

Сравнение Python и Ruby

Масляков Глеб Олегович

ВМК МГУ

- 1 Вступление
- 2 Синтаксис
- 3 Machinelearning
- 4 Общие измышления
- 5 Итоги

1 Вступление

2 Синтаксис

3 Machinelearning

4 Общие измышления

5 Итоги

- Python
 - Высокого уровня
 - Объектно-ориентированный
 - Строгая динамическая типизация
- Ruby
 - Высокого уровня
 - Объектно-ориентированный
 - Строгая динамическая типизация

- Python
 - Web-разработка
 - Игры
 - Научные исследования
 - Desktopные приложения
- Ruby
 - Web-разработка
 - Научные исследования

1 Вступление

2 Синтаксис

3 Machinelearning

4 Общие измышления

5 Итоги

- Python

```
1     def func(first, second):  
2         return first+second  
3     func(first=1, second=2)
```

- Ruby

```
1     def func(kargs = {})  
2         kargs[ :first] + kargs[ :second]  
3     end  
4     func(first: 1, second: 2)
```

- Python

```
1     @threadsafe
2     def func():
3         print("decorated")
```

- Ruby

```
1     class Bar
2         extend Threadsafe
3         def func()
4             print("decorated")
5         end
6         threadsafe :func
7     end
```

- Python

```
1 "My_name_is_{name}".format(name=name) "
```

- Ruby

```
1 "My_name_is_#{name}"
```

- Python
 - Просто нет
- Ruby

```
1     case language
2         when "pascal" then print("Top")
3         when "ruby" then print("Not bad")
4         when "python" then print("Good")
5         else print("Bad")
6     end
```

1 Вступление

2 Синтаксис

3 Machinelearning

4 Общие измышления

5 Итоги

- Python

```
1 import numpy
2 n = np.ones((2000,2000), dtype=int)
3 start=time.clock()
4 n.dot(n)
5 finish=start.clock()
6 print(start-finish)
7 >> 23.87994
```

- Ruby

```
1 require "nmatrix"
2 n = NMatrix.new([2000,2000], [1])
3 start = Time.now.to_f
4 n.dot(n)
5 finish = Time.now.to_f
6 print(start - finish)
7 >>5.02431
```

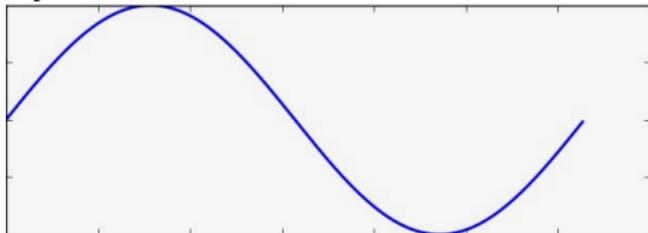
- Python

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3 x = np.arange(360) * np.pi / 180
4 plt.subplot(212)
5 plt.plot(x, np.sin(x), lw=2)
6 plt.show()
```

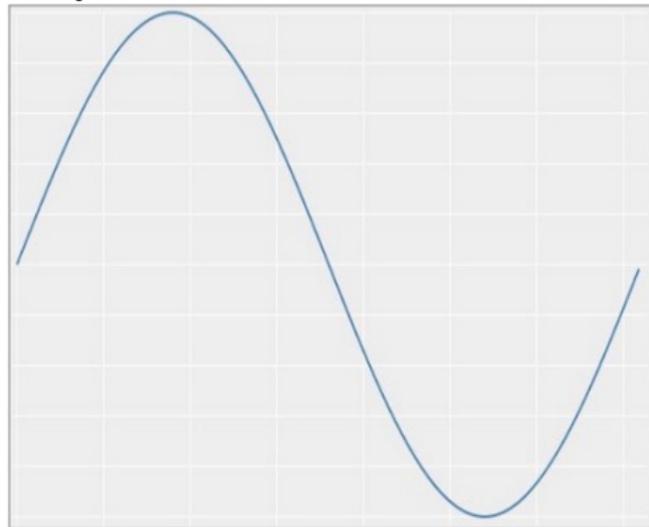
- Ruby

```
1 require "nyaplot"
2 x = Array.new(360) {|i| i}
3 y = x.map {|i| Math::sin(i*Math::pi/180)}
4 p = Nyaplot::Plot
5 p.add(:line, x, y)
6 p.show
```

Python



Ruby



- Python

```
1 import pandas
2 df = pandas.DataFrame({'a': [1,2,3,4,5], 'b': ←
    [0,1,2,3,4]})
3 print(df.a[0], df['b'][1])
4 print(df.a.mean, df.b.max)
5 >> 1 0
6 >>3 4
```

- Ruby

```
1 require "daru"
2 df = Daru::DataFrame({a: [1,2,3,4,5], b: ←
    Array.new(5){|i| i}})
3 puts("#{df.a[0]}_#{df['b'][1]}")
4 puts("#{df.a.mean}_#{df.b.max}")
5 >>1 0
6 >>3 4
```

- Python

```
1 import sklearn
```

- Ruby

```
1 require "classifier-reborn"  
2 require "decisiontree"  
3 require "knn"  
4 require "kdtree"  
5 ...
```

- 1 Вступление
- 2 Синтаксис
- 3 Machinelearning
- 4 Общие измышления**
- 5 Итоги

- Python

- + Прекрасная документация
- + Мощные средства для анализа данных
- + Удобный синтаксис
- + Ipython

- Ruby

- ++ Удобный синтаксис
- + Есть аналоги питоновских библиотек
- + Iruby
 - Отвратительная документация
 - Проблемы с разными версиями языка

- 1 Вступление
- 2 Синтаксис
- 3 Machinelearning
- 4 Общие измышления
- 5 Итоги**

- Python и Ruby похожи во всём
- Различия между ними гораздо меньше, чем их противопоставление другим языкам
- Давайте дружить, а не выяснять, кто круче.