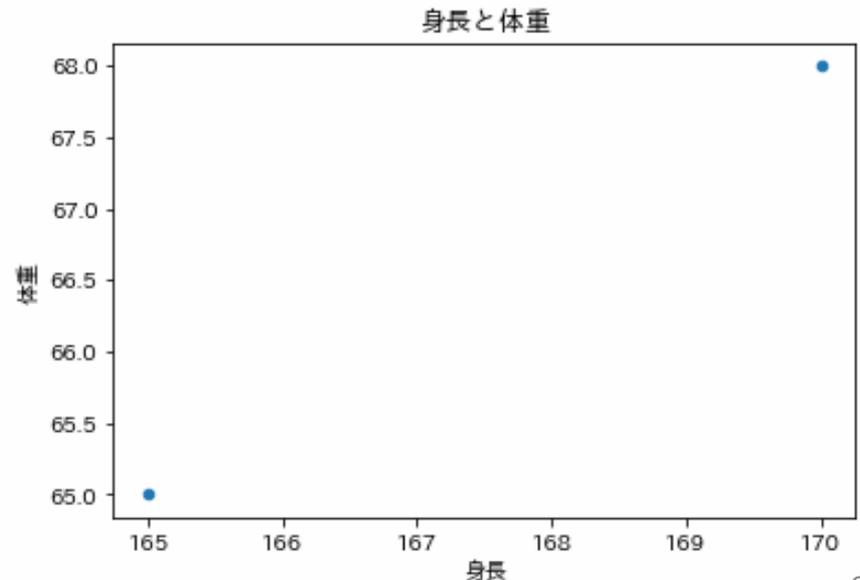
試してみよう

PyPIから取り込むパッケージの使用例

```
%matplotlib inline
import pandas as pd
import matplotlib.pyplot as plt
a = pd.DataFrame(
    [[165.0, 65.0], [170.0, 68.0]],
    columns=['身長', '体重'],
    index=['田中', '山田'])
a.plot.scatter(x='身長', y='体重')
plt.title('身長と体重')
plt.xlabel('身長')
plt.ylabel('体重')
plt.show()
```

← 画面に表示させる
おまじない

a.plot.scatter(x='身長', y='体重') ← 散布図を描く



時間があれば
試してみよう

(おまけ)

importの書き方

単純なimport

```
import math
...
y = math.log10(x)
...
```

名前の読替え

```
import pandas as pd
...
a = pd.DataFrame(...)
...
```

```
import pandas
...
a = pandas.DataFrame(...)
...
```

中身の呼び出し

matplotlib

pyplot

```
import matplotlib.pyplot
...
matplotlib.pyplot.show()
```

```
from matplotlib import pyplot
...
pyplot.show()
```

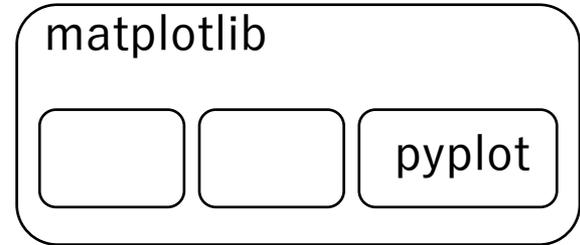
```
import matplotlib.pyplot as plt
...
plt.show()
```

```
from matplotlib import pyplot as plt
...
plt.show()
```

(おまけ) importの書き方

```
from matplotlib import pyplot
...
pyplot.show()
```

matplotlibからpyplotだけをインポート



```
from matplotlib import *
...
pyplot.show()
```

matplotlibからすべてをインポート
不要なものまでインポートするので
余分なスペースを占有する

```
from matplotlib.pyplot import *
...
show()
```

matplotlib.pyplot からすべてをインポート
スペース的には左上と同じ
名前付けの指す範囲が違う
もし、他のモジュールがshowという
名前を持っていればバッティングする

遊んでみよう

```
%matplotlib inline
import pandas as pd
import matplotlib.pyplot as plt
```

← 画面に表示させる
おまじない

プロットするデータをaに作る

```
a = pd.DataFrame(index=[1, 2, 3, 4, 5], columns=['x', 'y'])
for i in [1, 2, 3, 4, 5]:
    a.at[i, 'x'] = i
    a.at[i, 'y'] = i**2
print(a)
```

```
a.plot.scatter(x='身長', y='体重') ← 散布図を描く
plt.title('身長と体重')
plt.xlabel('身長')
plt.ylabel('体重')
plt.show()
```

aのカラム名に直す

この辺は適宜書き直す

試してみよう

(時間があれば) 遊んでみよう

```
%matplotlib inline
import pandas as pd
import matplotlib.pyplot as plt
```

← 画面に表示させる
おまじない

プロットするデータをaに作る

```
ix = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
a = pd.DataFrame(index=ix, columns=['x', 'y'])
for i in ix:
    a.at[i, 'x'] = i
    a.at[i, 'y'] = i**2
print(a)
```

```
a.plot.scatter(x='身長', y='体重') ← 散布図を描く
```

```
plt.title('身長と体重')
```

```
plt.xlabel('身長')
```

```
plt.ylabel('体重')
```

```
plt.show()
```

aのカラム名に直す

この辺は適宜書き直す

時間があれば
試してみよう

(時間があれば) 遊んでみよう

```
%matplotlib inline
import pandas as pd
import matplotlib.pyplot as plt
import math
```

← 画面に表示させる
おまじない

プロットするデータをaに作る

```
ix = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
a = pd.DataFrame(index=ix, columns=['x', 'y'])
for i in ix:
    a.at[i, 'x'] = i
    a.at[i, 'y'] = i**2
print(a)
```

```
a.plot.scatter(x='身長', y='体重') ← 散布図を描く
```

```
plt.title('身長と体重')
```

```
plt.xlabel('身長')
```

```
plt.ylabel('体重')
```

```
plt.show()
```

aのカラム名に直す

この辺は適宜書き直す

時間があれば
試してみよう

TAチェック (スライド9ページ)

```
%matplotlib inline
import pandas as pd
import matplotlib.pyplot as plt
```

← 画面に表示させる
おまじない

プロットするデータをaに作る

```
a = pd.DataFrame(index=[1, 2, 3, 4, 5], columns=['身長', '体重'])
for i in [1, 2, 3, 4, 5]:
    a.at[i, '身長'] = i
    a.at[i, '体重'] = i**2
print(a)
```

```
a.plot.scatter(x='身長', y='体重')
plt.title('身長と体重')
plt.xlabel('身長')
plt.ylabel('体重')
plt.show()
```

← 散布図を描く

aのカラム名に直す

この辺は適宜書き直す

TAの人に、動作をチェックしてもらうこと