LẬP TRÌNH PYTHON

PiMA 2019 - Python cơ bản

Phan Ngọc Tiên

York University, Toronto, Canada

Mục lục

Giới thiệu về Python

Giới thiệu về Python

- Python là ngôn ngữ lập trình bậc cao, ra mắt lần đầu vào năm 1991
- Python là ngôn ngữ lập trình đơn giản, cú pháp (syntax) đơn giản, dễ đọc, rất gần với ngôn ngữ tự nhiên
- Hỗ trợ lập trình hướng cấu trúc, lập trình hướng cấu trúc và lập trình hàm (yếu)

Basic Input/Output (Nhập xuất cơ bản)

```
# Input (Nhập)
x = input()

# Output (Xuất)
print(x)
```

Các phép toán

- Python hỗ trợ các phép toán $+, -, *, /$ (chia lấy kết quả float), // (chia lấy phần nguyên, kết quả int), $**$ (lên lũy thừa).
- Các phép biến đổi bit: & (AND), | (OR), ^ (XOR).
- Đối với biến logic có **and, or, not**.
- Các phép toán so sánh $<$ (nhỏ hơn), $>$ (lớn hơn), $<=$ (nhỏ hơn hoặc bằng), $>=$ (lớn hơn hoặc bằng), $==$ (bằng)

Các kiểu dữ liệu

- Python tạo kiểu động (dynamically typed)

```
# Số nguyên (Integer)
x = 20
# Số thực (Float)
y = 17.5
# Số phức (Complex)
z = 20 + 17j
```

```
#include <iostream>
#include <complex>
using namespace std;

int main() {
int x = 20;
double y = 17.5;
complex<double> z4 = 1.5 + 2i;
}
```

Các kiểu dữ liệu: Boolean

```
# Boolean
x = True
y = False
```

```
#include <iostream>
#include <complex>
using namespace std;

int main() {
bool x = true;
bool y = false;
}
```

Các kiểu dữ liệu: String (Chuỗi)

- Python

```
# String (Chuỗi)
x = "PiMA 2019 'Deep Learning'"
```

- C++

```
#include <iostream>
#include <string>
using namespace std;

int main() {
  string s = "PiMA 2019 'Deep Learning'";
}
```

Các kiểu dữ liệu: List (Danh sách)

- Python

```
x = [1, 2, 3, 4, 5]
# List can be nested (danh sách có thể lồng vào nhau)
x = [[1, 2, 3], [4, 5, 6]]

## Có thể có nhiều kiểu dữ liệu trong cùng 1 danh sách
x = [[3.14, 2], 3 + 4j, [5, 6]]
```

- C++

```
#include <iostream>
#include <vector>
using namespace std;
int main() {
  int arr[] = {1, 2, 3, 4, 5, 6};
  vector <double> a {3.14, 2.71, 2.11};
}
```

Các kiểu dữ liệu: Dictionary (Từ Điển)

```
x = {"P": "Project",
     "i": "in",
     "M": "Mathematics",
     "A": "Applications"}
```

Các kiểu dữ liệu: Tuple

```
point = (1, 2, 3)

# Lấy ra giá trị từ tuple
x, y, z = point
```

Các kiểu dữ liệu: Tuple

```
point = (1, 2, 3)

# Lấy ra giá trị từ tuple
x, y, z = point
```

Tuple vs List: điểm giống và khác nhau giữa List và Tuple là gì?

Python cơ bản

Refer to notebook.

Sections

Sections group slides of the same topic

```
\section{Elements}
```

for which **metropolis** provides a nice progress indicator ...

Titleformats

Metropolis titleformats

metropolis supports 4 different titleformats:

- Regular
- Smallcaps
- allsmallcaps
- ALLCAPS

They can either be set at once for every title type or individually.

Small caps

This frame uses the `smallcaps` titleformat.

Potential Problems

Be aware, that not every font supports small caps. If for example you typeset your presentation with pdfTeX and the Computer Modern Sans Serif font, every text in smallcaps will be typeset with the Computer Modern Serif font instead.

all small caps

This frame uses the `allsmallcaps` titleformat.

Potential problems

As this titleformat also uses smallcaps you face the same problems as with the `smallcaps` titleformat. Additionally this format can cause some other problems. Please refer to the documentation if you consider using it.

As a rule of thumb: Just use it for plaintext-only titles.

ALL CAPS

This frame uses the `allcaps` titleformat.

Potential Problems

This titleformat is not as problematic as the `allsmallcaps` format, but basically suffers from the same deficiencies. So please have a look at the documentation if you want to use it.

Elements

Typography

```
The theme provides sensible defaults to
        \emph{emphasize} text, \alert{accent} parts
        or show \textbf{bold} results.
```

becomes

The theme provides sensible defaults to *emphasize* text, accent parts or show **bold** results.

Font feature test

- Regular
- *Italic*
- SmallCaps
- **Bold**
- ***Bold Italic***
- **Bold SmallCaps**
- `Monospace`
- *`Monospace Italic`*
- **`Monospace Bold`**
- ***`Monospace Bold Italic`***

Lists

Items

- Milk
- Eggs
- Potatos

Enumerations

1. First,
2. Second and
3. Last.

Descriptions

PowerPoint Meeh.

Beamer Yeeeha.

Animation

- This is important

Animation

- This is important
- Now this

Animation

- This is important
- Now this
- And now this

Animation

- This is really important
- Now this
- And now this

Figures

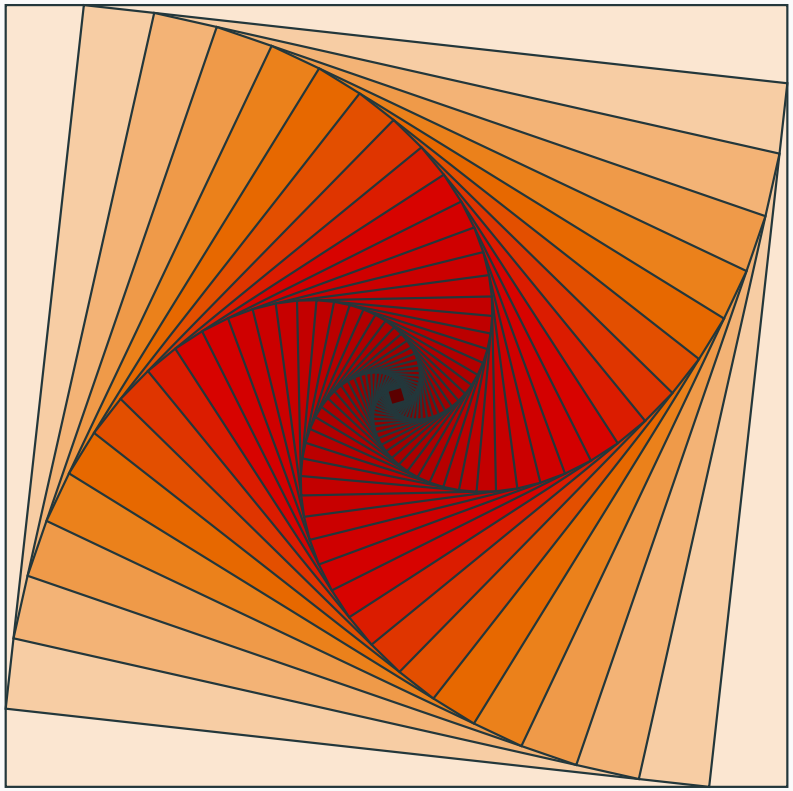


Figure 1: Rotated square from texample.net.

Tables

Table 1: Largest cities in the world (source: Wikipedia)

City	Population
Mexico City	20,116,842
Shanghai	19,210,000
Peking	15,796,450
Istanbul	14,160,467

Blocks

Three different block environments are pre-defined and may be styled with an optional background color.

Default
Block content.

Alert
Block content.

Example
Block content.

Default

Block content.

Alert

Block content.

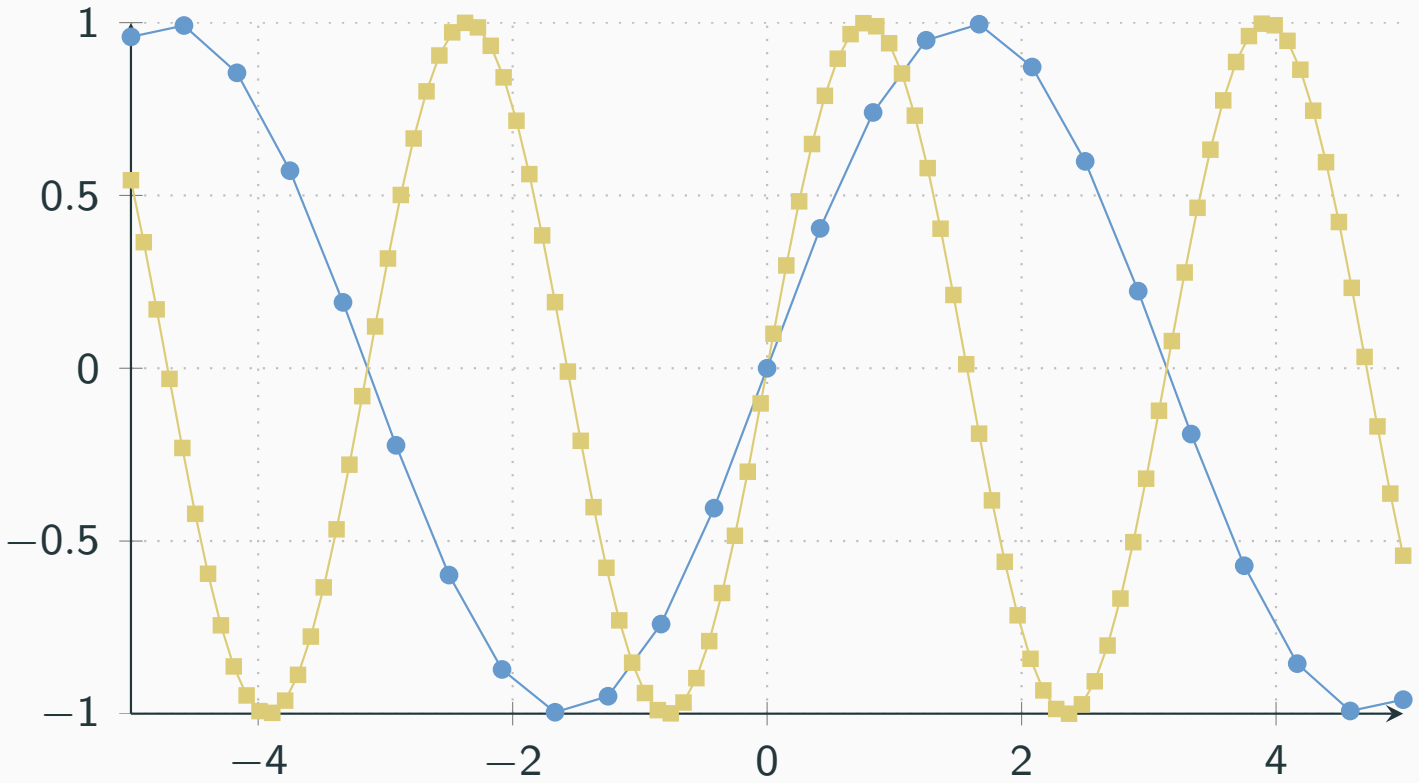
Example

Block content.

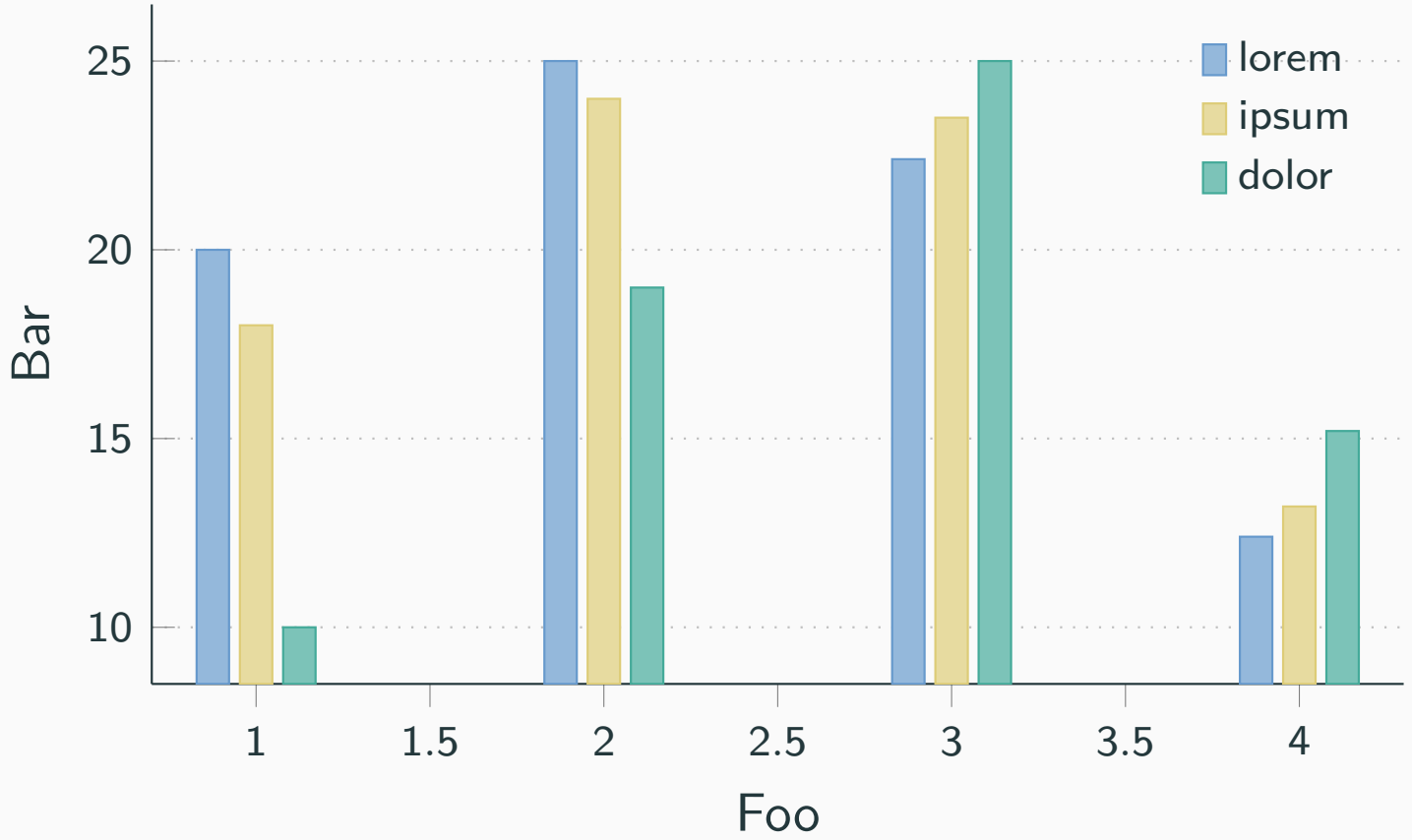
Math

$$e = \lim_{n \to \infty} \left(1 + \frac{1}{n}\right)^n$$

Line plots



Bar charts



Quotes

Veni, Vidi, Vici

Frame footer

metropolis defines a custom beamer template to add a text to the footer. It can be set via

```
\setbeamertemplate{frame footer}{My custom footer}
```

References

Some references to showcase [allowframebreaks] [4, 2, 5, 1, 3]

Conclusion

Summary

Get the source of this theme and the demo presentation from

`github.com/matze/mtheme`

Questions?

Backup slides

Sometimes, it is useful to add slides at the end of your presentation to refer to during audience questions.

The best way to do this is to include the `appendixnumberbeamer` package in your preamble and call `\appendix` before your backup slides.

metropolis will automatically turn off slide numbering and progress bars for slides in the appendix.

References i

P. Erdős.

A selection of problems and results in combinatorics.

In *Recent trends in combinatorics (Matrahaza, 1995)*, pages 1–6. Cambridge Univ. Press, Cambridge, 1995.

R. Graham, D. Knuth, and O. Patashnik.

***Concrete mathematics*.**

Addison-Wesley, Reading, MA, 1989.

G. D. Greenwade.

The Comprehensive Tex Archive Network (CTAN).

TUGBoat, 14(3):342–351, 1993.

D. Knuth.

Two notes on notation.

Amer. Math. Monthly, 99:403–422, 1992.

References ii


H. Simpson.
Proof of the Riemann Hypothesis.
preprint (2003), available at
`http://www.math.drofnats.edu/riemann.ps`, 2003.