

# Python Knižnice

## Spracovanie a Vizualizácia Dát



# Ako Začneme?

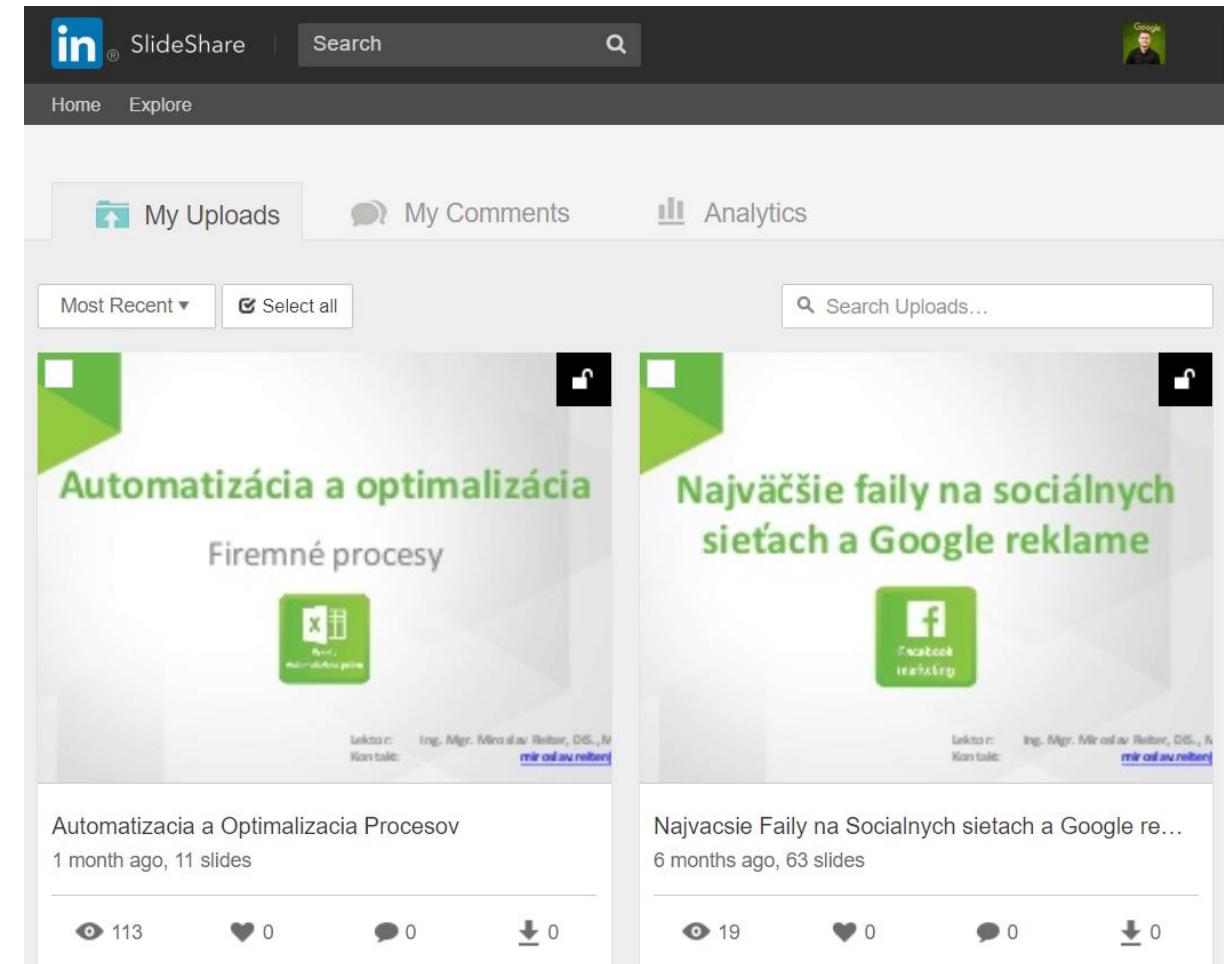
## 1. Stiahnite si Cvičný Súbor

- [https://github.com/miroslav-reiter/Kurzy\\_SAV\\_Analytika\\_Python\\_R](https://github.com/miroslav-reiter/Kurzy_SAV_Analytika_Python_R)

## 2. Pridajte si ma na LinkedIn

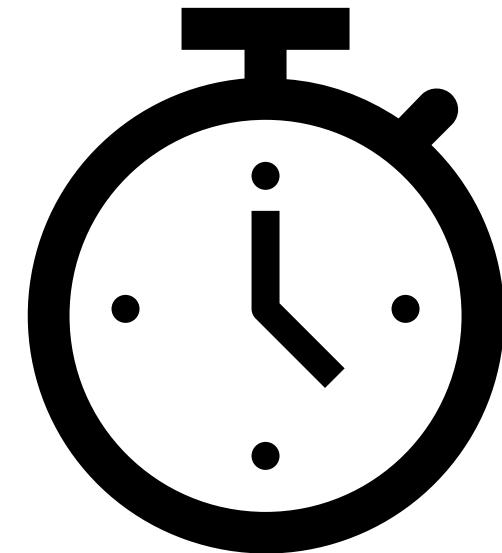
- [www.linkedin.com/in/miroslav-reiter/](http://www.linkedin.com/in/miroslav-reiter/)

## 3. Prezentácia a materiály po prednáške



# Úvodné informácie

- Časový rozvrh (9:00-13:30)
  - Prestávky
  - Mobilné telefóny a zariadenia
- 
- Priprav si otázky a rovno sa pýtaj
  - Interaktívna forma



# O lektorovi - Miroslav Reiter

10000+  
klientov a  
500+ firiem

Programátor  
Analytik  
Manažér

Google,  
Microsoft  
ISTQB tréner

122  
certifikácií

83 príručiek a  
publikácií

13 škôl

50+  
projektov

Vlastná firma



---

**MOTIVÁCIA**

# Študuje 5 odborov a absolvoval už 12 univerzít. Ako zvláda stres a manažovanie času?



Foto: Jakub Kovalík pre FMK UCM | Miroslav Reiter na prednáške Grow with Google na FMK UCM.

**Nikola Kotláriková**

19. júl 2022 · 8 min. čítania



# Miroslav Reiter



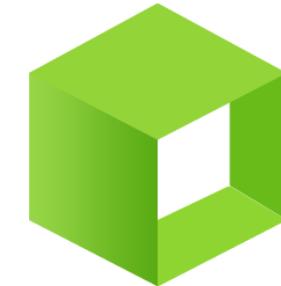
1. PhDr. VŠM (Podnikovný manažment)
2. Ing. STU FEI (**Aplikovaná informatika**)
3. Mgr. UK FM (**Strategický manažment a marketing**)
4. Mgr. VŠM (**Manažérstvo kvality**)
5. Mgr. VŠEMVŠ (Verejná správa)
6. Mgr. DTI (Učiteľstvo ekonomických predmetov)
7. DiS. AMOS (Cestovný ruch)
8. MBA LIGS (Executive management)
9. DBA Humanum (**IT manažment**)
10. MPA IES (Verejná správa a samospráva krajov)
11. MSc. Humanum (**Bezpečnosť informačných systémov**)
12. Ing. Paed. IGIP
13. Mgr. PEVŠ (**Bezpečnosť informačných systémov**)



DIGITÁLNA  
UNIVERZITA



FAKULTA MANAGEMENTU  
Univerzity Komenského  
v Bratislave



IT ACADEMY



# Vyberte si online kurz

Naučte sa programovať, tvoriť webstránky a grafiku, manažovať alebo sa zamerajte na osobný rozvoj. Všetko jednoducho vďaka našim online kurzom z pohodlia domova.

Ročné predplatné na všetky online kurzy

2299,99€

399,99€

Prístup pre Vás do všetkých aktuálnych aj pripravovaných online kurzov

12 mesačná platnosť

Kúpiť teraz



407 kurzov v ponuke



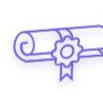
Zábavné online lekcie



Akreditované kurzy



11 rokov skúseností



Certifikovaní profesionálni lektori

Odporučame Kurzy špeciálne pre vás



Online kurz SAP I.  
Začiatočník  
224,00€ 292,50€



Online kurz Testovanie  
Softvéru I. Začiatočník  
140,00€ 183,30€



Online kurz Java I.  
Začiatočník  
67,00€ 88,40€



Online kurz PRINCE2  
Foundation  
224,00€ 308,70€



Online kurz Lektor  
(Akreditovaný Kurz Lektor)  
127,00€ 167,50€



Online kurz Copywriting I.  
Začiatočník  
50,00€ 66,30€

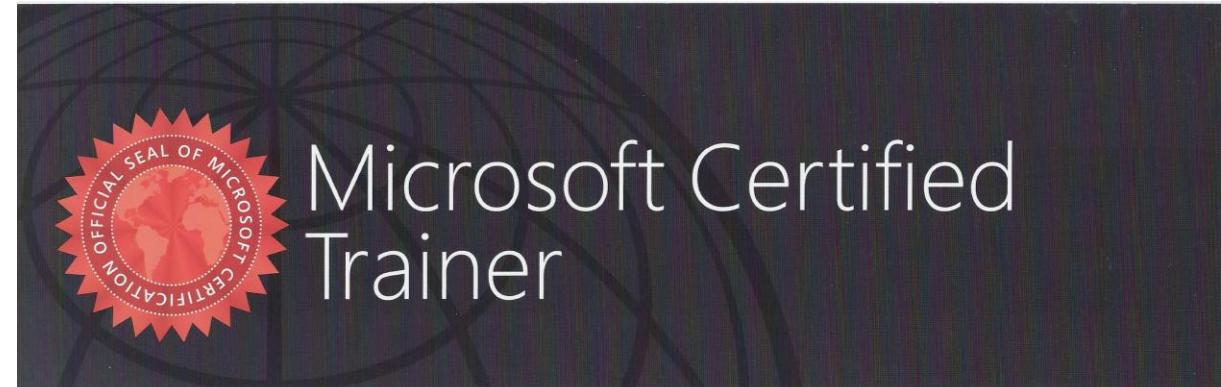
# Miroslav Reiter

získava status

## Google Certified Trainer

Automation

Google



Microsoft Certified  
Trainer

MIROSLAV REITER

Has successfully completed the requirements to be recognized as a Microsoft Certified Trainer

N. S.  
Satya Nadella  
Chief Executive Officer

Microsoft  
CERTIFIED  
Trainer



Luigi, Mário  
a Yoshi

# Čo robíte?

1. Študent/učiteľ

2. Zamestnanec

3. Podnikateľ

4. Nezamestnaný/materská

5. Dievča pre všetko



National competence centre for high performance computing  
SLOVAKIA

C  
EURO



**EuroHPC**  
Joint Undertaking





### Vzdelávanie

Kurzy:  
[itkurzy.sav.sk](https://itkurzy.sav.sk)



### Propagácia

Prednášky:  
[https://eurocc.nscc.sk  
/news/prednasky/](https://eurocc.nscc.sk/news/prednasky/)



### HPC služby

Prístup k  
výpočtovým  
prostriedkom



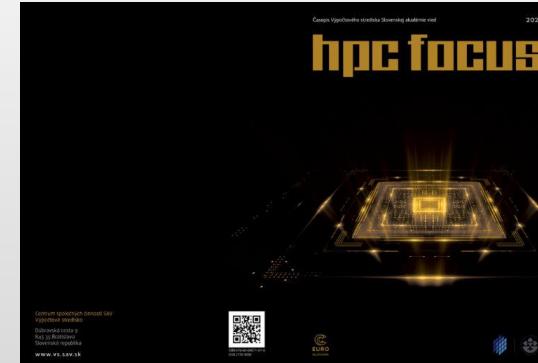
### Mapovanie HPC prostredia

Prieskum:  
[https://eurocc.nscc.sk  
/mapping-survey/](https://eurocc.nscc.sk/mapping-survey/)



### Spolupráca

Pilotné projekty  
Dlhodobá  
spolupráca



**Qubit**  
Conference

robíme it

Slovenská  
obchodná  
a priemyselná  
komora

### S kým spolupracujeme:

- Akademické inštitúcie, univerzity,  
ústavy SAV,...
- Verejná správa
- Súkromné firmy, tretí sektor

### Naučte sa pracovať v prostredí HPC systémov:

Najbližší kurz:

[HPC infraštruktúra](#) / 18. máj 2022

### Hľadáme nových kolegov do tímu!

<https://eurocc.nscc.sk/career/>



**Sledujte nás na sociálnych sietiach:**



# Interaktívna prednáška

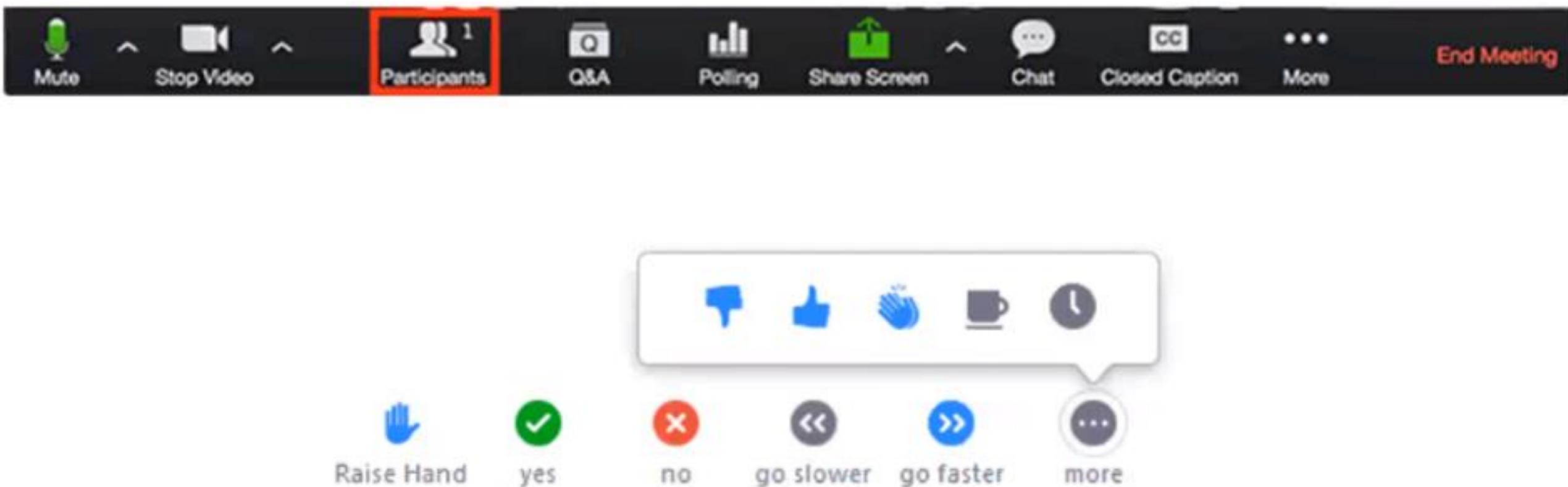
# Aktívne používanie a zapájanie sa

Participants (20)

Find a participant

| Participant             | Color Swatches | Mute | Video     |           |      |
|-------------------------|----------------|------|-----------|-----------|------|
| MR Miroslav Reiter (Me) |                |      |           |           |      |
|                         |                |      |           |           |      |
|                         |                |      |           |           |      |
|                         |                |      |           |           |      |
|                         |                |      |           |           |      |
|                         |                |      |           |           |      |
|                         |                |      |           |           |      |
|                         |                |      |           |           |      |
|                         |                |      |           |           |      |
|                         |                |      |           |           |      |
| Raise Hand              | yes            | no   | go slower | go faster | more |
|                         |                |      |           |           |      |

# Používame Zoom



# Vaše Ciele a Očakávania

1. Doplniť si znalosti z jazyka Python

2. Základy analytického myslenia

3. Základy spracovania dát

4. Základy vizualizácie dát

5. Doplniť si znalosti z JetBrains Datalore

6. Využitie AI pri programovaní

Zábava je v zaručená v každom bode :-)



Ako na spracovanie dát v Python?



File

Edit

View

Insert

Cell

Kernel

Widgets

Help

Snippets

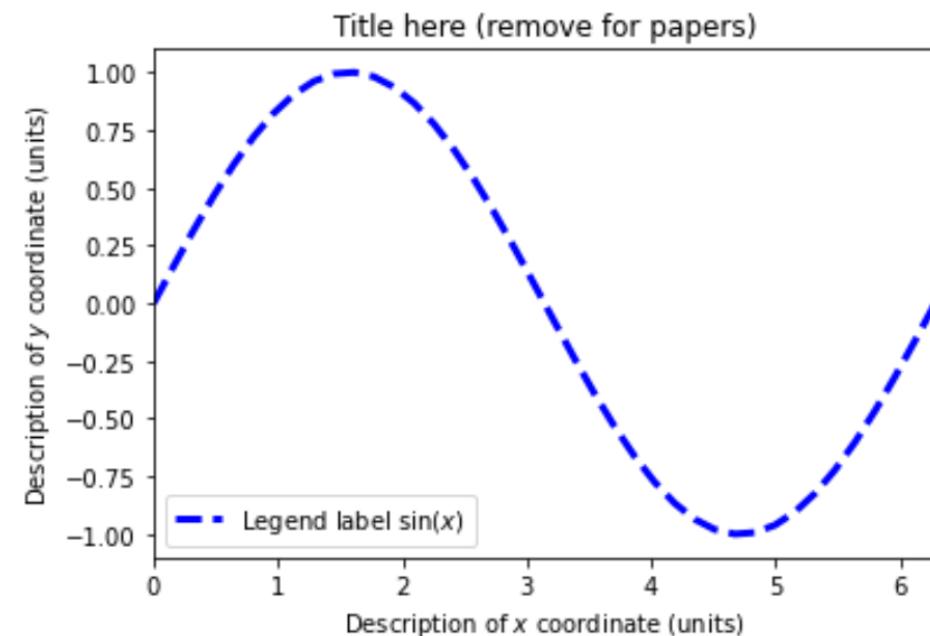
Trusted

Python 3



20 plt.show()

executed in 215ms, finished 01:58:33 2022-05-18



In [55]:

```
1 from __future__ import print_function, division
2 from sympy import *
3 a, s, t, u, v, w, x, y, z = symbols("a, s, t, u, v, w, x, y, z")
4 k, m, n = symbols("k, m, n", integer=True)
```

# Nástroje Dátovej Analýzy

 python



# Používané Nástroje Dátovej Analýzy

-  Python - momentálne **najpopulárnejší jazyk pre vedu o údajoch (Data science)**. Python je súčasťou interpretovaný a pomalší jazyk ako C/C++, ale využitím knižníc, ako je NumPy (a Pandas navyše), sa môžeme priblížiť rýchlosťi C.
-  Jupyter/Datalore - **nástroj s prostredím interaktívneho notebooku**, mocný prieskumný, vizualizačný a komunikačný nástroj.
-  Pandas - **nástroj** na manipuláciu s **tabuľkovými údajmi** v Python. Je to obrovská **knižnica**, ale ak poznáte **správne metódy**, môžeme vytvoriť výkonnú **analýzu**.
-  Matplotlib - **grafická knižnica s rôznymi grafmi a vizualizáciami**. V spojení s Pandas a Jupyter/Datalore môžeme rýchlo získať prehľad o našich údajoch.
-  Numpy - **knižnica** jazyka Python, ktorá poskytuje infraštruktúru pre **prácu s vektormi, maticami** a všeobecne **viacozmernými poliami**.
-  Existuje 300 000+ knižníc/modulov pre Python.

# Knižnice/Moduly pre Dátových Vedcov

22

1. Pandas
2. Matplotlib
3. Numpy
4. Seaborn
5. Plotly
6. pyparsing
7. Beatiful Soup
8. Scikit-Learn

Name A Better Trio. I'll Wait 😊



```
1 import numpy as np  
2 import pandas as pd  
3 import matplotlib.pyplot as plt
```

That's it.

# Pandas

## Important Methods *Pandas* Packages

### Data Importing

```
pd.read_csv()  
pd.read_table()  
pd.read_excel()  
pd.read_sql()  
pd.read_json()  
pd.read_html()  
pd.DataFrame()  
pd.concat()  
pd.Series()  
pd.date_range()
```



### Data Cleaning

```
pd.fillna()  
pd.dropna()  
pd.sort_values()  
pd.apply()  
pd.groupby()  
pd.append()  
pd.join()  
pd.rename()  
pd.to_csv()  
pd.set_index()
```

### Data Statistic

```
pd.head()  
pd.tail()  
pd.describe()  
pd.info()  
pd.mean()  
pd.median()  
pd.count()  
pd.std()  
pd.max()  
pd.min()
```

# Spracovanie Chýb a Výnimiek



 python

# Spracovanie Výnimiek

```
while True:  
    try:  
        x = int(input("Please enter a number: "))  
        break  
    except ValueError:  
        print("Oops! That was no valid number. Try again...")
```

# Spracovanie Výnimiek

```
... except (RuntimeError, TypeError, NameError):  
...     pass
```

# Spracovanie Výnimiek

---

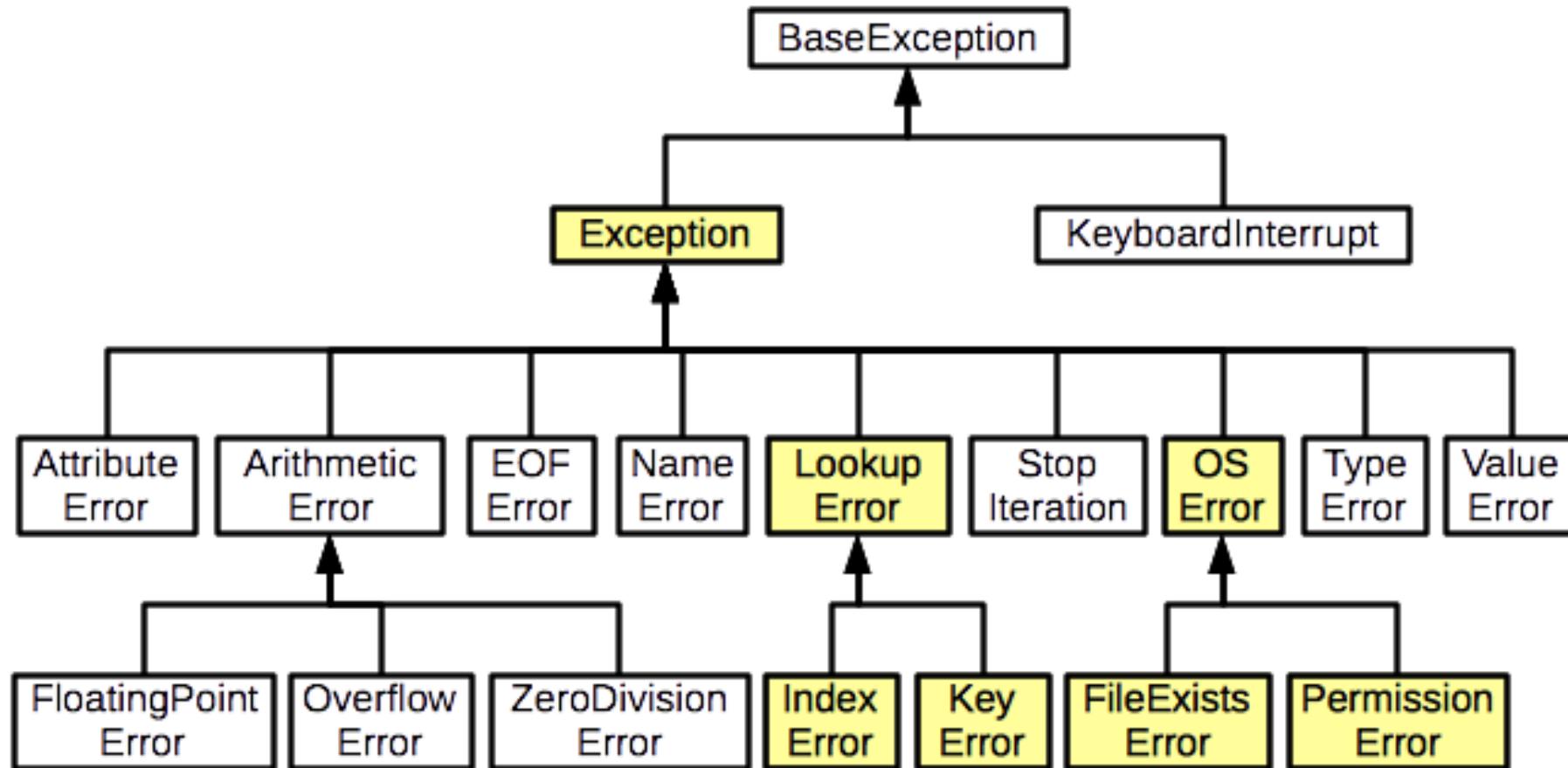
```
class B(Exception):
    pass

class C(B):
    pass

class D(C):
    pass

for cls in [B, C, D]:
    try:
        raise cls()
    except D:
        print("D")
    except C:
        print("C")
    except B:
        print("B")
```

# Diagram Tried Výnimiek



Multiexcept: `except (exception1, exception2) as e`

# Spracovanie výnimiek

- Try
- Except
- Try
- Finally
- Try
- Except
- Finally

```
try:  
    fh = open("testfile", "w")  
    fh.write("Toto je moj testovaci subor...")  
except IOError:  
    print "Chyba: nemozem najst subor alebo citat data"  
else:  
    print "Obsah uspesne zapisany"  
    fh.close()
```

# Spracovanie Výnimiek

```
UnboundLocalError
AssertionError
EOFError KeyError
IOError SyntaxError
SystemExit FloatingPointError
StopIteration OverflowError
StandardError
KeyboardInterrupt
ZeroDivisionError
ImportError
Exception
```

```
def vypocitajCenuDPH(cena):
    try:
        return int(cena)
    except ValueError as exVal:
        print "Argument neobsahuje cislo"
        print exVal
    ##      print exVal.message
    except TypeError as exType:
        print "Nezadal si argument..."

    print vypocitajCenuDPH("xyz")
    print vypocitajCenuDPH(100)
    print vypocitajCenuDPH(None)
    print vypocitajCenuDPH()
```

# Otváranie a Zatváranie Súborov

```
f = open('data.txt')  
try:  
    data = f.read()  
finally:  
    f.close()  
  
with open('data.txt') as f:  
    data = f.read()
```

# Odstraňovanie Súborov

```
try:  
    os.remove('somefile.tmp')  
except OSError:  
    pass  
  
with ignored(OSError):  
    os.remove('somefile.tmp')
```

# Asercia

Testovanie  
programu

Princíp raise-if

Ak je výraz True,  
pokračuje  
sa vo vykonávaní  
príkazov

Ak je výraz False,  
vyvolá sa  
výnimka  
**AssertionError**

```
def kelvinNaFahrenheit(teplota):  
    assert (teplota >= 0), "Menej ako absolutna 0!"  
    return ((teplota-273)*1.8)+32
```

```
print kelvinNaFahrenheit(100)  
print int(kelvinNaFahrenheit(500.55))  
print kelvinNaFahrenheit(-5)
```

# Úvod do Python Knižníc



# Knižnice/Moduly pre Dátových Vedcov

1. Pandas
2. Matplotlib
3. Numpy
4. Seaborn
5. Plotly
6. Scikit-Learn
7. Beatiful Soup

Name A Better Trio. I'll Wait 😊



```
1 import numpy as np  
2 import pandas as pd  
3 import matplotlib.pyplot as plt
```

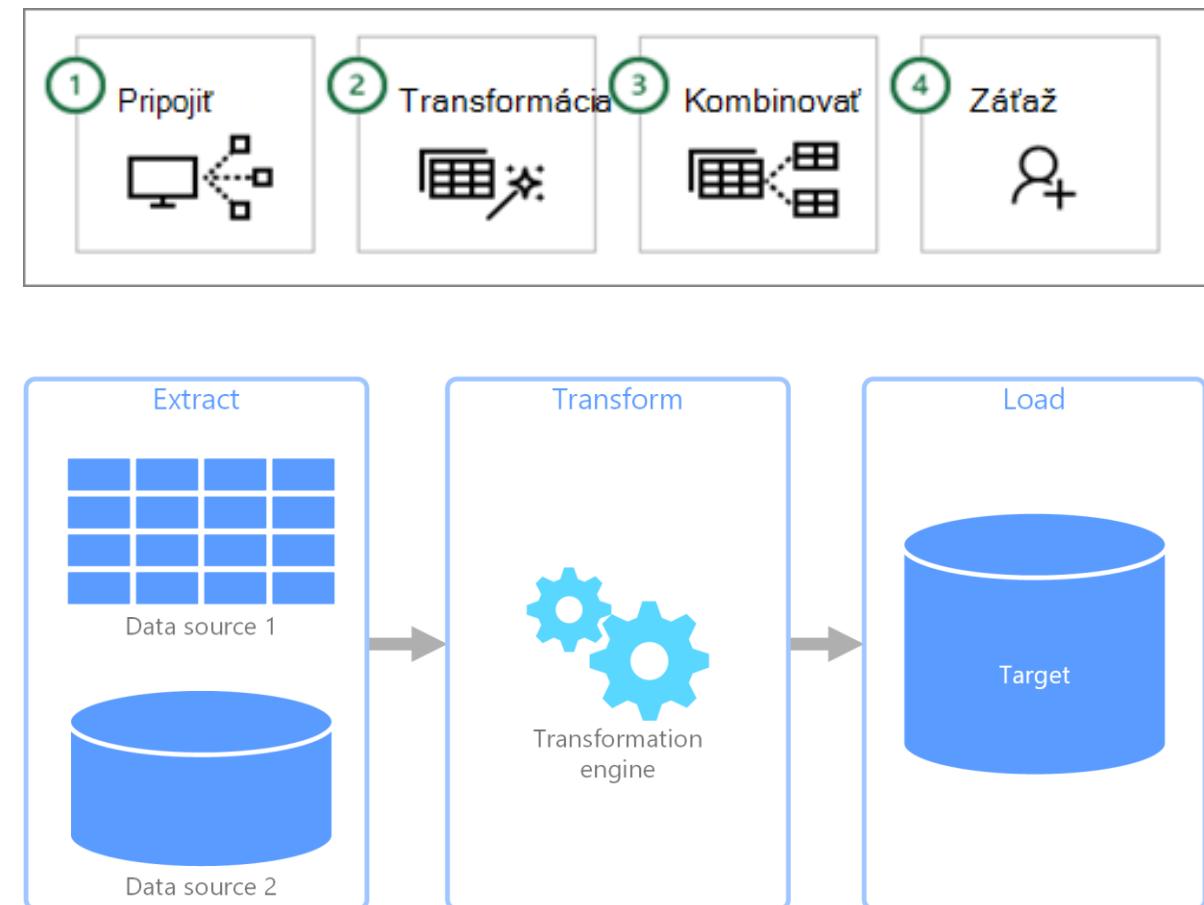
That's it.

# Používané Nástroje Dátovej Analýzy

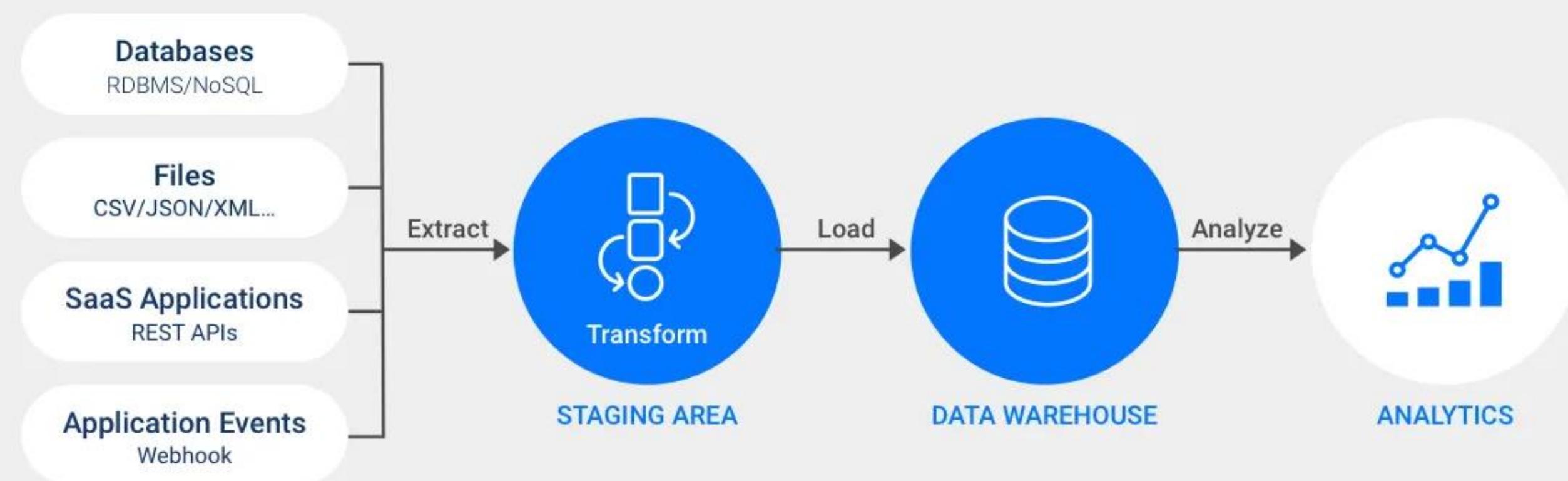
-  Python - momentálne najpopulárnejší jazyk pre vedu o údajoch (Data science). Python je súčasťou interpretovaného a pomalšieho jazyka ako C/C++, ale využitím knižníc, ako je NumPy (a Pandas navyše), sa môžeme priblížiť rýchlosťi C.
-  Jupyter - nástroj s prostredím interaktívneho notebooku, mocný prieskumný, vizualizačný a komunikačný nástroj.
-  Pandas - nástroj na manipuláciu s tabuľkovými údajmi v Python. Je to obrovská knižnica, ale ak poznáte správne metódy, môžeme vytvoriť výkonnú analýzu.
-  Matplotlib - grafická knižnica s rôznymi grafmi a vizualizáciami. V spojení s Pandas a Jupyter môžeme rýchlo získať prehľad o našich údajoch.
-  Numpy - knižnica jazyka Python, ktorá poskytuje infraštruktúru pre prácu s vektormi, maticami a všeobecne viacozmernými poliami.
-  Existuje 300 000+ knižníc/modulov pre Python.

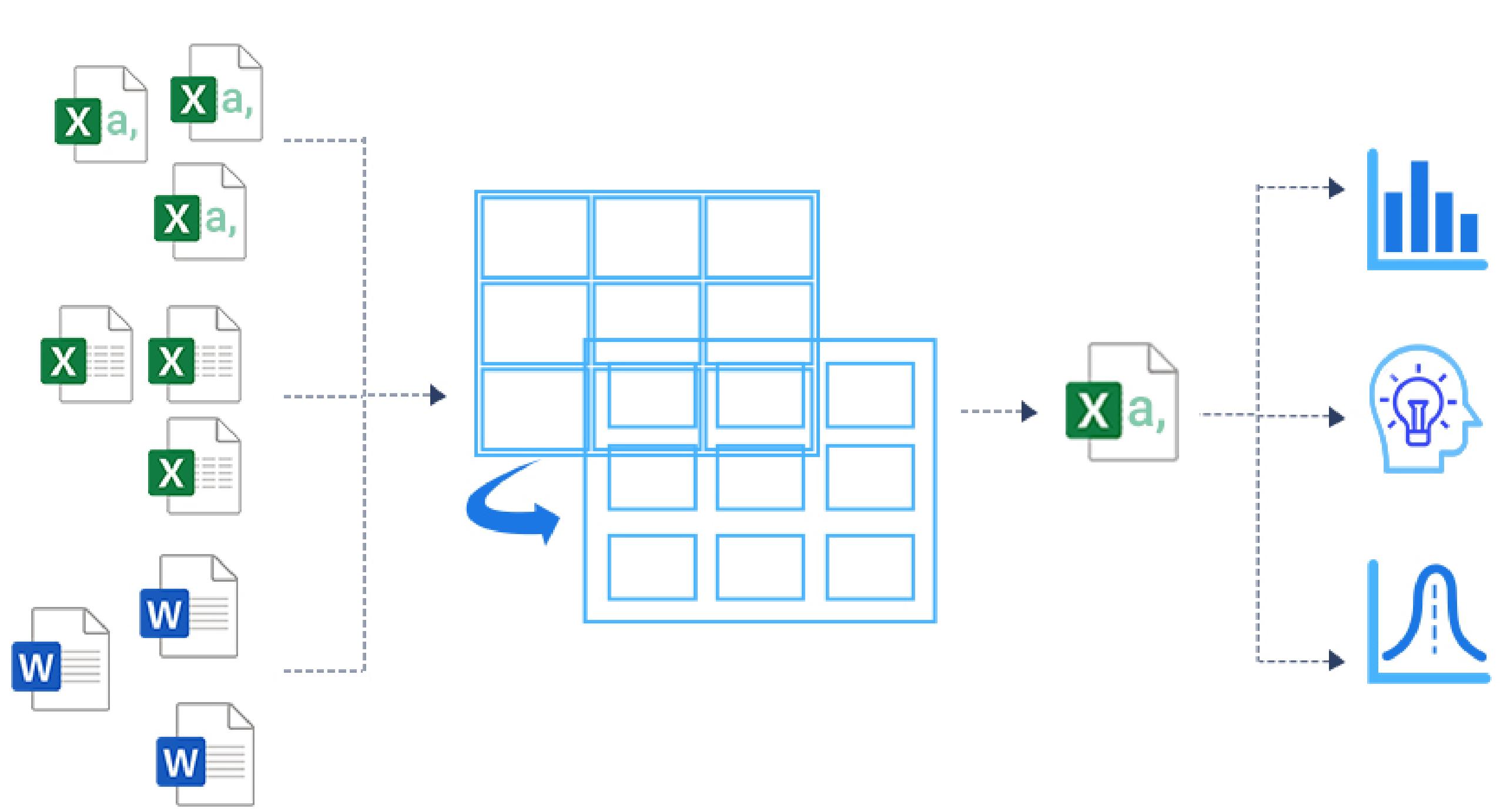
# 3 (4) Fázy Procesu ETL

- 1. Pripojiť** - Pripojenie k údajom v cloude, v službe alebo lokálne
- 2. Transformovať** - Tvarovanie údajov podľa vašich potrieb, zatiaľ čo pôvodný zdroj zostane nezmenený
- 3. Kombinovať** - Integrácia údajov z viacerých zdrojov na získanie jedinečného zobrazenia údajov
- 4. Zaťaženie** - Dokončenie dopytu a načítajte ho do dátového rámca alebo inej štruktúry a pravidelne ho obnovujte



# ETL PROCESS





 bratislava.sk 2 current visitors

Filter

All time

UNIQUE VISITORS**492k**

## TOTAL VISITS

**584k**

## TOTAL PAGEVIEWS

**1.3M**

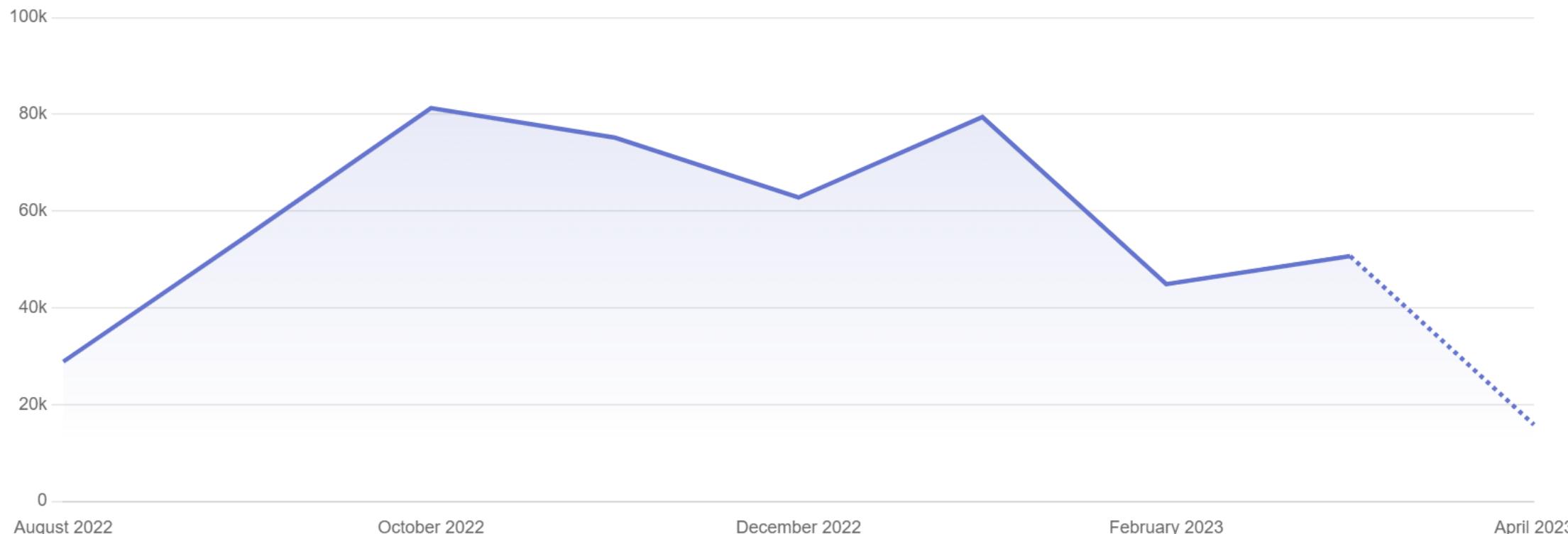
## VIEWS PER VISIT

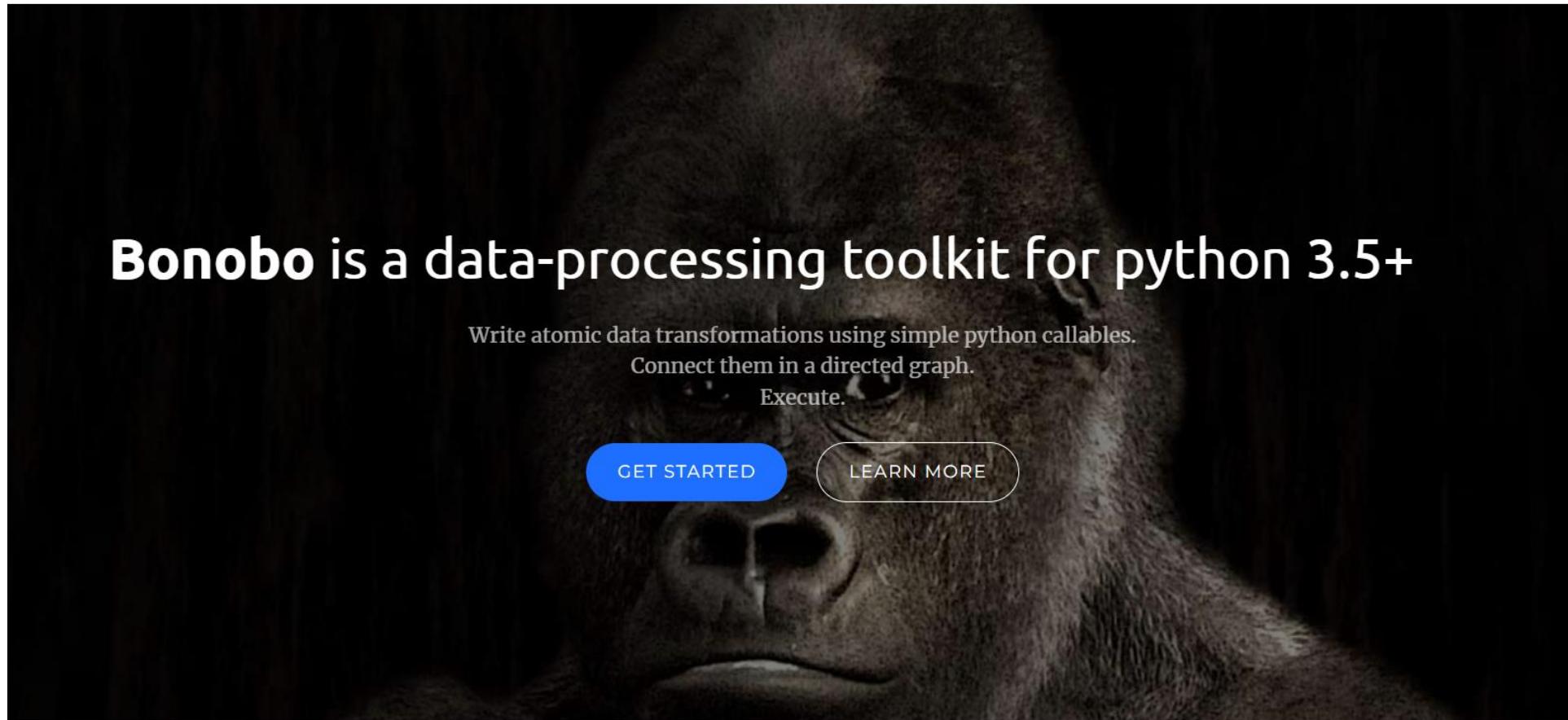
**2.33**

## BOUNCE RATE

**67%**

## VISIT DURATION

**2m 25s** Months 



# Bonobo is a data-processing toolkit for python 3.5+

Write atomic data transformations using simple python callables.

Connect them in a directed graph.

Execute.

[GET STARTED](#)[LEARN MORE](#)

pip install bonobo

## What is Bonobo?

Bonobo is a lightweight Extract-Transform-Load (ETL) framework for Python 3.5+.

It provides tools for building data transformation pipelines, using plain python primitives, and executing them in parallel.

## Releases

Current: [0.6.3 / !\[\]\(59f0787578e904376cbd9ab065793c8e\_img.jpg\) master \(docs\)](#)

Next: [0.7 / !\[\]\(f8e2499c664adc6845a0af59287fcccf\_img.jpg\) develop \(docs\)](#)



## Extract ...

Use built-in extractors (CSV, JSON, XML, XLS, ...), official add-ons (SQL), or write your own.



## Transform ...

Apply changes using simple python, or provided transformation classes.



## Load ...

Use our built-in writers (CSV, JSON, XML, XLS, ...) or connect to your custom services.



## ... in a Simple

Tired of learning new APIs?  
You'll be up and running in 10 minutes,  
if you know some python.



## ... Atomic

Each transformation has a specific,  
unique, small and scoped purpose (think  
UNIX), enhancing testability and ease of  
maintenance.



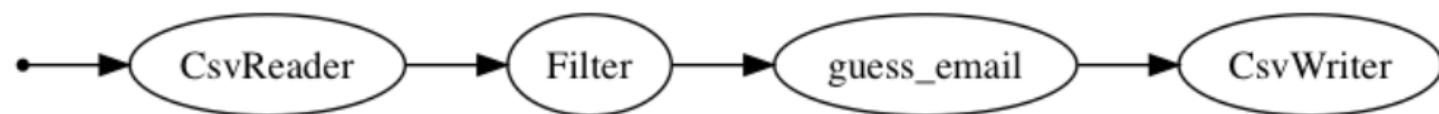
## ... and Pythonic way!

It's just python ! We worked hard to  
provide a clean API, using the standard  
data structures you already know.

## It's just Python!

The smallest building blocks of Bonobo scripts are plain old python objects. Anything callable (or iterable) can be used as a node.

Bonobo provides the tools to combine the nodes in graphs, visualize the structure and execute them efficiently.



```
import bonobo

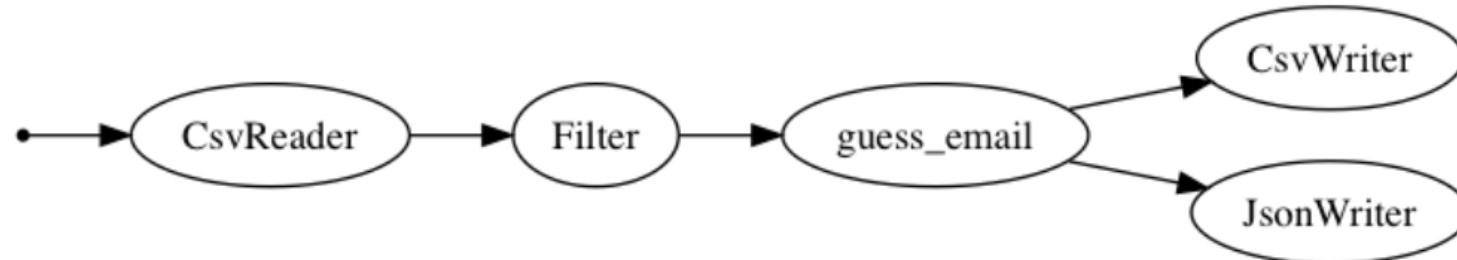
def guess_email(**row):
    return {
        **row,
        'email': row['name'] + '@' + row['domain']
    }

graph = bonobo.Graph(
    bonobo.CsvReader('employees.csv'),
    bonobo.Filter(
        lambda **row: row['position'] != 'CEO'
    ),
    guess_email,
    bonobo.CsvWriter('employees.output.csv'),
)
```

## Enhance your graph!

It's easy to design your graph according to your business requirements.

```
graph.add_chain(  
    bonobo.JsonWriter('employees.output.json'),  
    _input=guess_email,  
)
```



## Build your own library...

Bonobo is just a minimalistic toolkit to help you build ETL jobs.

Use `bonobo.config` to create parametrizable classes that will get configured at runtime with options and service dependencies, or use built-in transformations to work with usual file formats and operations.

Clean, readable and easy to test!

```
from bonobo.config import Configurable, Service, Option  
  
class SendMail(Configurable):  
    subject = Option(positional=True, default='Hello, {name}')  
    smtp = Service('smtp')  
  
    def call(self, smtp, **row):  
        subject = self.subject.format(**row)  
        body = ''  
        We're so glad to have you there!  
        Let's have a coffee, someday.  
        ...  
        smtp.send(subject, body)
```



Search projects



Help

Sponsors

Log in

Register

# bonobo 0.6.4

`pip install bonobo`



[Latest version](#)

Released: May 16, 2019

Bonobo, a simple, modern and atomic extract-transform-load toolkit for python 3.5+.

## Navigation

Project description

Release history

Download files

## Project links

Homepage

Download

## Project description

Data-processing for humans.

v0.6.4 3.5 | 3.6 passing Continuous Integration (Linux) build passing maintainability ?  
 coverage 80%

Bonobo is an extract-transform-load framework for python 3.5+ (see comparisons with other data tools).

Bonobo uses plain old python objects (functions, generators and iterators), allows them to be linked together in a directed graph, and then executed using a parallelized strategy, without having to worry about the underlying complexity.

Developers can focus on writing simple and atomic operations, that are easy to unit-test by-design, while the focus of the framework is to apply them concurrently to rows of data.

# Zistenie informácií o krajine

- License: MIT License
- Author: Porimol Chandro
- pip install **countryinfo**
- **from countryinfo import**  
CountryInfo
- <https://pypi.org/project/countryinfo/>

```
1 from countryinfo import CountryInfo
2 tmp = input("Zadaj nazov krajiny: ")
3 krajina = CountryInfo(tmp)
4 print("Hlavne mesto: ", krajina.capital())
5 print("Meny: ", krajina.currencies())
6 print("Jazyky: ", krajina.languages())
7 print("Hranice: ", krajina.borders())
8 print("Ine nazvy: ", krajina.alt_spellings())
```

executed in 3.96s, finished 02:11:34 2022-05-18

```
Zadaj nazov krajiny: Slovakia
Hlavne mesto: Bratislava
Meny: ['EUR']
Jazyky: ['sk']
Hranice: ['AUT', 'CZE', 'HUN', 'POL', 'UKR']
Ine nazvy: ['SK', 'Slovak Republic', 'Slovenská republika']
```



Search projects



Help

Sponsors

Log in

Register

# countryinfo 0.1.2



Latest version

pip install countryinfo

Released: Jun 1, 2020

countryinfo is a python module for returning data about countries, ISO info and states/provinces within them.

## Navigation

Project description

Release history

Download files

## Project links

Homepage

## Statistics

GitHub statistics:

## Project description

A python module for returning data about countries, ISO info and states/provinces within them.

## Table of Contents

- [Install](#)
- [API Usage](#)

## APIs

- [.info\(\)](#)
- [.provinces\(\)](#)
- [.alt\\_spellings\(\)](#)
- [.area\(\)](#)



File Edit View Insert Cell Kernel Widgets Help Snippets

Trusted

Python 3



Description of x coordinate (units)

In [5]:

```
1 from PIL import Image  
2 im = Image.open("psicky.png")  
3 im.show()
```

executed in 1.50s, finished 02:34:10 2022-05-18

In [22]:

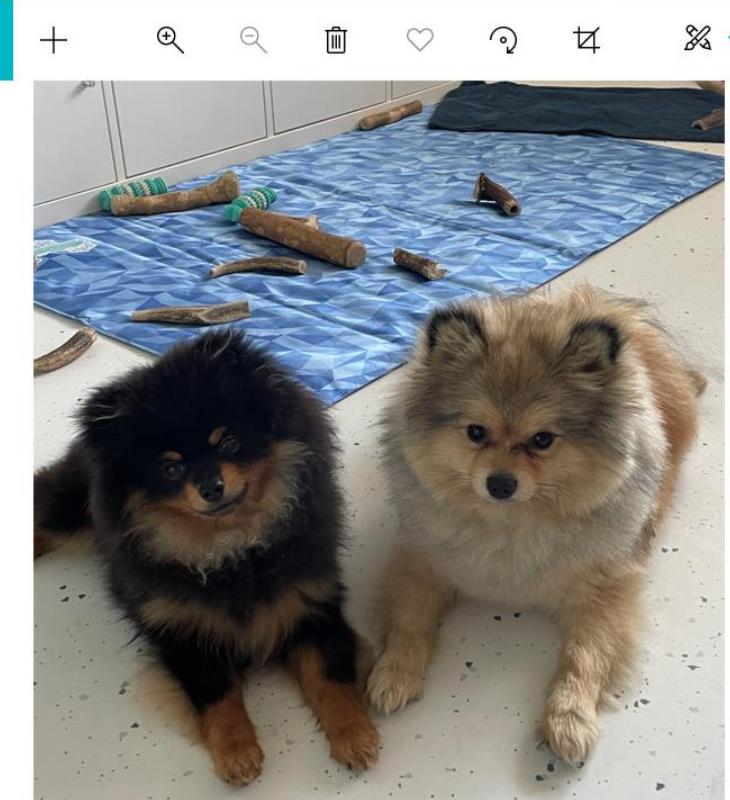
```
1 #Import Library  
2 from IPython.display import Image  
3 # Load image from local storage  
4 Image(filename = "excel-grafy.jpg", width = 150, height = 150)  
5
```

executed in 21ms, finished 01:02:19 2022-05-18

out[22]:



Fotografie – tmp9danxky3.PNG



Všetko

Mapy

Obrázky

Správy

Videá

Viac

Nástroje

Približne 1 180 000 výsledkov (0,33 sekundy)



12 °C | °F

Pravdepodobnosť zrážok: 2%

Vlhkosť: 80%

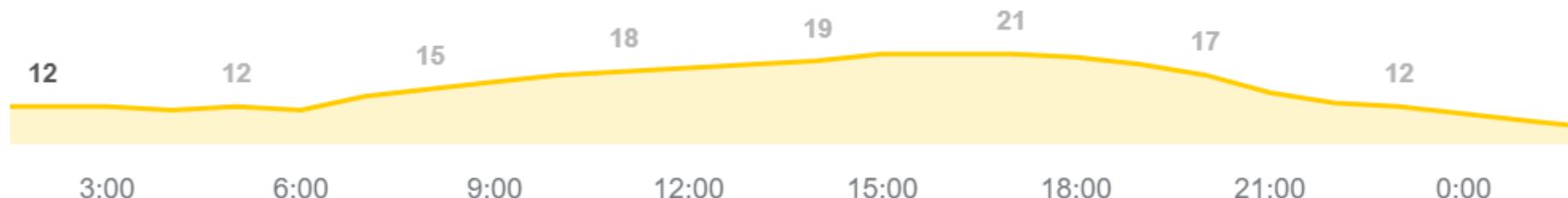
Vietor: 13 km/hod.

Bratislava  
streda 02:00  
Jasno

Teplota

Zrážky

Vietor



Približne 1 180 000 výsledkov (0,33 sekundy)



12 °C | °F

Pravdepodobnosť zrážok: 2%  
Vlhkosť: 80%  
Vietor: 13 km/hod.

Teplota | Zrážky | Vietor

Bratislava

streda

Jasno

span#wob\_dc 42.7 x 18.18

Prvky Konzola Zdroje Siet Výkonnosť Pamäť Aplikácia Zabezpečenie Lighthouse

Štýly Vypočítané Rozloženie >

Filtrovat :hov .cls +

element.style {

}

Zdedené zo zdroja div#wob\_dcp.wob\_dcp

.wob\_dts, .wob\_dcp {

color: #70757a; font-size: 16px; line-height: 20px; }

Zdedené zo zdroja div.VQF4g

.VQF4g {

text-align: right; }

Zdedené zo zdroja div#res.eqAnXb

html body#gsr.srp div#main.main div#cnt.e9EfHf div#rcnt.GyAeWb div#center\_col.s6JM6d div#res.eqAnXb div#search div ...

<hr class="v5jHUB">

<div class="ULSxyf">

<div class="KIy09e obcontainer wDYxhc" data-md="2">

<div data-hveid="CAEQAA" data-ved="2ahUKEwiksb2V5ef3AhWKyaQKHRaiCzcQohJ6BAgBEAA">

<h2 class="Uo8X3b OhScic zsYMMe">Výsledok predpovede počasia</h2>

<div class="nawv0d" id="wob\_wc">

<div class="UQt4rd">...</div>

<div class="VQF4g">

<span aria-level="3" role="heading">

<div class="wob\_loc q8U8x" id="wob\_loc">Bratislava</div>

<div class="wob\_dts" id="wob\_dts">streda 02:00</div>

<div class="wob\_dcp" id="wob\_dcp">

<span id="wob\_dc">Jasno</span> == \$0

</div>

</span>

</div>

<div id="wob\_d">...</div>



← Štatistika

Denný kurzový lístok ECB

Mesačné, kumulatívne a ročné prehľady kurzov

Kurzový lístok vybraných cudzích mien voči EUR

Archív kurzových lístkov NBS

#### SÚVISEJACE ODKAZY

Grafy kurzov →

# Kalkulačka

Dátum

18.05.2022



Kurz zo dňa 17.05.2022

Suma, ktorú chcete zmeniť

EUR

50



1 EUR = 24,712 CZK

Suma, ktorú dostanete

CZK

1 235,6



In [11]:

```
1 # !pip install forex_python
2 from forex_python.converter import CurrencyRates
3
4 c = CurrencyRates()
5 ciastka = int(input("Zadajte ciastku: "))
6 z_meny = input("Z meny: ").upper()
7 do_meny = input("Do meny: ").upper()
8 print("Z: ", z_meny, " Do: ", do_meny, " Ciastka: ", ciastka)
9
10 vysledok = c.convert(z_meny, do_meny, ciastka)
11 print(vysledok, do_meny)
```

executed in 12.2s, finished 02:49:10 2022-05-18

Zadajte ciastku: 1000

Z meny: eur

Do meny: czk

Z: EUR Do: CZK Ciastka: 1000

24712.0 CZK

# Grafy a Vizualizácie





 **Square**

 **plotly**

 **seaborn**

 **matplotlib**

# Grafy a Vizualizácie

1. Matplotlib
2. Seaborn
3. Plotly
4. Squarify
5. d3

# GRAPHICS LIBRARIES



## Matplotlib

With a syntax similar to Matlab, matplotlib is the most used low-level charting library in Python



## Seaborn

seaborn is a matplotlib wrapper. Makes it possible to create beautiful charts with few lines of code



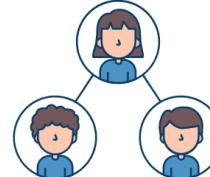
## Plotly

If you prefer dynamic charts over static, then plotly / plotly Express is your best choice



## Distribúcia

Distribučné grafy umožňujú vizualizovať, ako sa údaje rozkladajú pozdĺž podpory, a porovnať niekoľko skupín



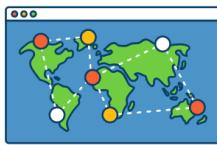
## Korelácia

Korelačné grafy sú užitočné na vizualizáciu vzťahu medzi dvoma alebo viacerými premennými



## Evolúcia

Vývojové grafy ukazujú, ako sa premenná alebo množina alebo premenné vyvíjajú, vo všeobecnosti v priestore alebo čase



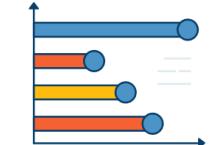
## Priestorový

Tieto grafy zobrazujú geografické oblasti a ich vzťahy založené na jednej alebo viacerých premenných spojených s týmito oblasťami



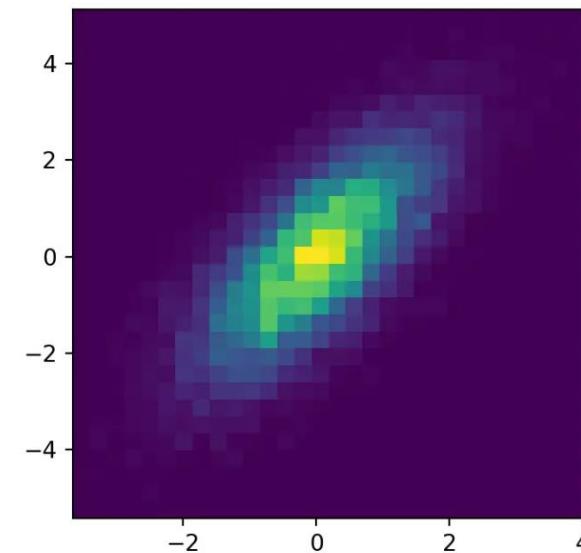
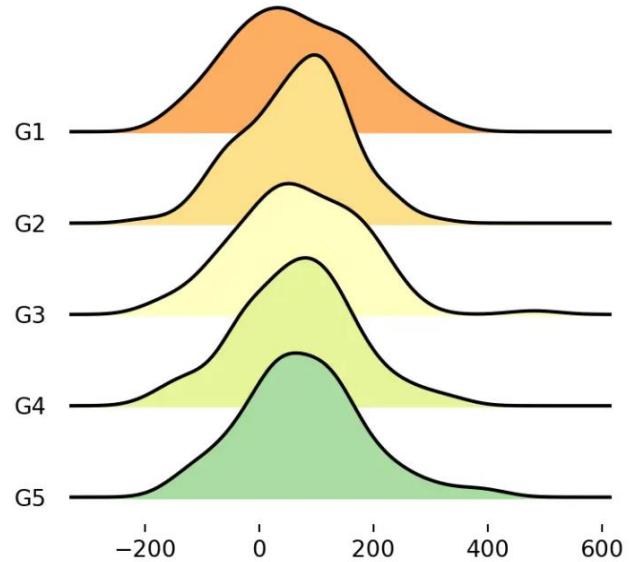
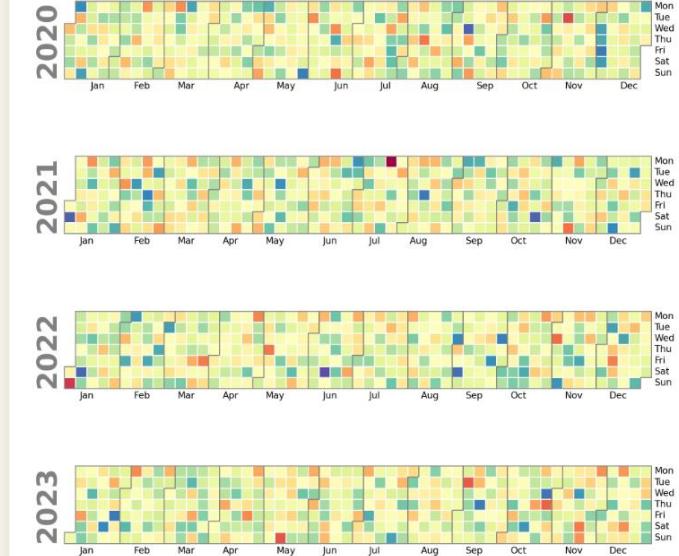
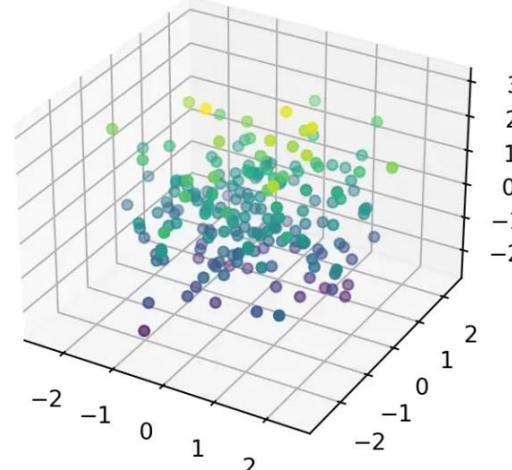
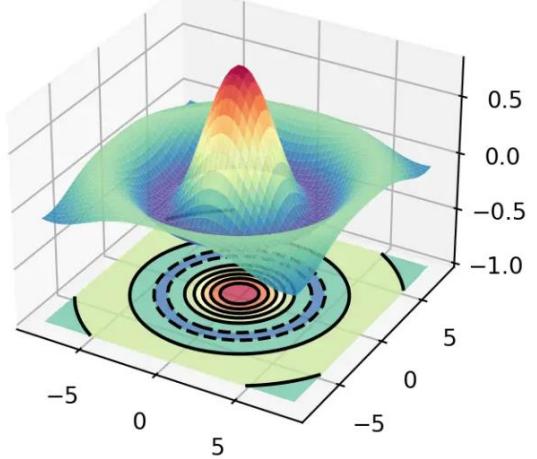
## Časť celku

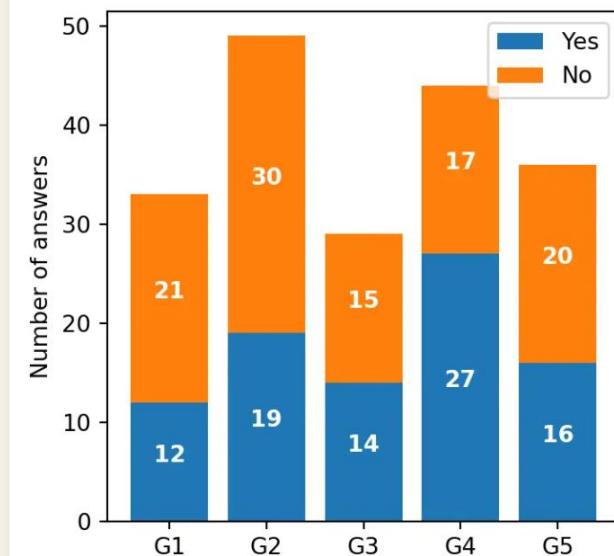
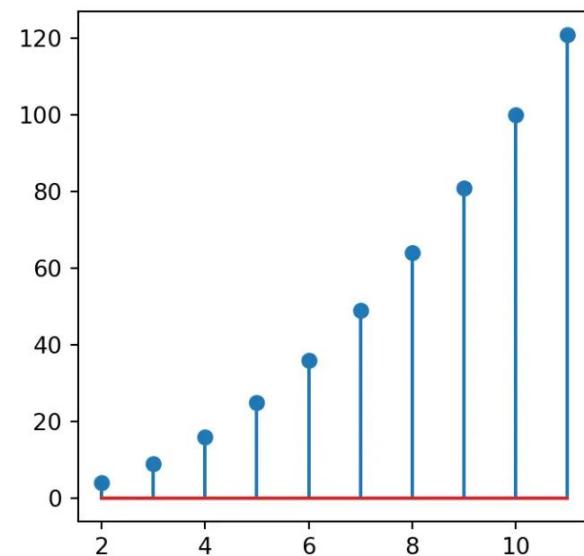
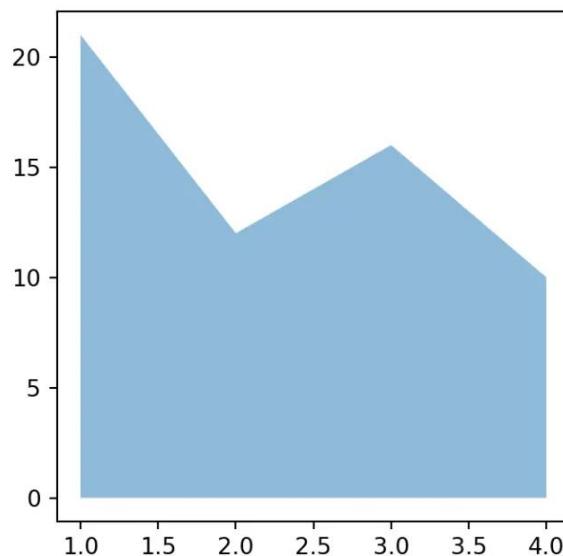
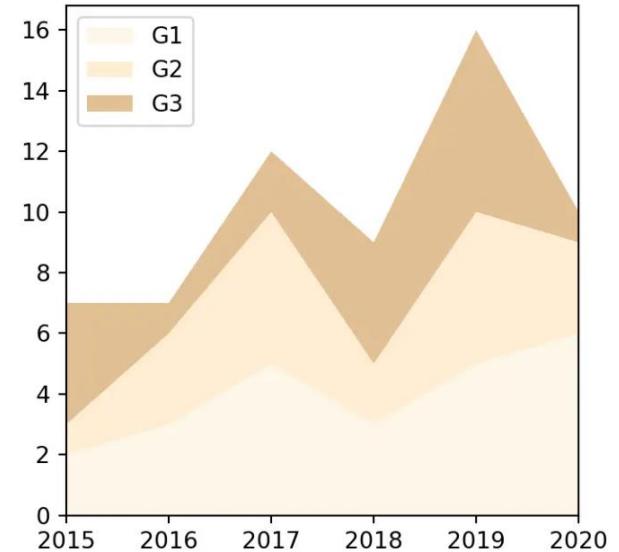
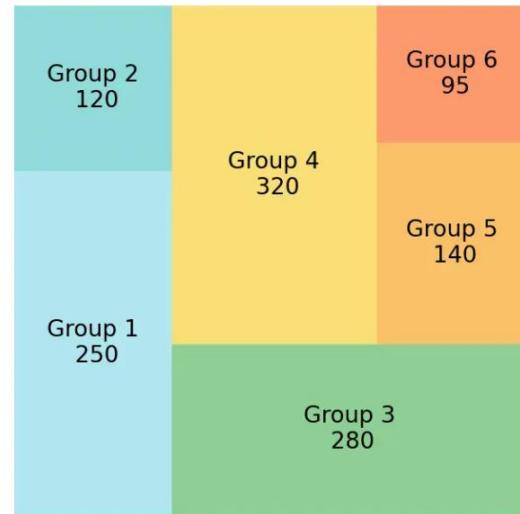
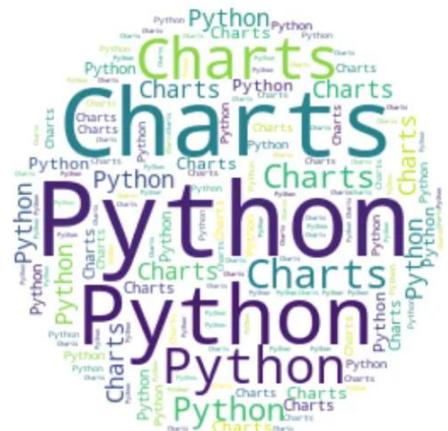
Časť celého grafu sumarizuje údaje v castiach alebo výrezoch. Tento typ grafov je veľmi užitočný na znázornenie počtov alebo skupín

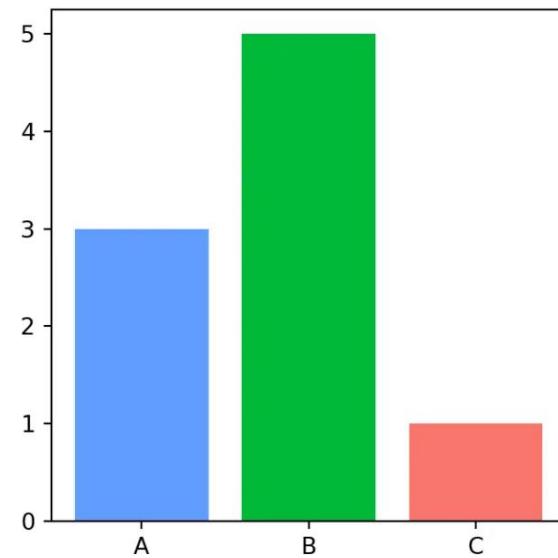
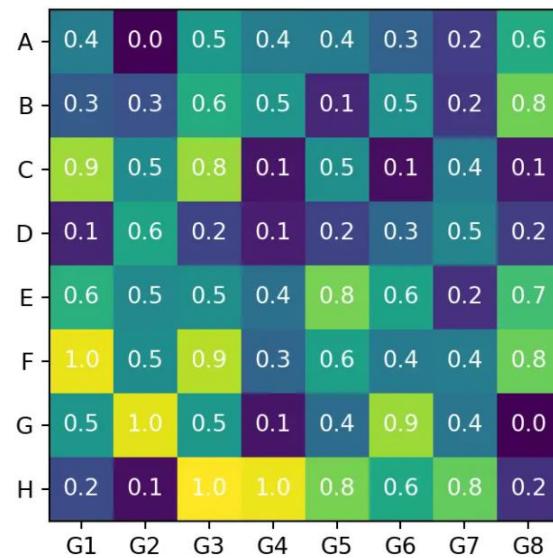
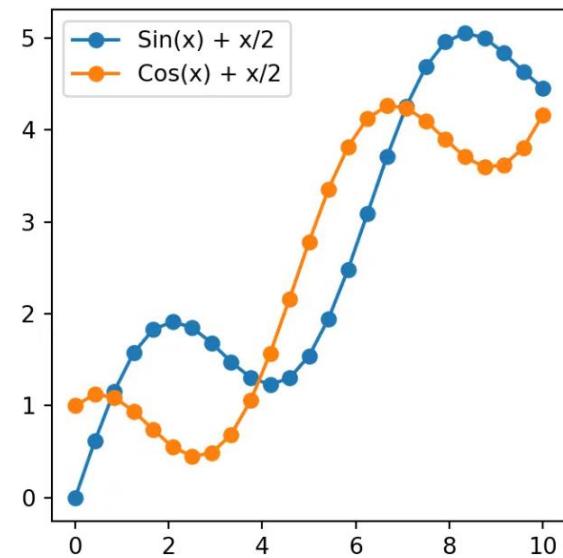
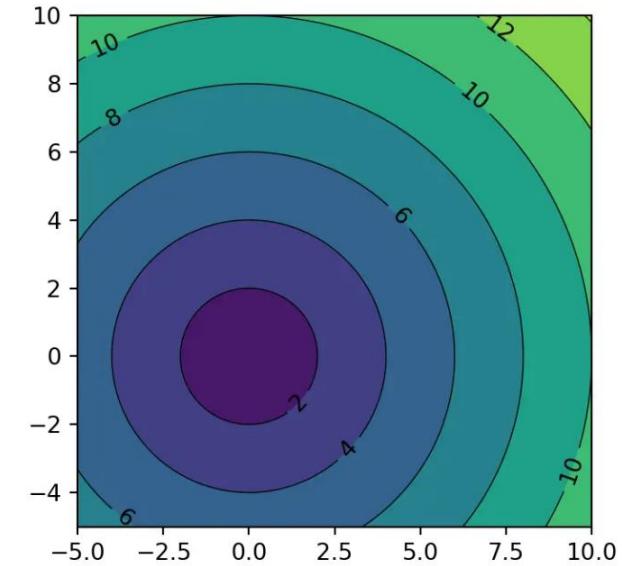
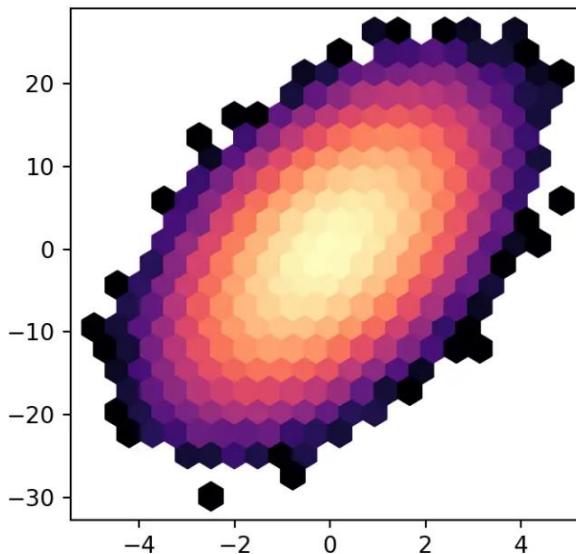
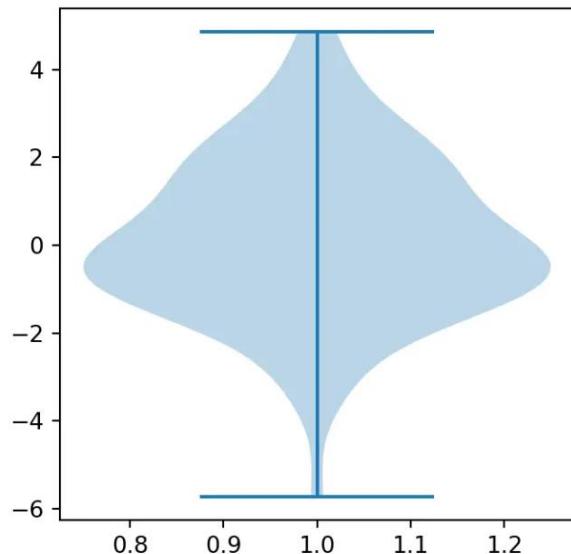


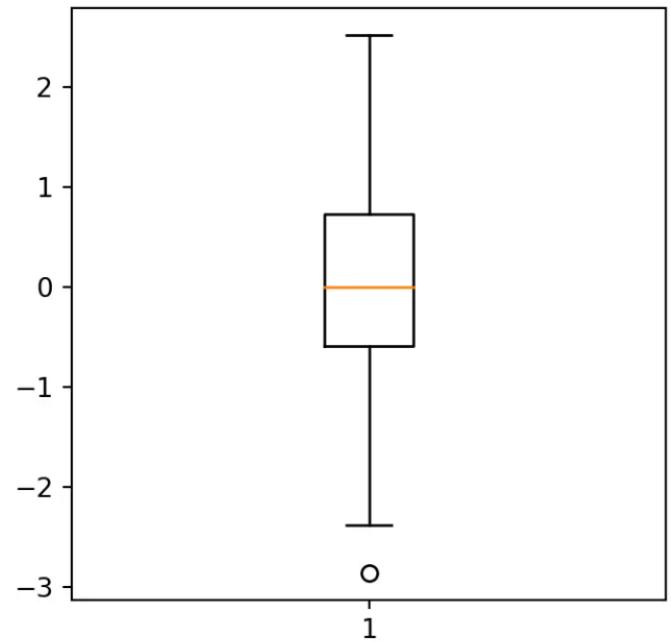
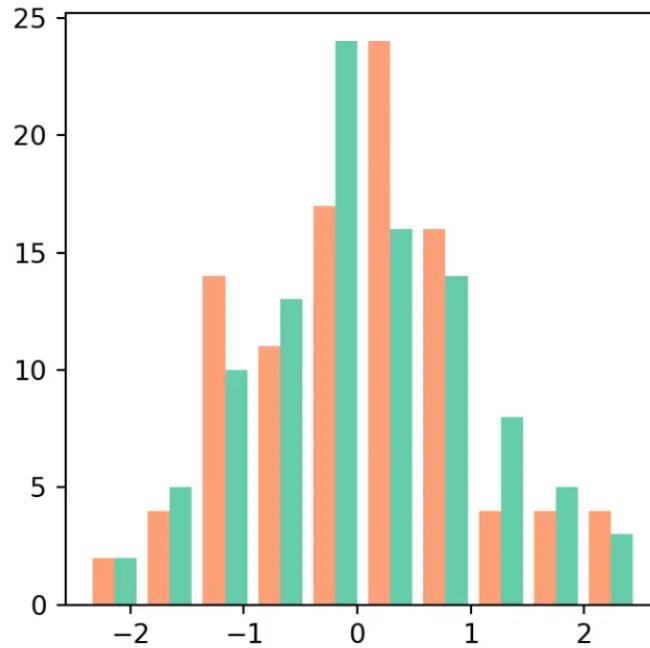
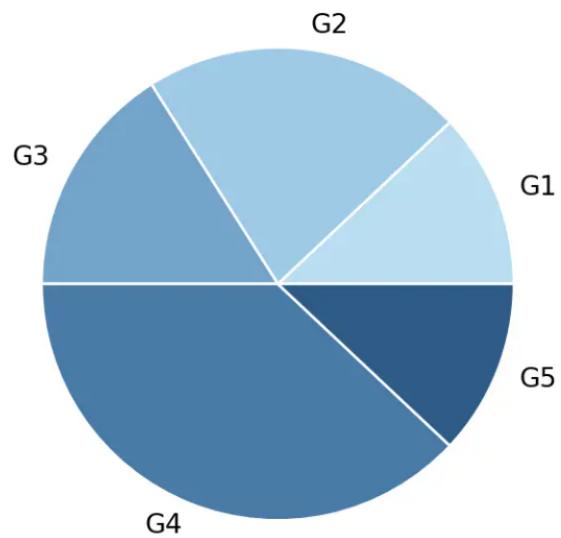
## Rebríček

Hodnotiace tabuľky umožňujú vizualizáciu klasifikácie medzi pozorovaniami premennej alebo medzi rôznymi premennými









## Quick start

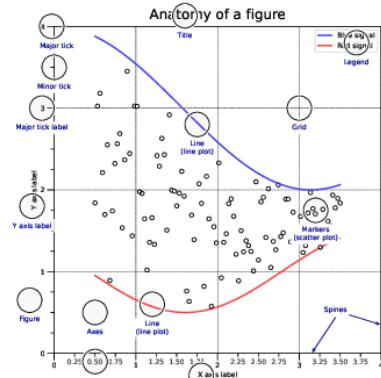
```
import numpy as np
import matplotlib as mpl
import matplotlib.pyplot as plt

X = np.linspace(0, 2*np.pi, 100)
Y = np.cos(X)

fig, ax = plt.subplots()
ax.plot(X, Y, color='green')

fig.savefig("figure.pdf")
plt.show()
```

## Anatomy of a figure



## Subplots layout

```
subplot[s](rows,cols,...)
fig, axs = plt.subplots(3, 3)

G = gridspec(rows,cols,...)
ax = G[0,:]

ax.inset_axes(extent)

d=make_axes_locatable(ax)
ax = d.new_horizontal('10%')
```

## Getting help

- [matplotlib.org](#)
- [github.com/matplotlib/matplotlib/issues](#)
- [discourse.matplotlib.org](#)
- [stackoverflow.com/questions/tagged/matplotlib](#)
- <https://gitter.im/matplotlib/matplotlib>
- [twitter.com/matplotlib](#)
- [Matplotlib users mailing list](#)

## Basic plots

|  |   |   |                     |
|--|---|---|---------------------|
|  | <code>plot([X], Y, [fmt], ...)</code>     | X, Y, fmt, color, marker, linestyle         | <a href="#">API</a> |
|  | <code>scatter(X, Y, ...)</code>           | X, Y, [s]izes, [c]olors, marker, cmap       | <a href="#">API</a> |
|  | <code>bar[h](x, height, ...)</code>       | x, height, width, bottom, align, color      | <a href="#">API</a> |
|  | <code>imshow(Z, ...)</code>               | Z, cmap, interpolation, extent, origin      | <a href="#">API</a> |
|  | <code>contour(f)([X], [Y], Z, ...)</code> | X, Y, Z, levels, colors, extent, origin     | <a href="#">API</a> |
|  | <code>pcolormesh([X], [Y], Z, ...)</code> | X, Y, Z, vmin, vmax, cmap                   | <a href="#">API</a> |
|  | <code>quiver([X], [Y], U, V, ...)</code>  | X, Y, U, V, C, units, angles                | <a href="#">API</a> |
|  | <code>pie(x, ...)</code>                  | Z, explode, labels, colors, radius          | <a href="#">API</a> |
|  | <code>text(x, y, text, ...)</code>        | x, y, text, va, ha, size, weight, transform | <a href="#">API</a> |
|  | <code>fill[_between](x)(...)</code>       | X, Y1, Y2, color, where                     | <a href="#">API</a> |

## Advanced plots

|  |  |                                     |                     |
|--|--|-------------------------------------|---------------------|
|  | <code>step(X, Y, [fmt], ...)</code>          | X, Y, fmt, color, marker, where     | <a href="#">API</a> |
|  | <code>boxplot(x, ...)</code>                 | X, notch, sym, bootstrap, widths    | <a href="#">API</a> |
|  | <code>errorbar(X, Y, xerr, yerr, ...)</code> | X, Y, xerr, yerr, fmt               | <a href="#">API</a> |
|  | <code>hist(X, bins, ...)</code>              | X, bins, range, density, weights    | <a href="#">API</a> |
|  | <code>violinplot(D, ...)</code>              | D, positions, widths, vert          | <a href="#">API</a> |
|  | <code>barbs([X], [Y], U, V, ...)</code>      | X, Y, U, V, C, length, pivot, sizes | <a href="#">API</a> |
|  | <code>eventplot(positions, ...)</code>       | positions, orientation, lineoffsets | <a href="#">API</a> |
|  | <code>hexbin(X, Y, C, ...)</code>            | X, Y, C, gridsize, bins             | <a href="#">API</a> |

## Scales

|  |   |        |                     |
|--|---|--------|---------------------|
|  | <code>ax.set_[xy]scale(scale, ...)</code> | linear | <a href="#">API</a> |
|  | any values                                | log    | values > 0          |
|  | any values                                | symlog | 0 < values < 1      |

## Projections

|  |   |                            |                     |
|--|---|----------------------------|---------------------|
|  | <code>subplot(..., projection=p)</code> | p='polar'                  | <a href="#">API</a> |
|  | p='3d'                                  | <a href="#">API</a>        |                     |
|  | p=ccrs.Orthographic()                   | import cartopy.crs as ccrs | <a href="#">API</a> |

## Tick locators

```
from matplotlib import ticker
ax.[x|y]axis.set_[minor|major]_locator(locator)

ticker.NullLocator()
ticker.MultipleLocator(0.5)
ticker.FixedLocator([0, 1, 5])
ticker.LinearLocator(numticks=3)
ticker.IndexLocator(base=8, 5, offset=0.25)
ticker.AutoLocator()
ticker.MaxNLocator(n=4)
ticker.LogLocator(base=10, numticks=15)
```

## Tick formatters

```
from matplotlib import ticker
ax.[x|y]axis.set_[minor|major]_formatter(formatter)

ticker.NullFormatter()
ticker.FixedFormatter(['zero', 'one', 'two', '...', 'five'])
ticker.FuncFormatter(lambda x, pos: "%.{2f} % % x")
ticker.FormatStrFormatter('%d')
ticker.ScalarFormatter()
ticker.StrMethodFormatter('x')
ticker.PercentFormatter(max=5)
```

## Lines

|  |        |         |              |              |
|--|--------|---------|--------------|--------------|
| <code>linestyle or ls</code>           | ---    | -.      | ....         | (0, 0.01, 2) |
| <code>capstyle or dash_capstyle</code> | "butt" | "round" | "projecting" |              |

## Markers

|            |            |            |            |            |              |
|------------|------------|------------|------------|------------|--------------|
| o          | □          | +          | ★          | △          | ▽            |
| •          | ◆          | ×          | ○          | ◇          | ▼            |
| ◦          | ▲          | ✖          | ◆          | ◆          | ▶            |
| ▴          | ▾          | ▴          | ▴          | ▴          | ▴            |
| ◂          | ▸          | ◂          | ◂          | ◂          | ◂            |
| ▴▴         | ▸▸         | ▴▴         | ▴▴         | ▴▴         | ▴▴           |
| ◂◂         | ▸▸         | ◂◂         | ◂◂         | ◂◂         | ◂◂           |
| ▴▴▴▴       | ▸▸▸▸       | ▴▴▴▴       | ▴▴▴▴       | ▴▴▴▴       | ▴▴▴▴         |
| ◂◂◂◂       | ▸▸▸▸       | ◂◂◂◂       | ◂◂◂◂       | ◂◂◂◂       | ◂◂◂◂         |
| ▴▴▴▴▴▴▴▴▴▴ | ▸▸▸▸▸▸▸▸▸▸ | ▴▴▴▴▴▴▴▴▴▴ | ▴▴▴▴▴▴▴▴▴▴ | ▴▴▴▴▴▴▴▴▴▴ | ▴▴▴▴▴▴▴▴▴▴   |
| ◂◂◂◂◂◂◂◂◂◂ | ▸▸▸▸▸▸▸▸▸▸ | ◂◂◂◂◂◂◂◂◂◂ | ◂◂◂◂◂◂◂◂◂◂ | ◂◂◂◂◂◂◂◂◂◂ | ◂◂◂◂◂◂◂◂◂◂   |
| markery    | o-o-o-o    | —(25, 5)   | —(0, -1)   | —(25, 5)   | —(0, 25, -1) |

## Colors

|         |           |         |           |         |         |             |              |             |              |
|---------|-----------|---------|-----------|---------|---------|-------------|--------------|-------------|--------------|
| C0      | C1        | C2      | C3        | C4      | C5      | C6          | C7           | C8          | C9           |
| b       | r         | c       | g         | m       | y       | k           | w            | 'x'         | 'Cn'         |
| DarkRed | Firebrick | Crimson | IndianRed | Salmon  | 'name'  | (1,0,0)     | (1,0,0,0.75) | (1,0,0,0.5) | (1,0,0,0.25) |
| #FF0000 | #FF008B   | #FF008B | #FF008B   | #FF008B | #FF008B | (R,G,B,[A]) | #RRGGBB      | #RRGGBB     | #RRGGBB[AA]  |

## Colormaps

|                    |  |          |          |        |  |
|--------------------|--|----------|----------|--------|--|
| plt.get_cmap(name) |  |          |          |        |  |
| Uniform            |  | viridis  | magma    | plasma |  |
| Sequential         |  | Greys    | YlOrBr   | Wistia |  |
| Diverging          |  | Spectral | coolwarm | RdGy   |  |
| Qualitative        |  | tab10    | tab20    |        |  |
| Cyclic             |  | twilight |          |        |  |

## Event handling

```
fig, ax = plt.subplots()
def on_click(event):
    print(event)
fig.canvas.mpl_connect('button_press_event', on_click)
```

## Animation

```
import matplotlib.animation as mpl

T = np.linspace(0, 2*np.pi, 100)
S = np.sin(T)
line, = plt.plot(T, S)
def animate(i):
    line.set_ydata(np.sin(T+i/50))
anim = mpl.FuncAnimation(
    plt.gcf(), animate, interval=5)
plt.show()
```

## Styles

|                      |  |  |  |
|----------------------|--|--|--|
| plt.style.use(style) |  |  |  |
|                      |  |  |  |
|                      |  |  |  |

## Quick reminder

```
ax.grid()
ax.set_[xy]lim(vmin, vmax)
ax.set_[xy]label(label)
ax.set_[xy]ticks(ticks, [labels])
ax.set_[xy]ticklabels(labels)
ax.set_title(title)
ax.tick_params(width=10, ...)
ax.set_axis_[on|off]()
```

```
fig.suptitle(title)
fig.tight_layout()
plt.gcf(), plt.gca()
mpl.rc('axes', linewidth=1, ...)
[fig|ax].patch.set_alpha(0)
text=r'$\frac{-e^{i\pi}}{2^n}$'
```

## Keyboard shortcuts

|        |                   |        |                   |
|--------|-------------------|--------|-------------------|
| ctrl+s | Save              | ctrl+w | Close plot        |
| r      | Reset view        | f      | Fullscreen 0/1    |
| v      | View forward      | b      | View back         |
| p      | Pan view          | o      | Zoom to rect      |
| x      | X pan/zoom        | y      | Y pan/zoom        |
| g      | Minor grid 0/1    | G      | Major grid 0/1    |
| l      | X axis log/linear | L      | Y axis log/linear |

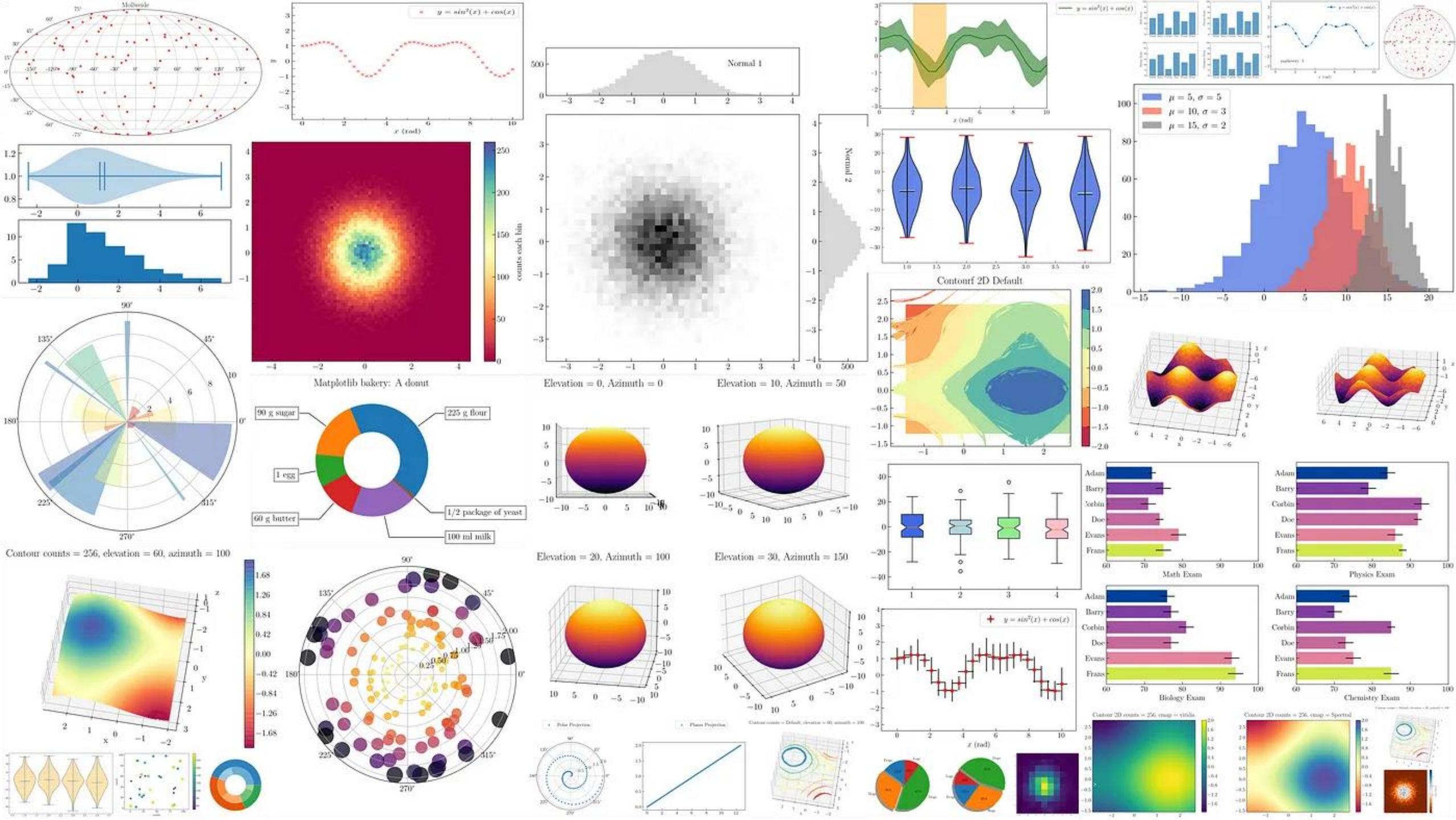
## Ten simple rules

1. Know your audience
2. Identify your message
3. Adapt the figure
4. Captions are not optional
5. Do not trust the defaults
6. Use color effectively
7. Do not mislead the reader
8. Avoid "chartjunk"
9. Message trumps beauty
10. Get the right tool

# Matplotlib

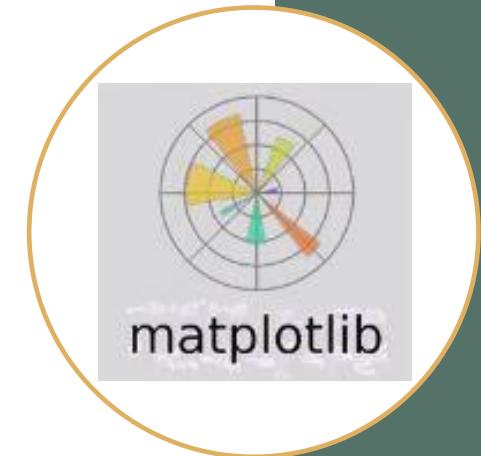
Najpopulárnejšia  
Python knižnica na  
**vytváranie grafov**





# Matplotlib

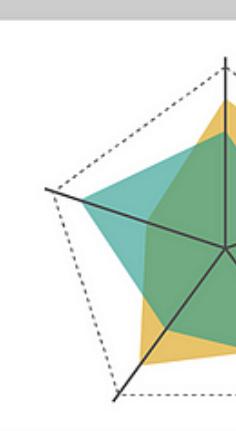
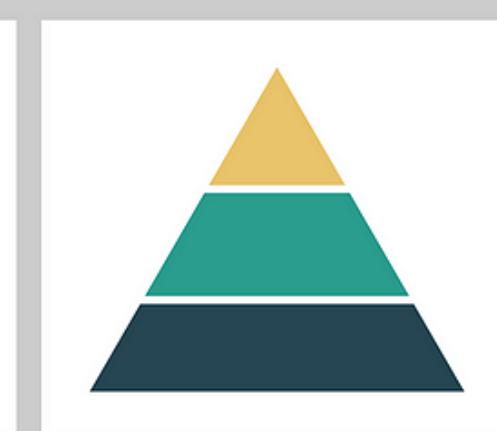
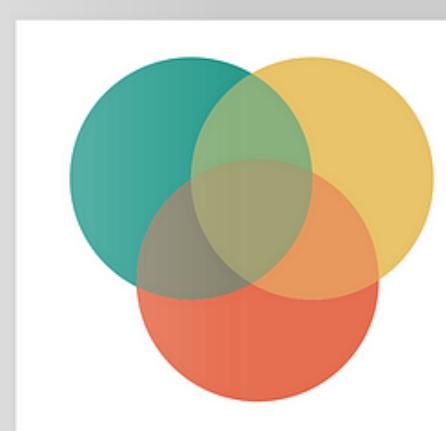
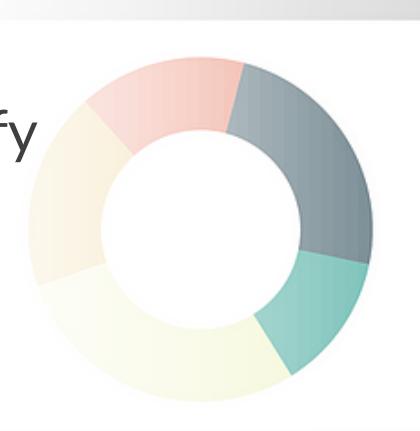
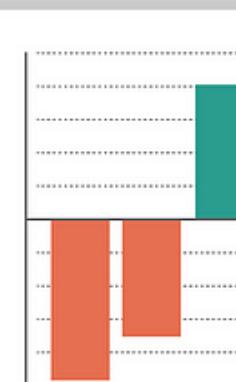
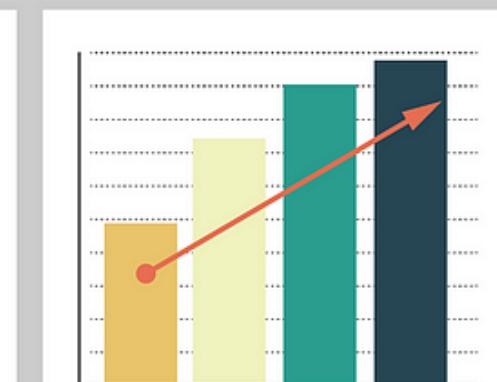
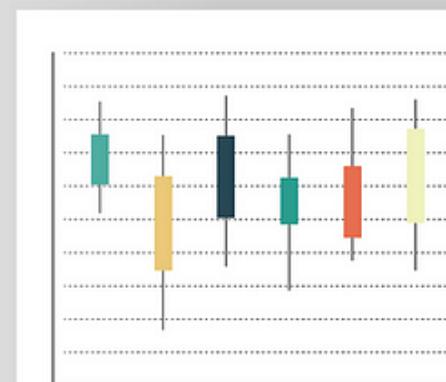
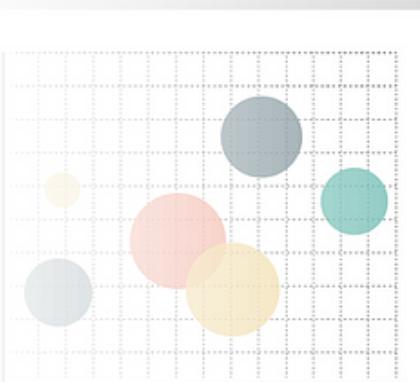
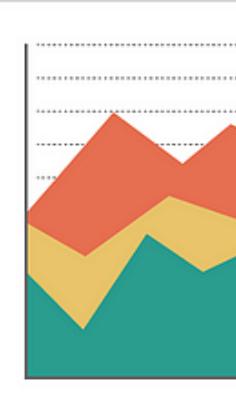
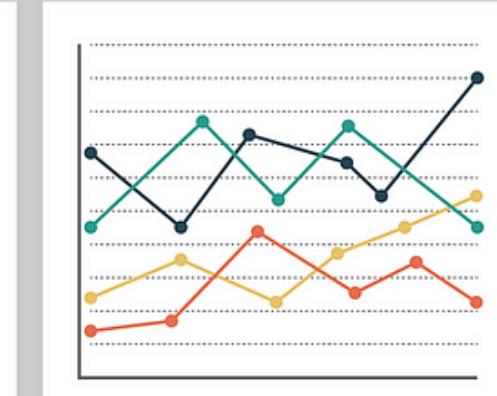
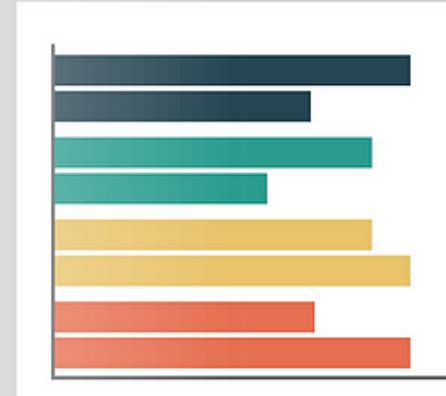
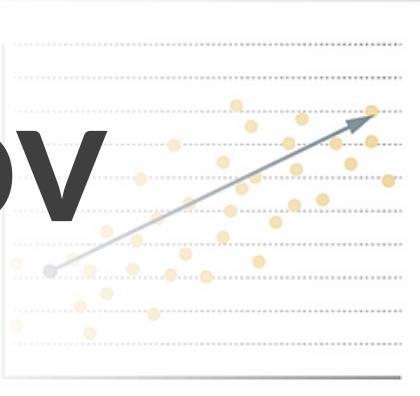
- Je nízkoúrovňová knižnica grafov
- Má syntax podobnú MATLABu



matplotlib

# Typy Grafov

1. Stípcové/Pruhované grafy
2. Histogramy
3. Bodové diagramy
4. Koláčové grafy
5. Prstencové grafy
6. Plošné/povrchové grafy
7. Krabicové/škatulové grafy
8. Bublinové grafy
9. Chybové úsečky/burzové grafy
10. Vennove diagramy
11. Radarové grafy
12. Lievikové/ihlanové grafy



## Quick start

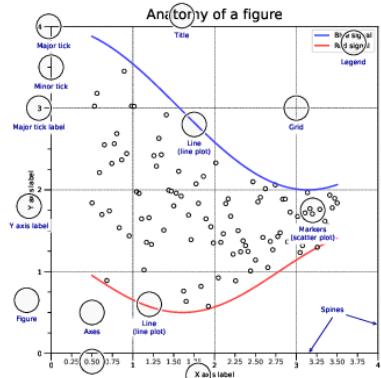
```
import numpy as np
import matplotlib as mpl
import matplotlib.pyplot as plt

X = np.linspace(0, 2*np.pi, 100)
Y = np.cos(X)

fig, ax = plt.subplots()
ax.plot(X, Y, color='green')

fig.savefig("figure.pdf")
plt.show()
```

## Anatomy of a figure



## Subplots layout

```
subplot(s)(rows,cols,...) API
fig, axs = plt.subplots(3, 3)

G = gridspec(rows,cols,...) API
ax = G[0,:]

ax.inset_axes(extent) API
d=make_axes_locatable(ax) API
ax = d.new_horizontal('10%')
```

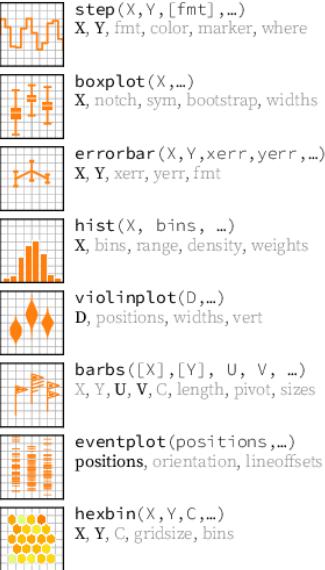
## Getting help

- [matplotlib.org](#)
- [github.com/matplotlib/matplotlib/issues](#)
- [discourse.matplotlib.org](#)
- [stackoverflow.com/questions/tagged/matplotlib](#)
- <https://gitter.im/matplotlib/matplotlib>
- [twitter.com/matplotlib](#)
- [Matplotlib users mailing list](#)

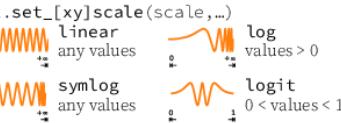
## Basic plots



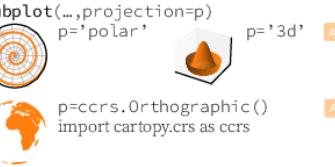
## Advanced plots



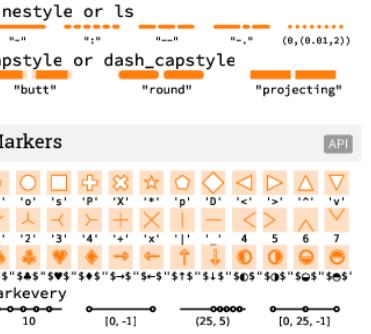
## Scales



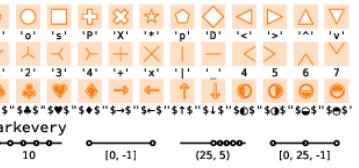
## Projections



## Lines



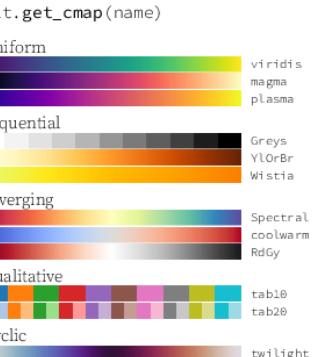
## Markers



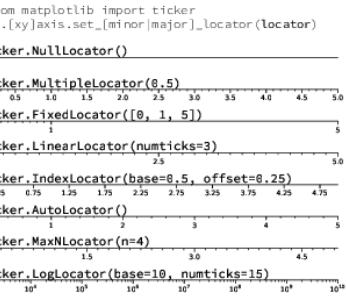
## Colors



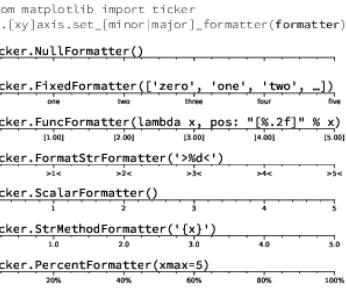
## Colormaps



## Tick locators

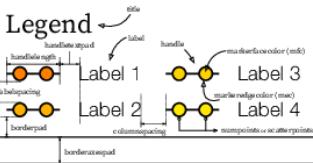


## Tick formatters



## Ornaments

```
ax.legend(...) API
handles, labels, loc, title, frameon
```



## Legend

```
ax.colorbar(...) API
mappable, ax, cax, orientation
```

```
ax.annotate(...) API
text, xy, xytext, xycoords, textcoords, arrowprops
```

```
text --> xy
textcoords --> xy
xycoords --> xy
```

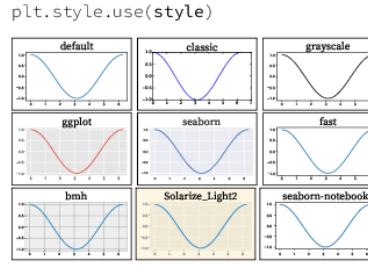
## Event handling

```
fig, ax = plt.subplots()
def on_click(event):
    print(event)
    fig.canvas.mpl_connect(
        'button_press_event', on_click)
```

## Animation

```
import matplotlib.animation as mpl
T = np.linspace(0, 2*np.pi, 100)
S = np.sin(T)
line, = plt.plot(T, S)
def animate(i):
    line.set_ydata(np.sin(T+i/50))
anim = mpl.FuncAnimation(
    plt.gcf(), animate, interval=5)
plt.show()
```

## Styles



## Quick reminder

```
ax.grid()
ax.set_[xy]lim(vmin, vmax)
ax.set_[xy]label(label)
ax.set_[xy]ticks(ticks, [labels])
ax.set_[xy]ticklabels(labels)
ax.set_title(title)
ax.tick_params(width=10, ...)
ax.set_axis_[on|off]()
```

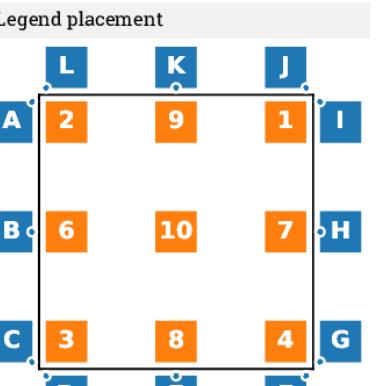
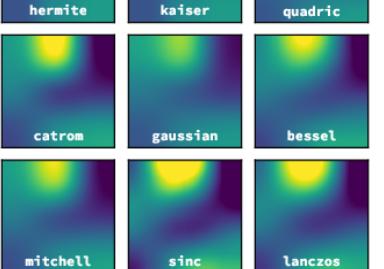
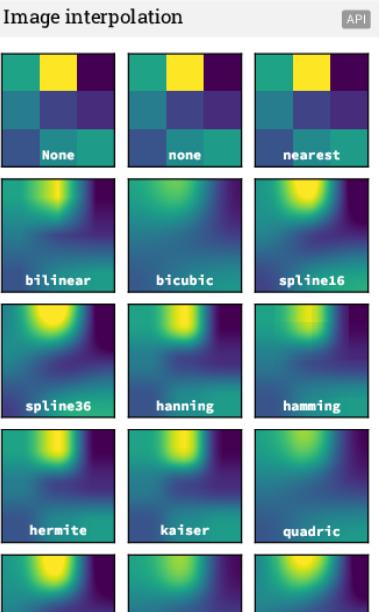
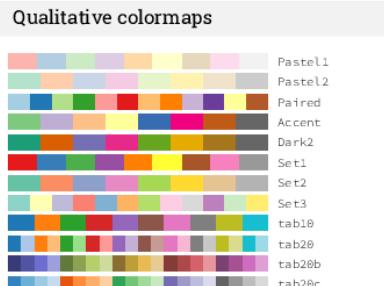
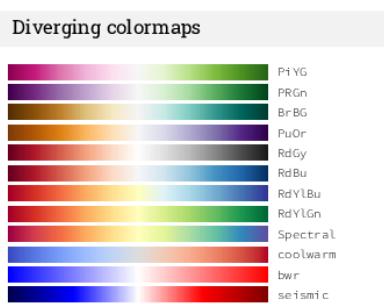
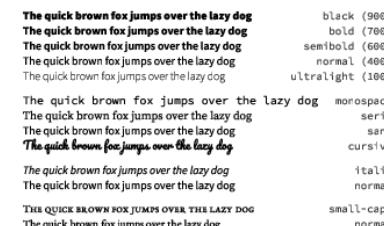
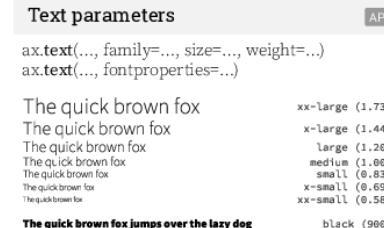
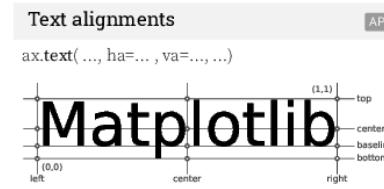
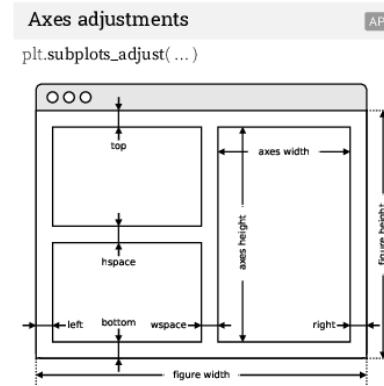
```
fig.suptitle(title)
fig.tight_layout()
plt.gcf(), plt.gca()
mpl.rc('axes', linewidth=1, ...)
[fig|ax].patch.set_alpha(0)
text=r'$\frac{-e^{i\pi}}{2^n}$'
```

## Keyboard shortcuts

|                        |                   |                        |                   |
|------------------------|-------------------|------------------------|-------------------|
| <b>ctrl</b> + <b>s</b> | Save              | <b>ctrl</b> + <b>w</b> | Close plot        |
| <b>r</b>               | Reset view        | <b>f</b>               | Fullscreen 0/1    |
| <b>f</b>               | View forward      | <b>b</b>               | View back         |
| <b>p</b>               | Pan view          | <b>o</b>               | Zoom to rect      |
| <b>x</b>               | X pan/zoom        | <b>y</b>               | Y pan/zoom        |
| <b>g</b>               | Minor grid 0/1    | <b>G</b>               | Major grid 0/1    |
| <b>l</b>               | X axis log/linear | <b>L</b>               | Y axis log/linear |

## Ten simple rules

1. Know your audience
2. Identify your message
3. Adapt the figure
4. Captions are not optional
5. Do not trust the defaults
6. Use color effectively
7. Do not mislead the reader
8. Avoid "chartjunk"
9. Message trumps beauty
10. Get the right tool

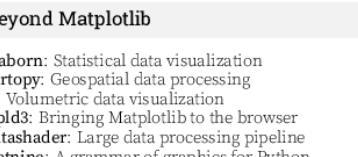
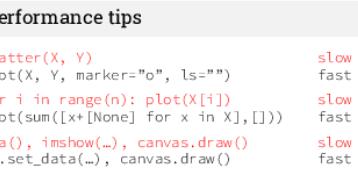
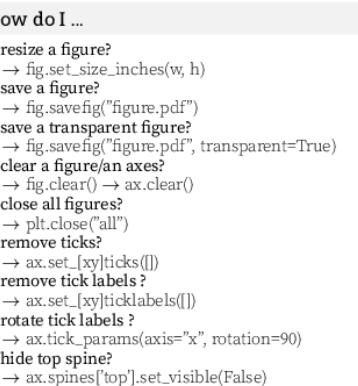


`ax.legend(loc="string", bbox_to_anchor=(x,y))`

|                |                 |                 |
|----------------|-----------------|-----------------|
| 2: upper left  | 9: upper center | 1: upper right  |
| 6: center left | 10: center      | 7: center right |
| 3: lower left  | 8: lower center | 4: lower right  |

A: upper right / (-0.1, 0.9)  
 C: lower right / (-0.1, 0.1)  
 E: upper center / (0.5, -0.1)  
 G: lower left / (1.1, 0.1)  
 I: upper left / (1.1, 0.9)  
 K: lower center / (0.5, 1.1)

B: center right / (-0.1, 0.5)  
 D: upper left / (0.1, -0.1)  
 F: upper right / (0.9, -0.1)  
 H: center left / (1.1, 0.5)  
 J: lower right / (0.9, 1.1)  
 L: lower left / (0.1, 1.1)



Matplotlib Cheatsheets  
 Copyright (c) 2021 Matplotlib Development Team  
 Released under a CC-BY 4.0 International License

# Matplotlib for beginners

Matplotlib is a library for making 2D plots in Python. It is designed with the philosophy that you should be able to create simple plots with just a few commands:

## 1 Initialize

```
import numpy as np
import matplotlib.pyplot as plt
```

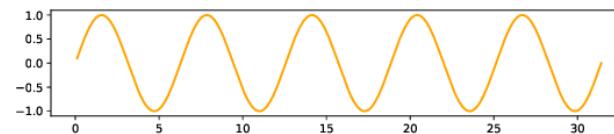
## 2 Prepare

```
X = np.linspace(0, 4*np.pi, 1000)
Y = np.sin(X)
```

## 3 Render

```
fig, ax = plt.subplots()
ax.plot(X, Y)
plt.show()
```

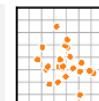
## 4 Observe



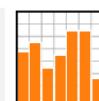
## Choose

Matplotlib offers several kind of plots (see Gallery):

```
X = np.random.uniform(0, 1, 100)
Y = np.random.uniform(0, 1, 100)
ax.scatter(X, Y)
```



```
X = np.arange(10)
Y = np.random.uniform(1, 10, 10)
ax.bar(X, Y)
```



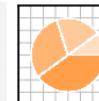
```
Z = np.random.uniform(0, 1, (8,8))
ax.imshow(Z)
```



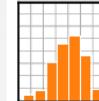
```
Z = np.random.uniform(0, 1, (8,8))
ax.contourf(Z)
```



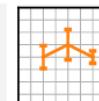
```
Z = np.random.uniform(0, 1, 4)
ax.pie(Z)
```



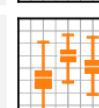
```
Z = np.random.normal(0, 1, 100)
ax.hist(Z)
```



```
X = np.arange(5)
Y = np.random.uniform(0, 1, 5)
ax.errorbar(X, Y, Y/4)
```



```
Z = np.random.normal(0, 1, (100,3))
ax.boxplot(Z)
```



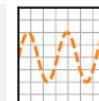
## Tweak

You can modify pretty much anything in a plot, including limits, colors, markers, line width and styles, ticks and ticks labels, titles, etc.

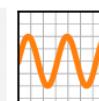
```
X = np.linspace(0, 10, 100)
Y = np.sin(X)
ax.plot(X, Y, color="black")
```



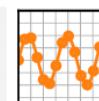
```
X = np.linspace(0, 10, 100)
Y = np.sin(X)
ax.plot(X, Y, linestyle="--")
```



```
X = np.linspace(0, 10, 100)
Y = np.sin(X)
ax.plot(X, Y, linewidth=5)
```



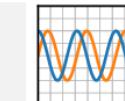
```
X = np.linspace(0, 10, 100)
Y = np.sin(X)
ax.plot(X, Y, marker="o")
```



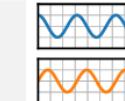
## Organize

You can plot several data on the same figure, but you can also split a figure in several subplots (named Axes):

```
X = np.linspace(0, 10, 100)
Y1, Y2 = np.sin(X), np.cos(X)
ax.plot(X, Y1, X, Y2)
```



```
fig, (ax1, ax2) = plt.subplots(2,1)
ax1.plot(X, Y1, color="C1")
ax2.plot(X, Y2, color="C0")
```



```
fig, (ax1, ax2) = plt.subplots(1,2)
ax1.plot(Y1, X, color="C1")
ax2.plot(Y2, X, color="C0")
```

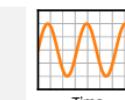


## Label (everything)

```
ax.plot(X, Y)
fig.suptitle(None)
ax.set_title("A Sine wave")
```



```
ax.plot(X, Y)
ax.set_ylabel(None)
ax.set_xlabel("Time")
```



## Explore

Figures are shown with a graphical user interface that allows to zoom and pan the figure, to navigate between the different views and to show the value under the mouse.

## Save (bitmap or vector format)

```
fig.savefig("my-first-figure.png", dpi=300)
fig.savefig("my-first-figure.pdf")
```

# Ako Tvoríť Grafy v Matplotlib?

## Objektovo-Orientovaný (OO) štýl

- Explicitne vytvoríte vizualizácie (Figures) a osi a voláte na nich metódy

```
x = np.cos(np.linspace(0, 2, 100))
```

```
fig, ax = plt.subplots()
ax.plot(x, x, label='linear')
ax.plot(x, x**2, label='quadratic')
ax.plot(x, x**3, label='cubic')
ax.set_xlabel('x label')
ax.set_ylabel('y label')
ax.set_title("Simple Plot")
ax.legend();
```

## Pyplot štýl

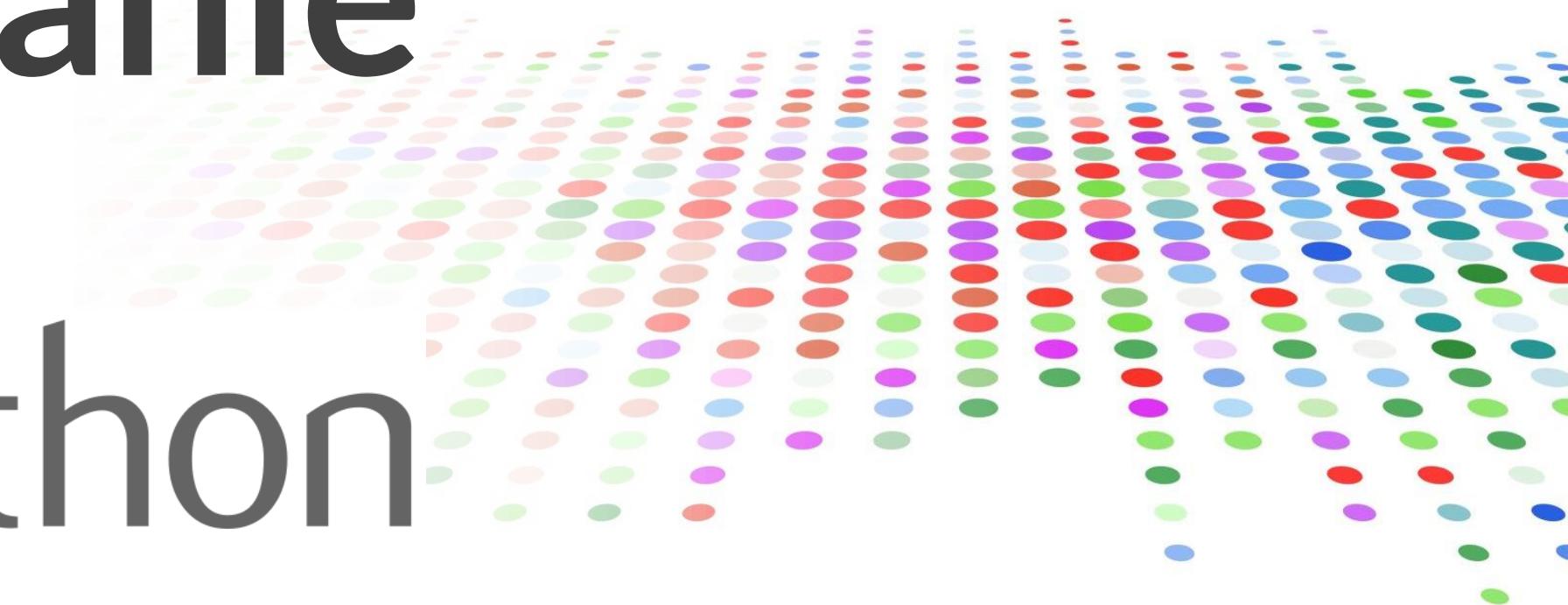
- Spoliehate sa na modul pyplot, ktorý automaticky vytvára a spravuje vizualizácie (Figures) a osi

```
x = np.cos(np.linspace(0, 2, 100))
plt.plot(x, x, label='linear')
plt.plot(x, x**2, label='quadratic')
plt.plot(x, x**3, label='cubic')
plt.xlabel('x label')
plt.ylabel('y label')
plt.title("Simple Plot")
plt.legend();
```

# Grafy Štýlovanie



python



# matplotlib

Cheat sheet  
Version 3.5.0

71

## Quick start

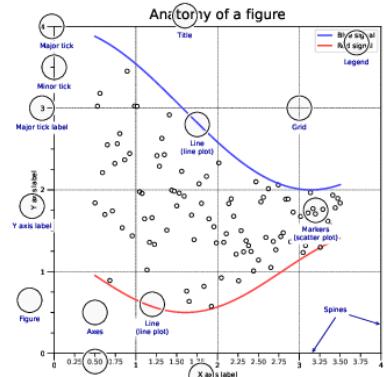
```
import numpy as np
import matplotlib as mpl
import matplotlib.pyplot as plt

X = np.linspace(0, 2*np.pi, 100)
Y = np.cos(X)

fig, ax = plt.subplots()
ax.plot(X, Y, color='green')

fig.savefig("figure.pdf")
plt.show()
```

## Anatomy of a figure



## Subplots layout

```
subplot[s](rows,cols,...)
fig, axs = plt.subplots(3, 3)

G = gridspec(rows,cols,...)
ax = G[0,:]

ax.inset_axes(extent)

d=make_axes_locatable(ax)
ax = d.new_horizontal('10%')
```

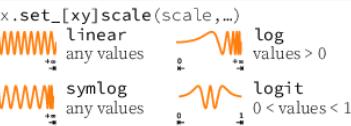
## Getting help

- [matplotlib.org](#)
- [github.com/matplotlib/matplotlib/issues](#)
- [discourse.matplotlib.org](#)
- [stackoverflow.com/questions/tagged/matplotlib](#)
- <https://gitter.im/matplotlib/matplotlib>
- [twitter.com/matplotlib](#)
- [Matplotlib users mailing list](#)

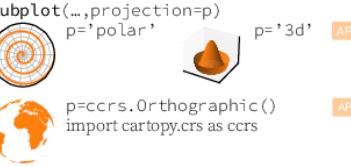
## Basic plots



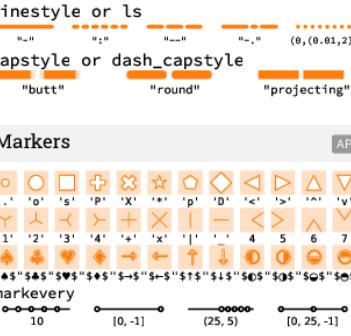
## Scales



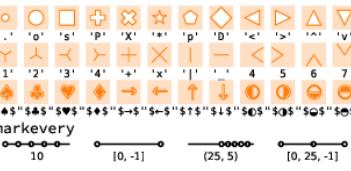
## Projections



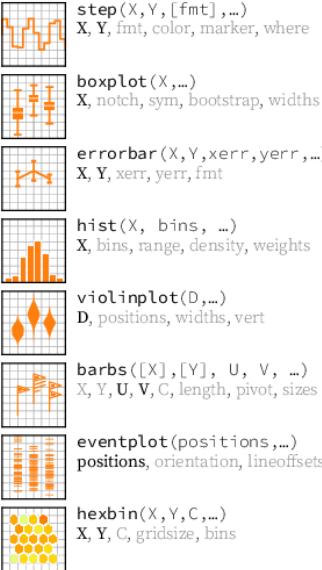
## Lines



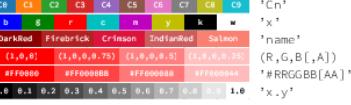
## Markers



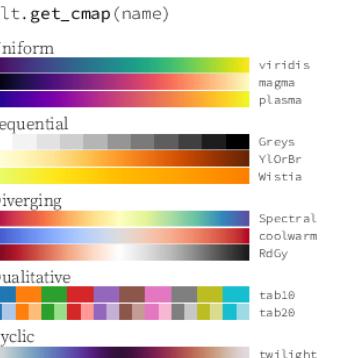
## Advanced plots



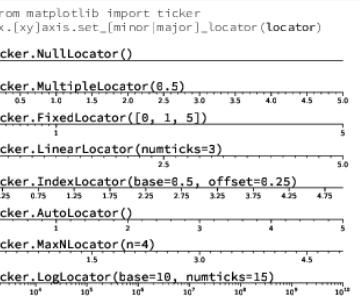
## Colors



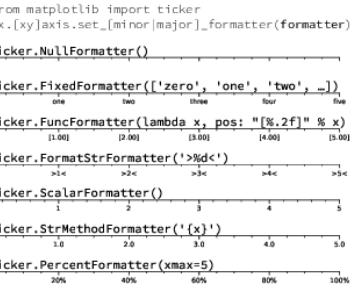
## Colormaps



## Tick locators

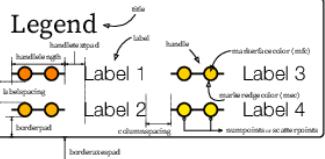


## Tick formatters



## Ornaments

ax.legend(...)  
handles, labels, loc, title, frameon



## ax.colorbar(...)

mappable, ax, cax, orientation

## ax.annotate(...)

text, xy, xytext, xycoords, textcoords, arrowprops



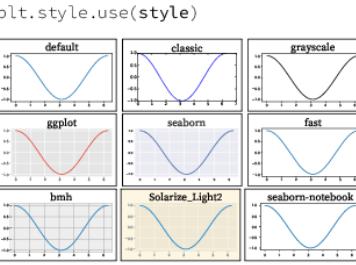
## Event handling

```
fig, ax = plt.subplots()
def on_click(event):
    print(event)
fig.canvas.mpl_connect('button_press_event', on_click)
```

## Animation

```
import matplotlib.animation as mpl
T = np.linspace(0, 2*np.pi, 100)
S = np.sin(T)
line, = plt.plot(T, S)
def animate(i):
    line.set_ydata(np.sin(T+i/50))
anim = mpl.FuncAnimation(
    plt.gca(), animate, interval=5)
plt.show()
```

## Styles



## Quick reminder

```
ax.grid()
ax.set_[xy]lim(vmin, vmax)
ax.set_[xy]label(label)
ax.set_[xy]ticks(ticks, [labels])
ax.set_[xy]ticklabels(labels)
ax.set_title(title)
ax.tick_params(width=10, ...)
ax.set_axis_[on|off]()
```

```
fig.suptitle(title)
fig.tight_layout()
plt.gcf(), plt.gca()
mpl.rc('axes', linewidth=1, ...)
[fig|ax].patch.set_alpha(0)
text=r'$\frac{-e^{i\pi}}{2^n}$'
```

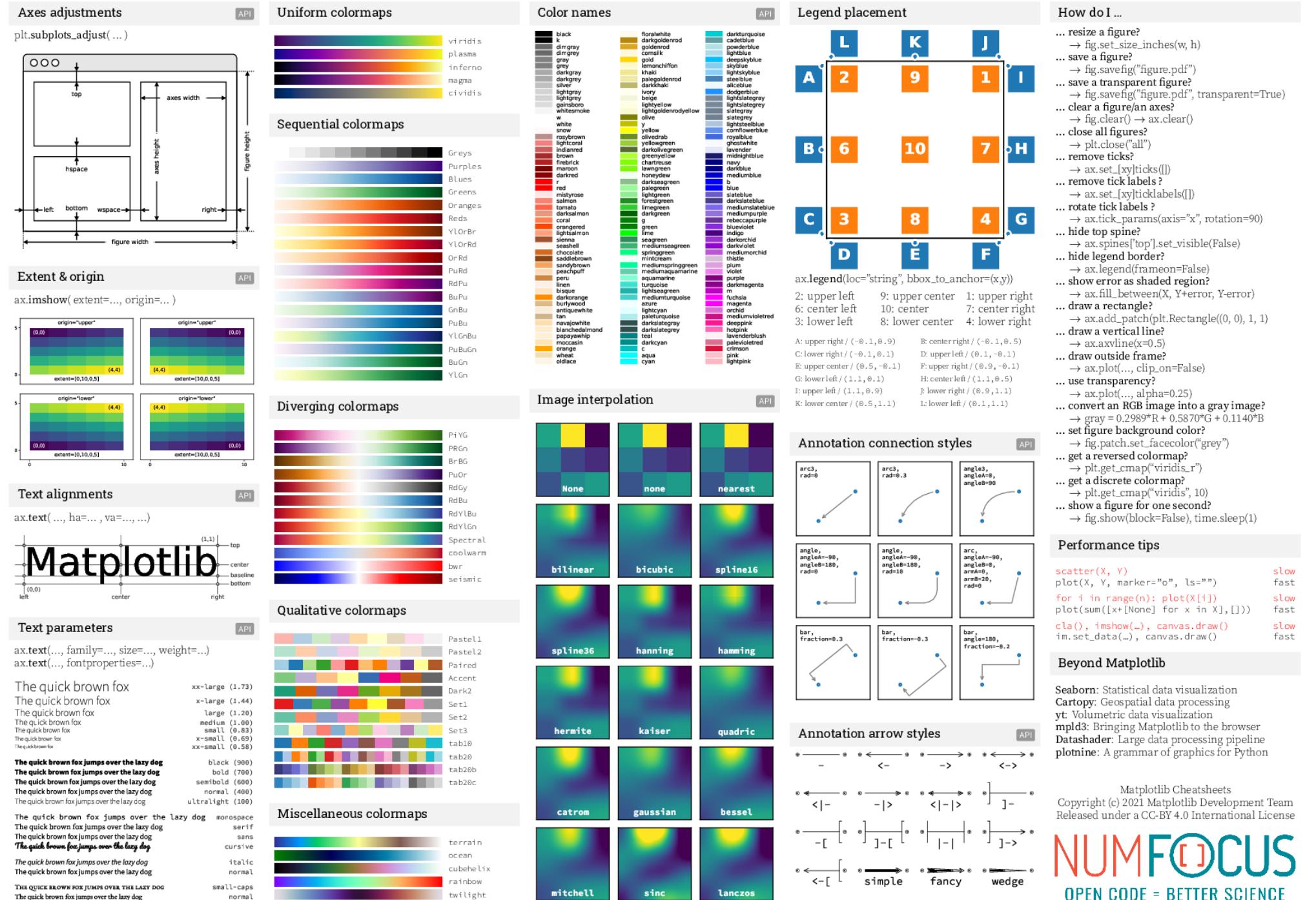
## Keyboard shortcuts

|        |                   |        |                   |
|--------|-------------------|--------|-------------------|
| ctrl+s | Save              | ctrl+w | Close plot        |
| r      | Reset view        | f      | Fullscreen 0/1    |
| v      | View forward      | b      | View back         |
| p      | Pan view          | o      | Zoom to rect      |
| x      | X pan/zoom        | y      | Y pan/zoom        |
| g      | Minor grid 0/1    | G      | Major grid 0/1    |
| l      | X axis log/linear | L      | Y axis log/linear |

## Ten simple rules

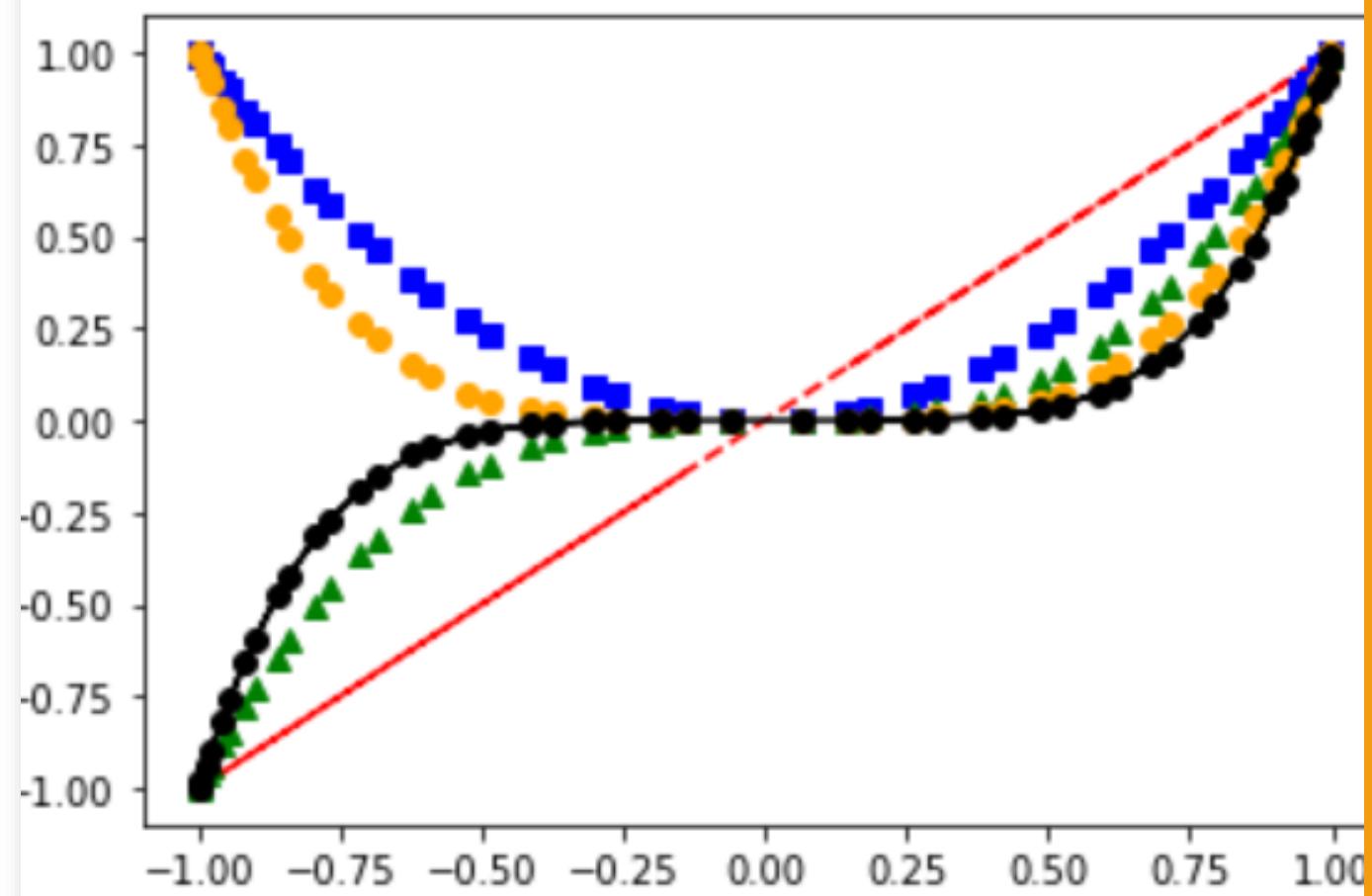
1. Know your audience
2. Identify your message
3. Adapt the figure
4. Captions are not optional
5. Do not trust the defaults
6. Use color effectively
7. Do not mislead the reader
8. Avoid "chartjunk"
9. Message trumps beauty
10. Get the right tool

READ

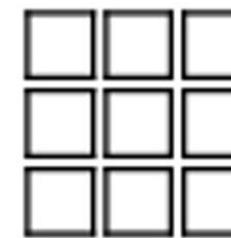
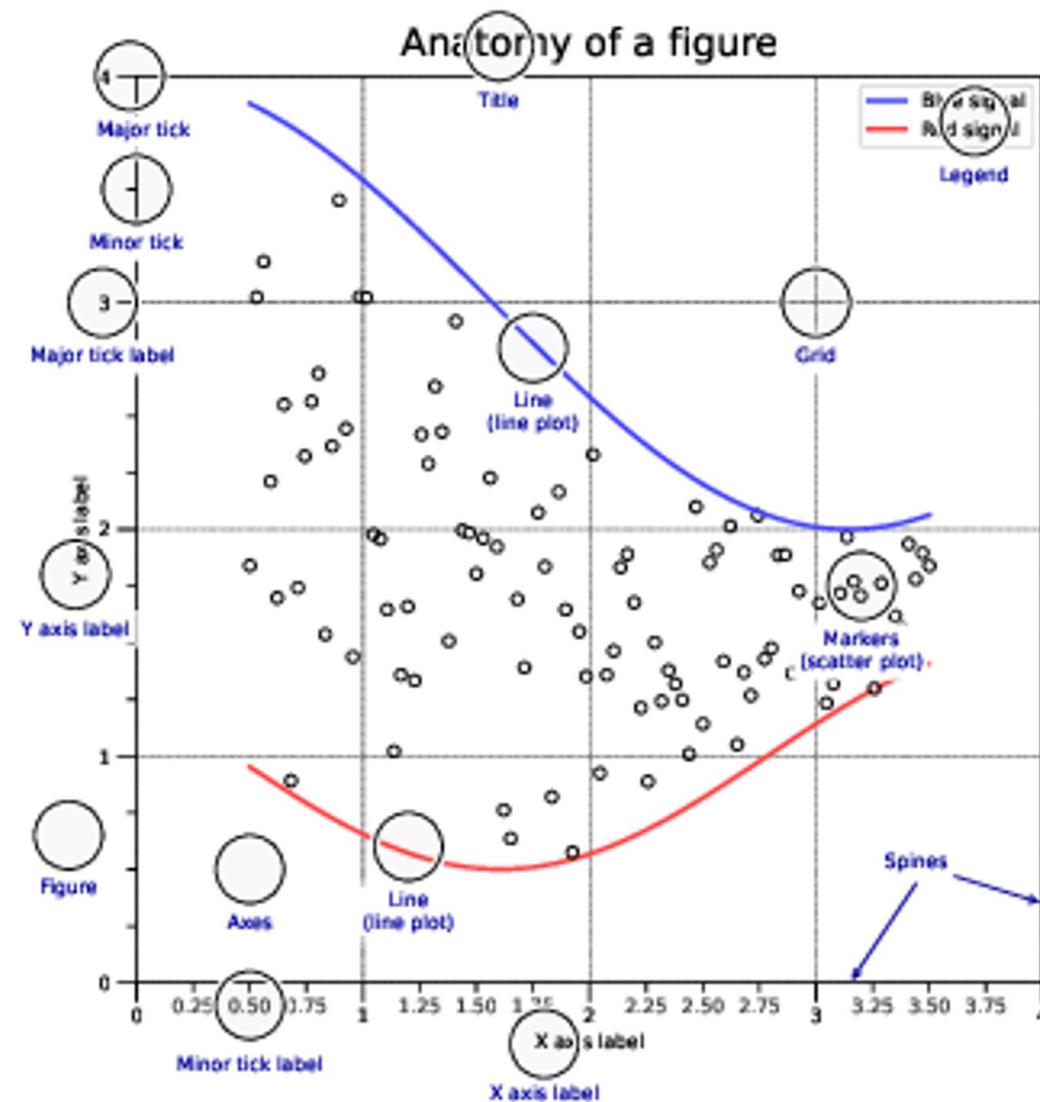


# Formátovanie štýlu vášho grafu

```
1. import matplotlib.pyplot as plt  
2. import numpy as np  
3. t = np.sin(np.linspace(-3,3,50))  
4. # červené čiarky, modré štvorce a zelené trojuholníky  
5. plt.plot(t, t, 'r--', color="red")  
6. plt.plot(t, t**2, 'bs', color="blue")  
7. plt.plot(t, t**3, 'g^', color="green")  
8. plt.plot(t, t**4, "o", color="orange")  
9. plt.plot(t, t**5, "o-", color="black")  
10. plt.show()
```



# Anatómia (Štruktúra) Grafu



`subplot[s](rows,cols,...)` API  
`fig, axs = plt.subplots(3, 3)`



`G = gridspec(rows,cols,...)` API  
`ax = G[0, :]`



`ax.inset_axes(extent)` API



`d=make_axes_locatable(ax)` API  
`ax = d.new_horizontal('10%)')`

# - Register Účtovných Závierok RUZ



python



 Skryť navigáciu

## ORGÁNY VEREJNEJ MOCI

EGOVERNMENT  
KOMPONENTY ISVS

Koncová služba &gt;

Proces

Aplikačná služba &gt;

Infraštruktúrna služba

Celkové náklady na  
vlastníctvo

Monitoring &gt;

## KRIT, ŠTÚDIE, PROJEKTY

## INTEGRÁCIE A SLA

INFORMAČNO KOMUNIKÁČNÉ  
TECHNOLÓGIEŠTANDARDIZÁCIA A DÁTOVÉ  
OBJEKTY[Hlavná stránka](#) » [ISVS:](#) » Register účtovných závierok ([www.registeruz.sk](http://www.registeruz.sk))

## ISVS: Register účtovných závierok ([www.registeruz.sk](http://www.registeruz.sk))

[Základné informácie](#)[Detailné informácie](#)[Vzťahy](#)[Dokumenty](#)[Integračné väzby](#)[Dostupnosť](#)[Licencie](#)**Správca:**[Ministerstvo financií Slovenskej republiky](#)**Názov informačného systému:**Register účtovných závierok ([www.registeruz.sk](http://www.registeruz.sk))**Anglický názov:****Popis:**

Modul Registeruz.sk alebo aj Register účtovných závierok poskytuje informácie o účtovných jednotkách a účtovných závierkach ako súboroch účtovných výkazov zostavovaných týmito účtovnými jednotkami. Bol vytvorený na základe zákona č. 431/2002 Z. z. o účtovníctve v znení neskorších predpisov a prevádzkuje ho DataCentrum z poverenia Ministerstva financií SR.

**Anglický popis:****Poznámka:****Kód MetaIS:**

isvs\_7238

**Referencovateľný identifikátor:**<https://data.gov.sk/id/egov/isvs/7238>

# Register účtovných závierok

Ministerstvo financií Slovenskej republiky



Zadajte názov účtovnej jednotky, IČO alebo DIČ

VYHĽADAŤ

[ZOBRAZIŤ VIAC MOŽNOSTÍ VYHĽADÁVANIA](#)



# Register účtovných závierok

Ministerstvo financií Slovenskej republiky

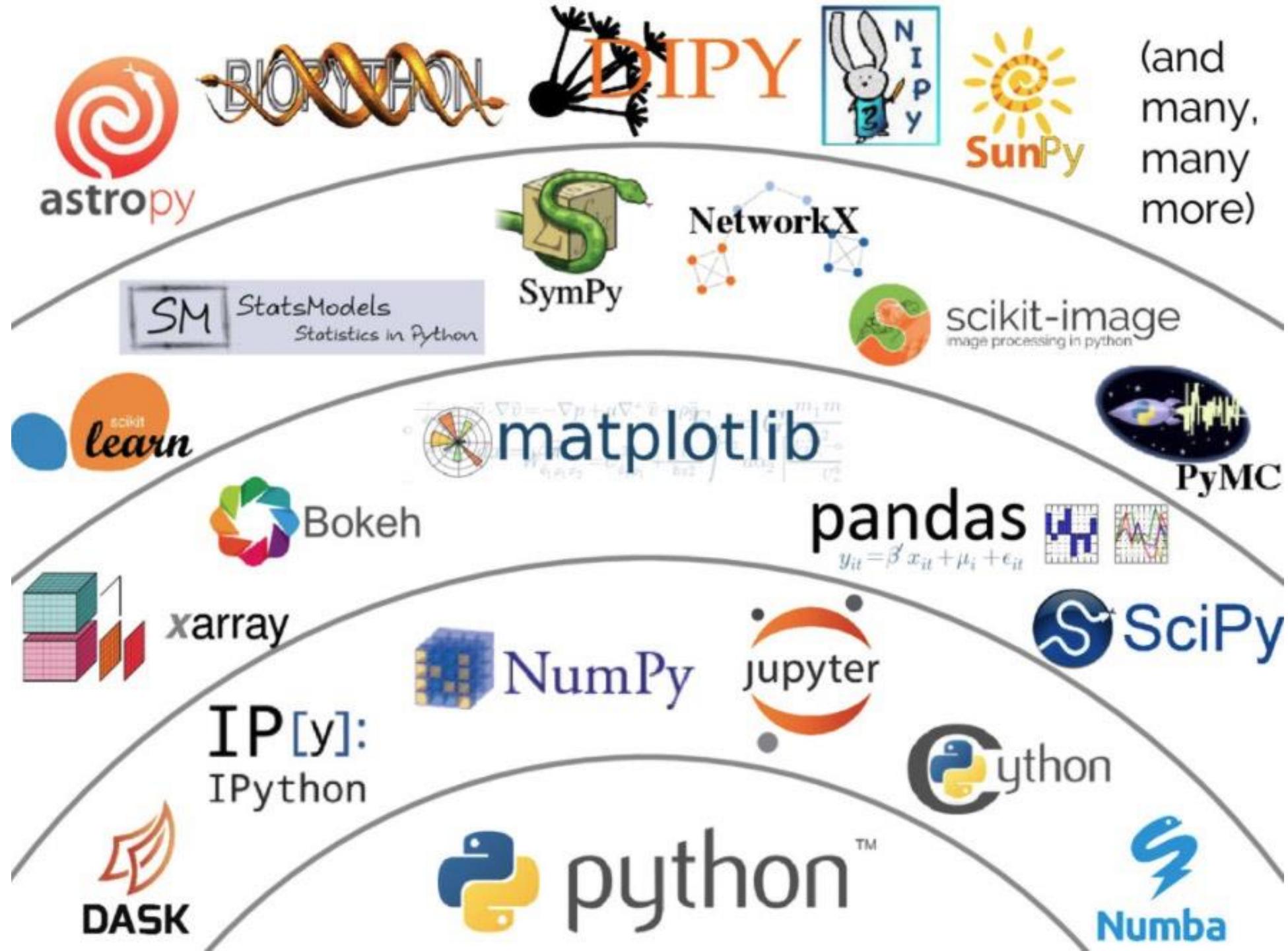


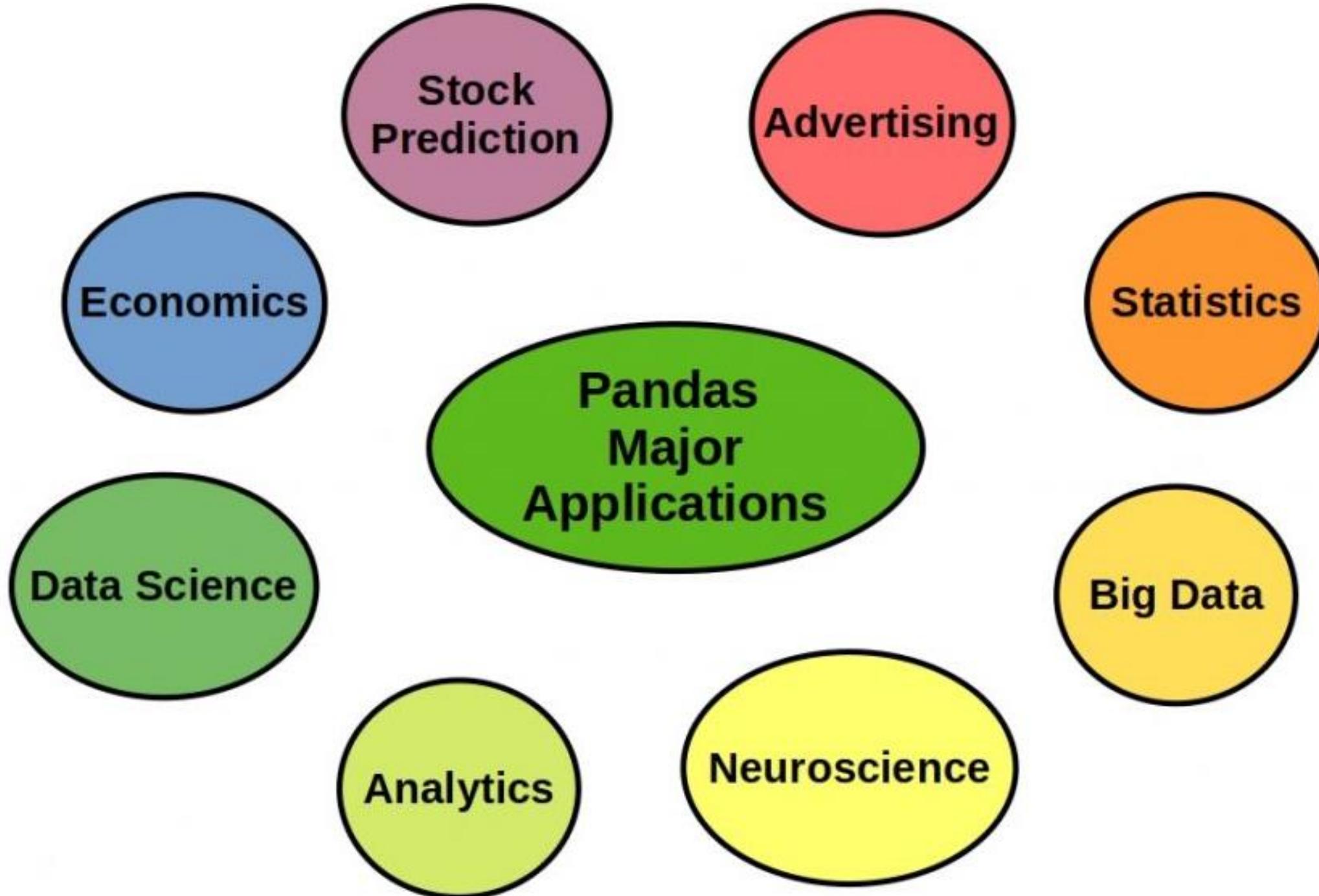
Stránka je dočasne nedostupná. Ospravedlňujeme sa za spôsobené komplikácie.



# Pandas







## Jupyter format

|   | YEARMODA | TEMP | MAX  | MIN  |
|---|----------|------|------|------|
| 0 | 20160601 | 65.5 | 73.6 | 54.7 |
| 1 | 20160602 | 65.8 | 80.8 | 55.0 |
| 2 | 20160603 | 68.4 | 77.9 | 55.6 |
| 3 | 20160604 | 57.5 | 70.9 | 47.3 |
| 4 | 20160605 | 51.4 | 58.3 | 43.2 |
| 5 | 20160606 | 52.2 | 59.7 | 42.8 |
| 6 | 20160607 | 56.9 | 65.1 | 45.9 |
| 7 | 20160608 | 54.2 | 60.4 | 47.5 |
| 8 | 20160609 | 49.4 | 54.1 | 45.7 |
| 9 | 20160610 | 49.5 | 55.9 | 43.0 |

Column labels

Data

Index

Pandas DataFrame

pandas.core.frame.DataFrame

## Standard Python format

|   | YEARMODA | TEMP | MAX  | MIN  |
|---|----------|------|------|------|
| 0 | 20160601 | 65.5 | 73.6 | 54.7 |
| 1 | 20160602 | 65.8 | 80.8 | 55.0 |
| 2 | 20160603 | 68.4 | 77.9 | 55.6 |
| 3 | 20160604 | 57.5 | 70.9 | 47.3 |
| 4 | 20160605 | 51.4 | 58.3 | 43.2 |
| 5 | 20160606 | 52.2 | 59.7 | 42.8 |
| 6 | 20160607 | 56.9 | 65.1 | 45.9 |
| 7 | 20160608 | 54.2 | 60.4 | 47.5 |
| 8 | 20160609 | 49.4 | 54.1 | 45.7 |
| 9 | 20160610 | 49.5 | 55.9 | 43.0 |

Index

Pandas Series

## Standard Python format

Index

0 65.5

1 65.8

2 68.4

3 57.5

4 51.4

5 52.2

6 56.9

7 54.2

8 49.4

9 49.5

Name: TEMP, dtype: float64

Data

Column label

Data type

Pandas Series

pandas.core.series.Series

Column Label/ Header

0 1 2 3 4

Index Label

Name Age Marks Grade Hobby

Column Index

|   |    |        |    |       |   |          |
|---|----|--------|----|-------|---|----------|
| 0 | S1 | Joe    | 20 | 85.10 | A | Swimming |
| 1 | S2 | Nat    | 21 | 77.80 | B | Reading  |
| 2 | S3 | Harry  | 19 | 91.54 | A | Music    |
| 3 | S4 | Sam    | 20 | 88.78 | A | Painting |
| 4 | S5 | Monica | 22 | 60.55 | B | Dancing  |

Row Index

Column

Element/ Value/ Entry

Row

# Series

# Series

# DataFrame

|   | apples |
|---|--------|
| 0 | 3      |
| 1 | 2      |
| 2 | 0      |
| 3 | 1      |

|   | oranges |
|---|---------|
| 0 | 0       |
| 1 | 3       |
| 2 | 7       |
| 3 | 2       |

|   | apples | oranges |
|---|--------|---------|
| 0 | 3      | 0       |
| 1 | 2      | 3       |
| 2 | 0      | 7       |
| 3 | 1      | 2       |

Series 1

Series 2

Series 3

DataFrame

|   | Mango |
|---|-------|
| 0 | 4     |
| 1 | 5     |
| 2 | 6     |
| 3 | 3     |
| 4 | 1     |

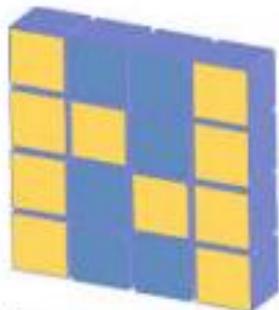
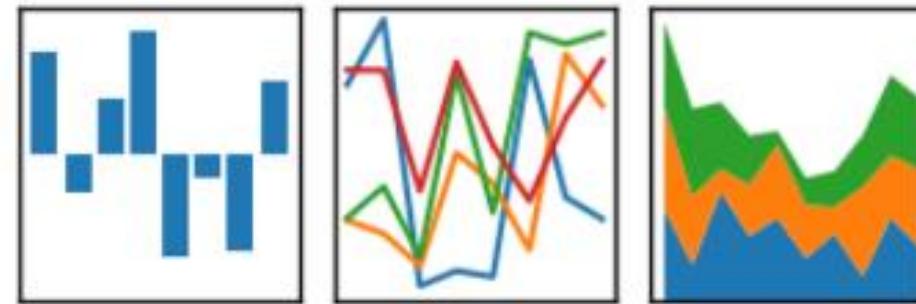
|   | Apple |
|---|-------|
| 0 | 5     |
| 1 | 4     |
| 2 | 3     |
| 3 | 0     |
| 4 | 2     |

|   | Banana |
|---|--------|
| 0 | 2      |
| 1 | 3      |
| 2 | 5      |
| 3 | 2      |
| 4 | 7      |

|   | Mango | Apple | Banana |
|---|-------|-------|--------|
| 0 | 4     | 5     | 2      |
| 1 | 5     | 4     | 3      |
| 2 | 6     | 3     | 5      |
| 3 | 3     | 0     | 2      |
| 4 | 1     | 2     | 7      |

# pandas

$$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$$



# NumPy



python™

| Pandas dtype  | Python type  | NumPy type   | Usage  |
|---------------|--------------|--|--|
| object        | str or mixed | string_, unicode_, mixed types                                 | Text or mixed numeric and non-numeric values |
| int64         | int          | int_, int8, int16, int32, int64, uint8, uint16, uint32, uint64 | Integer numbers                              |
| float64       | float        | float_, float16, float32, float64                              | Floating point numbers                       |
| bool          | bool         | bool_  | True/False values                            |
| datetime64    | NA           | datetime64[ns]   | Date and time values                         |
| timedelta[ns] | NA           | NA   | Differences between two datetimes            |
| category      | NA           | NA   | Finite list of text values                   |

# Pandas Knižnica

 python



## Jupyter format

|   | YEARMODA | TEMP | MAX  | MIN  |
|---|----------|------|------|------|
| 0 | 20160601 | 65.5 | 73.6 | 54.7 |
| 1 | 20160602 | 65.8 | 80.8 | 55.0 |
| 2 | 20160603 | 68.4 | 77.9 | 55.6 |
| 3 | 20160604 | 57.5 | 70.9 | 47.3 |
| 4 | 20160605 | 51.4 | 58.3 | 43.2 |
| 5 | 20160606 | 52.2 | 59.7 | 42.8 |
| 6 | 20160607 | 56.9 | 65.1 | 45.9 |
| 7 | 20160608 | 54.2 | 60.4 | 47.5 |
| 8 | 20160609 | 49.4 | 54.1 | 45.7 |
| 9 | 20160610 | 49.5 | 55.9 | 43.0 |

Column labels

Data

Index

Pandas DataFrame

pandas.core.frame.DataFrame

## Standard Python format

|   | YEARMODA | TEMP | MAX  | MIN  |
|---|----------|------|------|------|
| 0 | 20160601 | 65.5 | 73.6 | 54.7 |
| 1 | 20160602 | 65.8 | 80.8 | 55.0 |
| 2 | 20160603 | 68.4 | 77.9 | 55.6 |
| 3 | 20160604 | 57.5 | 70.9 | 47.3 |
| 4 | 20160605 | 51.4 | 58.3 | 43.2 |
| 5 | 20160606 | 52.2 | 59.7 | 42.8 |
| 6 | 20160607 | 56.9 | 65.1 | 45.9 |
| 7 | 20160608 | 54.2 | 60.4 | 47.5 |
| 8 | 20160609 | 49.4 | 54.1 | 45.7 |
| 9 | 20160610 | 49.5 | 55.9 | 43.0 |

Index

Data

Pandas Series

## Standard Python format

Index

0

1

2

3

4

5

6

7

8

9

Name: TEMP, dtype: float64

Column label

Data type

Pandas Series

pandas.core.series.Series

Column Label/ Header

0 1 2 3 4

Index Label

Name Age Marks Grade Hobby

Column Index

|   |    |        |    |       |   |          |
|---|----|--------|----|-------|---|----------|
| 0 | S1 | Joe    | 20 | 85.10 | A | Swimming |
| 1 | S2 | Nat    | 21 | 77.80 | B | Reading  |
| 2 | S3 | Harry  | 19 | 91.54 | A | Music    |
| 3 | S4 | Sam    | 20 | 88.78 | A | Painting |
| 4 | S5 | Monica | 22 | 60.55 | B | Dancing  |

Row Index

Column

Element/ Value/ Entry

Row

# Series

# Series

# DataFrame

|   | apples |
|---|--------|
| 0 | 3      |
| 1 | 2      |
| 2 | 0      |
| 3 | 1      |

|   | oranges |
|---|---------|
| 0 | 0       |
| 1 | 3       |
| 2 | 7       |
| 3 | 2       |

|   | apples | oranges |
|---|--------|---------|
| 0 | 3      | 0       |
| 1 | 2      | 3       |
| 2 | 0      | 7       |
| 3 | 1      | 2       |

Series 1

Series 2

Series 3

DataFrame

|   | Mango |
|---|-------|
| 0 | 4     |
| 1 | 5     |
| 2 | 6     |
| 3 | 3     |
| 4 | 1     |

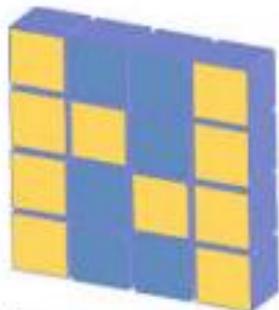
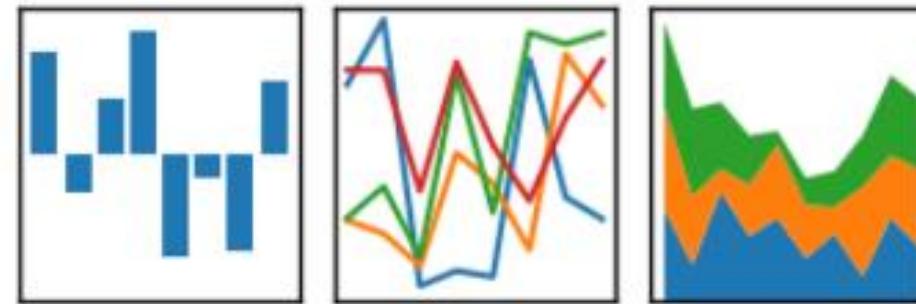
|   | Apple |
|---|-------|
| 0 | 5     |
| 1 | 4     |
| 2 | 3     |
| 3 | 0     |
| 4 | 2     |

|   | Banana |
|---|--------|
| 0 | 2      |
| 1 | 3      |
| 2 | 5      |
| 3 | 2      |
| 4 | 7      |

|   | Mango | Apple | Banana |
|---|-------|-------|--------|
| 0 | 4     | 5     | 2      |
| 1 | 5     | 4     | 3      |
| 2 | 6     | 3     | 5      |
| 3 | 3     | 0     | 2      |
| 4 | 1     | 2     | 7      |

# pandas

$$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$$



# NumPy



python™

| Pandas dtype  | Python type  | NumPy type   | Usage  |
|---------------|--------------|--|--|
| object        | str or mixed | string_, unicode_, mixed types                                 | Text or mixed numeric and non-numeric values |
| int64         | int          | int_, int8, int16, int32, int64, uint8, uint16, uint32, uint64 | Integer numbers                              |
| float64       | float        | float_, float16, float32, float64                              | Floating point numbers                       |
| bool          | bool         | bool_  | True/False values                            |
| datetime64    | NA           | datetime64[ns]   | Date and time values                         |
| timedelta[ns] | NA           | NA   | Differences between two datetimes            |
| category      | NA           | NA   | Finite list of text values                   |

# Pandas

## Important Methods *Pandas* Packages

### Data Importing

```
pd.read_csv()  
pd.read_table()  
pd.read_excel()  
pd.read_sql()  
pd.read_json()  
pd.read_html()  
pd.DataFrame()  
pd.concat()  
pd.Series()  
pd.date_range()
```



### Data Cleaning

```
pd.fillna()  
pd.dropna()  
pd.sort_values()  
pd.apply()  
pd.groupby()  
pd.append()  
pd.join()  
pd.rename()  
pd.to_csv()  
pd.set_index()
```

### Data Statistic

```
pd.head()  
pd.tail()  
pd.describe()  
pd.info()  
pd.mean()  
pd.median()  
pd.count()  
pd.std()  
pd.max()  
pd.min()
```

 **bratislava.sk**  22 current visitors

Q Filter

Last 30 days 

v **vita.sk** ▾ ● 4 current visitors

Q Filter

Last 12 months

▼

UNIQUE VISITORS**112k**

↑ 79%

TOTAL VISITS**130k**

↑ 88%

TOTAL PAGEVIEWS**500k**

↑ 68%

VIEWS PER VISIT**3.95**

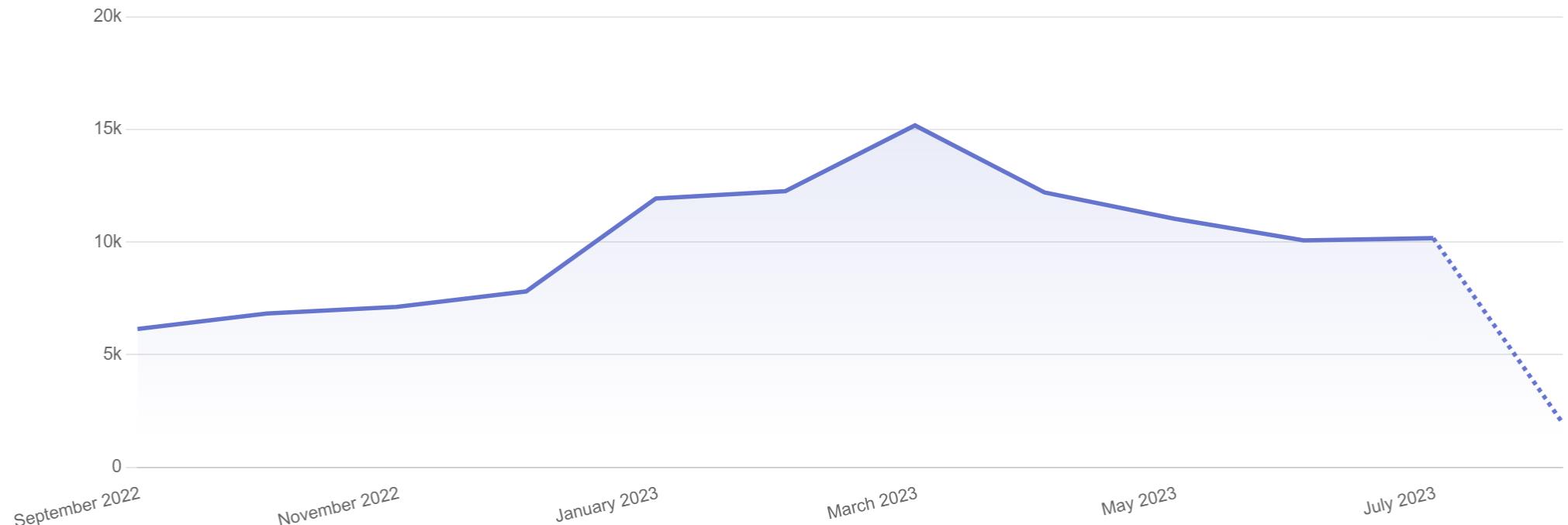
↑ 100%

BOUNCE RATE**58%**

↑ 4%

VISIT DURATION**14m 10s**

↓ 41%

  Months ▾

**Doprava**

187 datasetov

**Ekonomika a práca**

37 datasetov

**Infraštruktúra, výstavba  
a bývanie**

46 datasetov

**Kultúra, šport, cestovný  
ruch a WiFi**

71 datasetov

**Obyvateľstvo**

127 datasetov

**Politika a volby**

39 datasetov

**Priestorové údaje**

82 datasetov

**Rozpočet, dane a  
zmluvy**

35 datasetov

**Sociálna oblasť**

36 datasetov

**Veda a vzdelávanie**

61 datasetov

**Zákony a spravodlivosť**

60 datasetov

**Zdravie, životné  
prostredie a klíma**

78 datasetov



A photograph of a server room. Rows of server racks are filled with glowing blue and white lights from the front panels. The floor has a blue-tinted light strip. In the background, a white door is visible at the end of the aisle.

# Úvod do Numpy Spracovania Dát



# Úvod do Numpy Spracovanie Dát

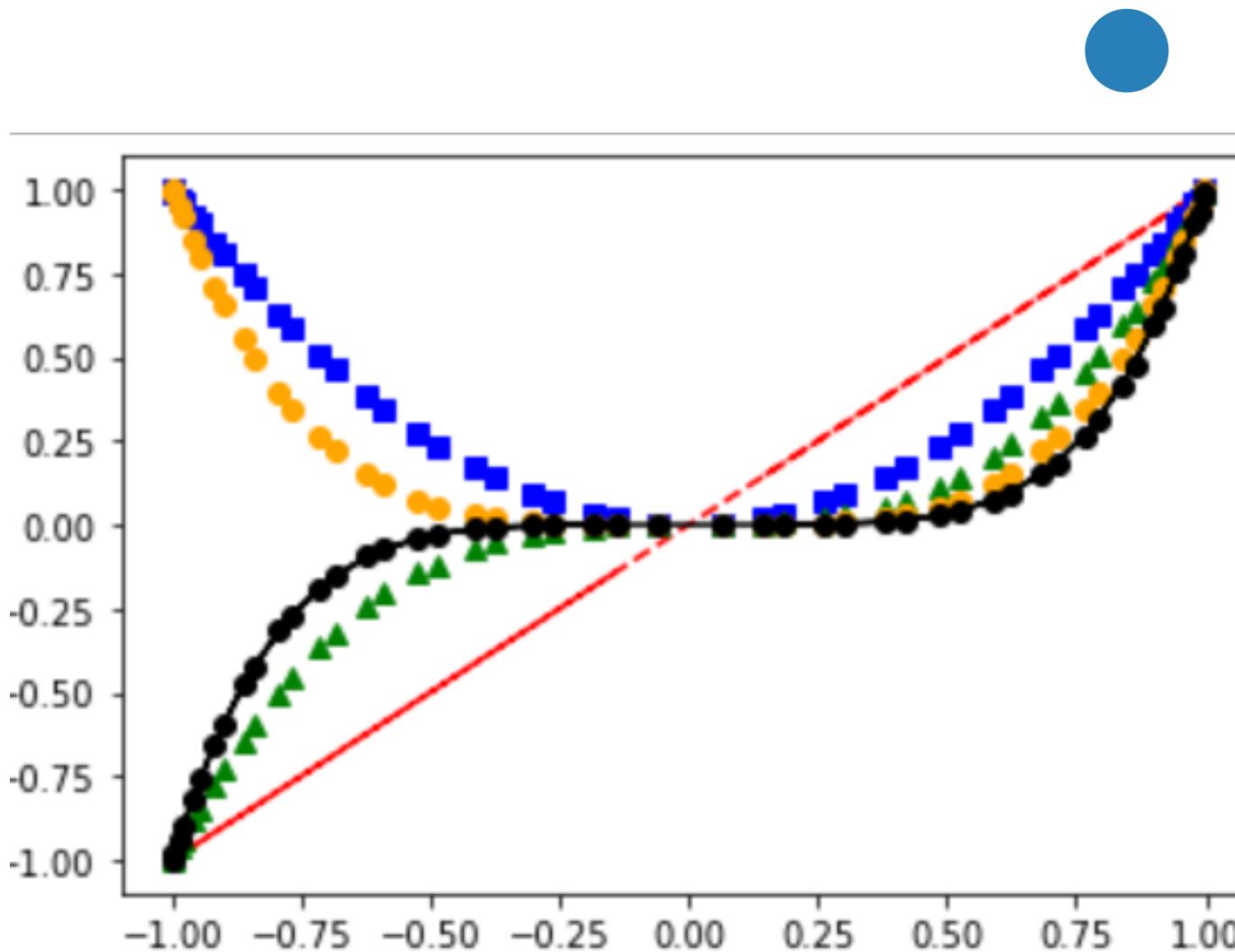
# Numpy (Numeric Python)

- Základná knižnica **pre vedecké výpočty** v Pythone
- Poskytuje **vysokovýkonný viacozmerný objekt poľa (Array)** a nástroje na prácu s týmito poľami
- **Výhody Numpy** oproti Listu
  - Menej pamäte
  - Rýchlejšie
  - Pohodlnnejšie



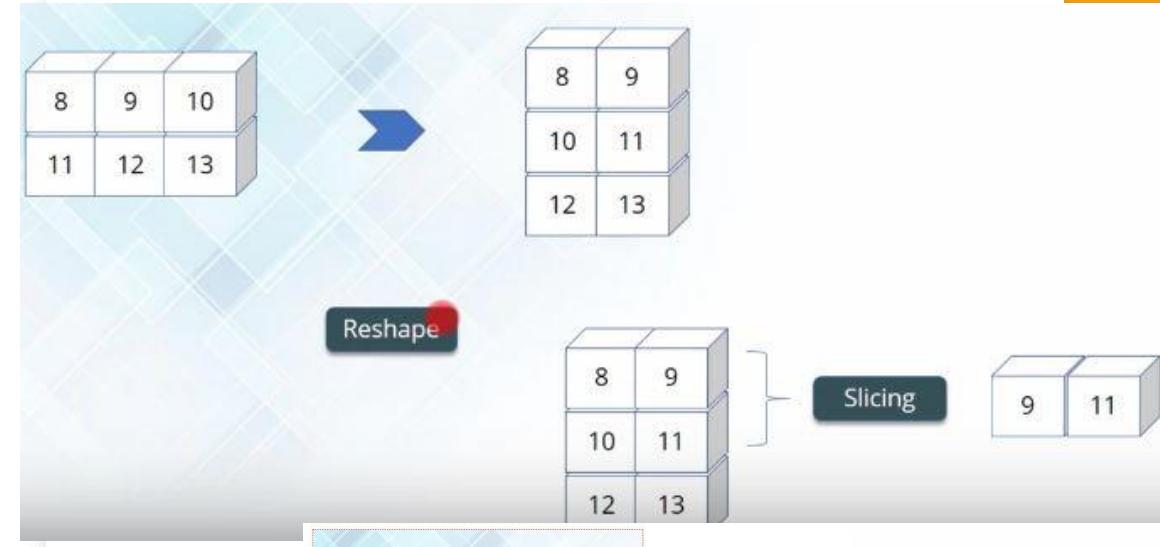
# Formátovanie štýlu vášho grafu

1. `import matplotlib.pyplot as plt`
2. `import numpy as np`
3. `t = np.sin(np.linspace(-3,3,50))`
4. `# červené čiarky, modré štvorce a zelené trojuholníky`
5. `plt.plot(t, t, 'r--', color="red")`
6. `plt.plot(t, t**2, 'bs', color="blue")`
7. `plt.plot(t, t**3, 'g^', color="green")`
8. `plt.plot(t, t**4, "o", color="orange")`
9. `plt.plot(t, t**5, "o-", color="black")`
10. `plt.show()`

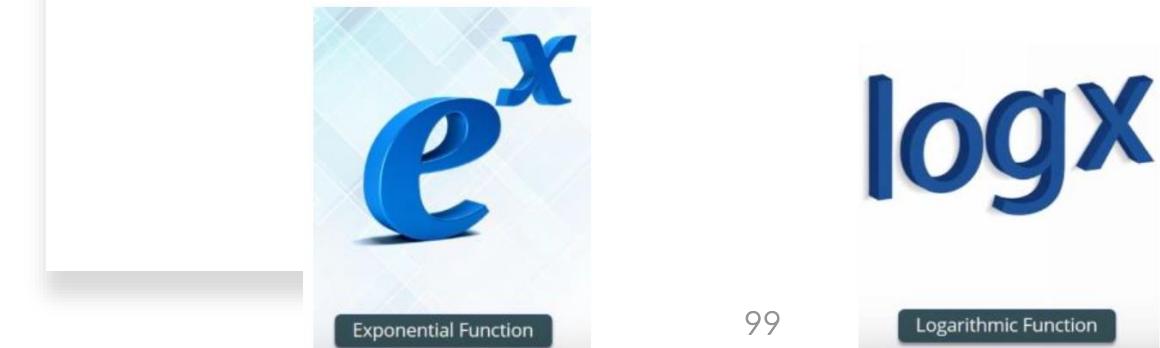


# Numpy Operácie

- Nájdenie rozmeru poľa
- Nájdenie veľkosť bajtov každého prvku
- Nájdenie typu údajov prvkov



Numpy Special Functions

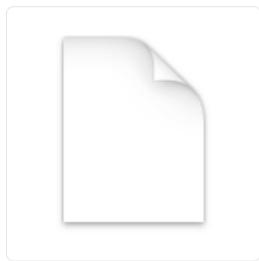


Exponential Function

# .PICKLE File Extension

Jump To: [File Information](#) [How to Open](#)

&lt;.PIC | .PICNC&gt;



## Python Pickle File

**Developer** Python**Popularity** 3.5 | 14 Votes**Category** [Developer Files](#)**Format** [Binary](#)

### What is a PICKLE file?

File created by pickle, a [Python](#) module that allows objects to be serialized to files on disk and deserialized back into the program at runtime; saves a byte stream that represents the objects; more often uses the [.P](#) extension rather than ".pickle."

### More Information

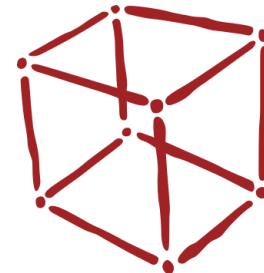
The process of serialization is called "pickling," and deserialization is called "unpickling."

# Plotly + squarify + seaborn

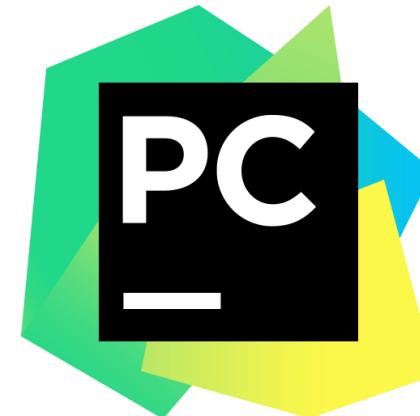
Najpopulárnejšia  
Python knižnica na  
vytváranie  
dynamických grafov



# Aké IDE mám použiť?



## NetBeans



## wxPython



## Visual Studio



# Chceme úplne všetko!

---



 ANACONDA NAVIGATOR[Sign in to Anaconda Cloud](#) Home

Applications on

base (root)

Channels

Refresh

 Environments Learning Community

## Documentation

## Developer Blog



| Applications on base (root)   |   | Channels  |
|---|---|---|
| <br>CMD.exe Prompt<br>0.1.1<br>Run a cmd.exe terminal with your current environment from Navigator activated                               | <br>JupyterLab<br>1.2.6<br>An extensible environment for interactive and reproducible computing, based on the Jupyter Notebook and Architecture.     | <br>Notebook<br>6.0.3<br>Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.                     |
| <a href="#">Launch</a>  | <a href="#">Launch</a>  | <a href="#">Launch</a>  |
| <br>VS Code<br>1.52.1<br>Streamlined code editor with support for development operations like debugging, task running and version control. | <br>Glueviz<br>0.15.2<br>Multidimensional data visualization across files. Explore relationships within and among related datasets.                  | <br>Orange 3<br>3.23.1<br>Component based data mining framework. Data visualization and data analysis for novice and expert. Interactive workflows with a large toolbox. |
| <a href="#">Launch</a>  | <a href="#">Install</a>   | <a href="#">Install</a>   |
| <br>Powershell Prompt<br>0.0.1<br>Run a Powershell terminal with your current environment from Navigator activated                       | <br>Qt Console<br>4.6.0<br>PyQt GUI that supports inline figures, proper multiline editing with syntax highlighting, graphical calltips, and more. | <br>Spyder<br>4.0.1<br>Scientific PYthon Development EnviRonment. Powerful Python IDE with advanced editing, interactive testing, debugging and introspection features   |
| <a href="#">Launch</a>  | <a href="#">Launch</a>  | <a href="#">Launch</a>  |

Microsoft Store

python

Filtre

# "python"

Všetky oddelenia Aplikácie Hry

Applikácie Hranie

Python 3.9 Python 3.8 Python 3.10 Python 3.7 Learn Django and Python by GoLearningBus WiBit.Net :: Programming in Python 计算机二级 Python 考试库

Aplikácie • Developer tools Aplikácie • Developer tools Aplikácie • Developer tools Aplikácie • Books & reference Aplikácie • Books & reference Aplikácie • Education

4.4 1 0,99 €

Nainštalované Bezplatné Bezplatné Bezplatné Bezplatné Bezplatné

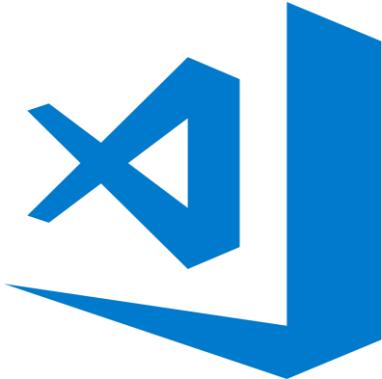
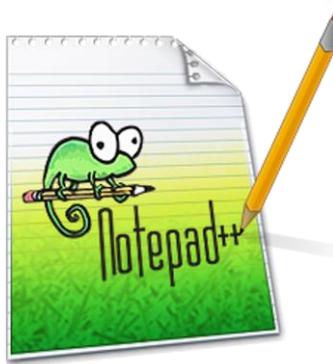
Data Science with Python Python Programs Learn Python Introduction to Python Programming by... Python Playground Python Programming Manual For Absolute... Python User Tutorial

Aplikácie • Education Aplikácie • Education Aplikácie • Books & reference Aplikácie • Books & reference Aplikácie • Developer tools Aplikácie • Utilities & tools

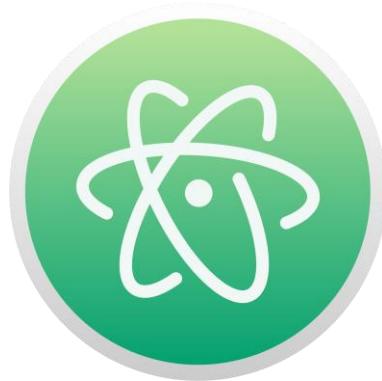
Python DATA SCIENCE Python Learn Python Python GoLearningBus Python Course Py Tutorial

Knižnica Pomocník

# Aký editor mám použiť?



```
:::  
iLE880j. :jD888880j:  
.LGitE888D.f8GjjjL8888E;  
iE :888Et. .G888.  
;i E888, ,8888,  
D888, :8888:  
D888, :8888:  
D888, :8888:  
D888, :8888:  
888W, :8888:  
W88W, :8888:  
W88W: :8888:  
DGGD: :8888:  
:8888:  
:W888:  
:8888:  
E888i  
tW88D
```



IDE ≠ editor



# Dashboard

Welcome, [itacademysk](#)**CPU Usage:** 0% used – 0.00s of 100s. Resets in 1 hour, 56 minutes [More Info](#)**File storage:** 0% full – 100.0 KB of your 512.0 MB quota [More Info](#)[Upgrade Account](#)

## Recent Consoles

[+ 5 -](#)*You have no recent consoles.*

## New console:

[\\$ Bash](#)[>>> Python ▾](#)[More...](#)

### Version

[2.7](#)[3.7](#)[3.8](#)

## Recent Files

[+ 5 -](#)

/home/itacademysk/.bashrc  
/home/itacademysk/.gitconfig  
/home/itacademysk/.profile  
/home/itacademysk/.pythonstartup.py  
/home/itacademysk/.vimrc

[+ Open another file](#)[Browse files](#)

## Recent Notebooks

[+ 5 -](#)

Your account does not support Jupyter Notebooks. [Upgrade your account](#) to get access!

## All Web apps

*You don't have any web apps.*[Open Web tab](#)

[MySQL](#)[Postgres](#)

## Initialize MySQL

Let's get started! The first thing to do is to initialize a MySQL server:

Enter a new password in the form below, and note it down: you'll need it to access the databases once you've created them. You will only need to do this once.

**New password:**

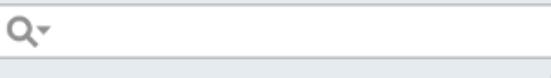
A password input field with a small circular icon containing a question mark in the top right corner.

**Confirm password:**

A password input field with a small circular icon containing a question mark in the top right corner.

**Initialize MySQL**

This should be different to your main PythonAnywhere password, because it is likely to appear in plain text in any web applications you write.



## Editor > Live Templates

By default expand with **Tab**

### Python

- compd (Dict comprehension)
- compdi (Dict comprehension with 'if')
- compg (Generator comprehension)
- compgi (Generator comprehension with 'if')
- compl (List comprehension)
- compli (List comprehension with 'if')
- comps (Set comprehension)
- compsi (Set comprehension with 'if')
- iter (Iterate (for ... in ...))
- itere (Iterate (for ... in enumerate))
- main (if \_\_name\_\_ == '\_\_main\_\_')
- prop (Property getter)
- props (Property getter/setter)
- propsd (Property getter/setter/deleter)
- super ('super(...)' call)

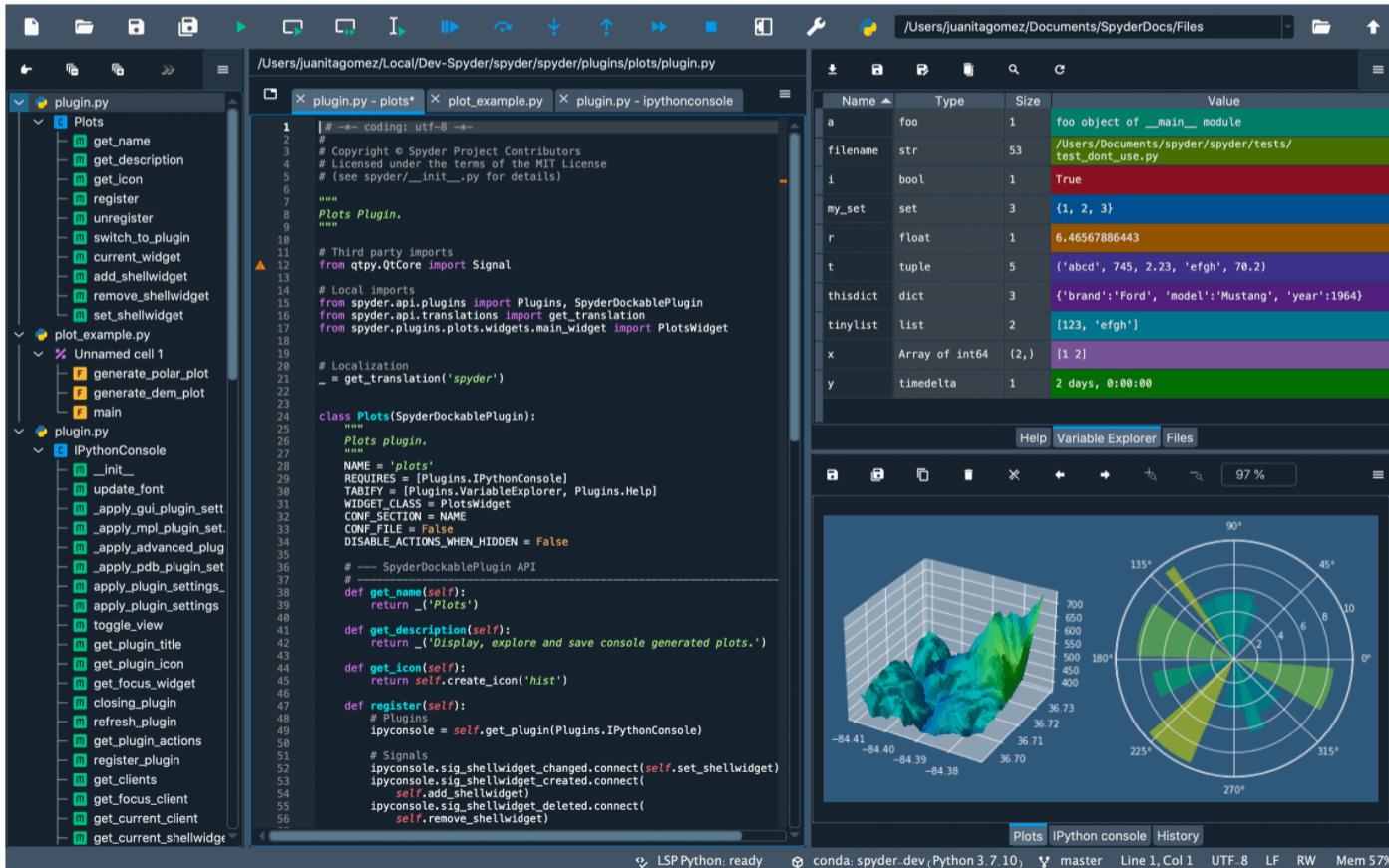
### R

### React

No live templates are selected



# The Scientific Python Development Environment



The screenshot displays the Spyder IDE interface with the following components:

- Code Editor:** Shows the file `/Users/juanitagomez/Local/Dev-Spyder/spyder/spyder/plugins/plots/plugin.py`. The code defines a `Plots` plugin for Spyder, which includes methods for generating polar plots and displaying them in the IPython console.
- Variable Explorer:** A table showing the current state of variables:

| Name     | Type           | Size | Value   |
|----------|----------------|------|---|
| a        | foo            | 1    | foo object of __main__ module                         |
| filename | str            | 53   | /Users/Documents/spyder/spyder/tests/test_dont_use.py |
| i        | bool           | 1    | True  |
| my_set   | set            | 3    | {1, 2, 3}   |
| r        | float          | 1    | 6.46567886443   |
| t        | tuple          | 5    | ('abcd', 745, 2.23, 'efgh', 78.2)                     |
| thisdict | dict           | 3    | {'brand': 'Ford', 'model': 'Mustang', 'year': 1964}   |
| tinylist | list           | 2    | [123, 'efgh']   |
| x        | Array of int64 | (2,) | [1 2]   |
| y        | timedelta      | 1    | 2 days, 0:00:00                                       |
- Plots:** Two plots are displayed: a 3D surface plot of a mountain-like function and a polar plot showing concentric rings with radial labels.
- Bottom Status Bar:** Shows the Python version (LSP Python: ready), conda environment (conda: spyder.dev), and system status (master, Line 1, Col 1, UTF-8, LF, RW, Mem 57%).

VERSION

Spyder 5

Search ...



WELCOME

QUICKSTART

INSTALLATION GUIDE

▶ INTRO VIDEOS

▶ PANES IN DEPTH

▶ SPYDER PLUGINS

▶ TROUBLESHOOTING

▶ WORKSHOPS

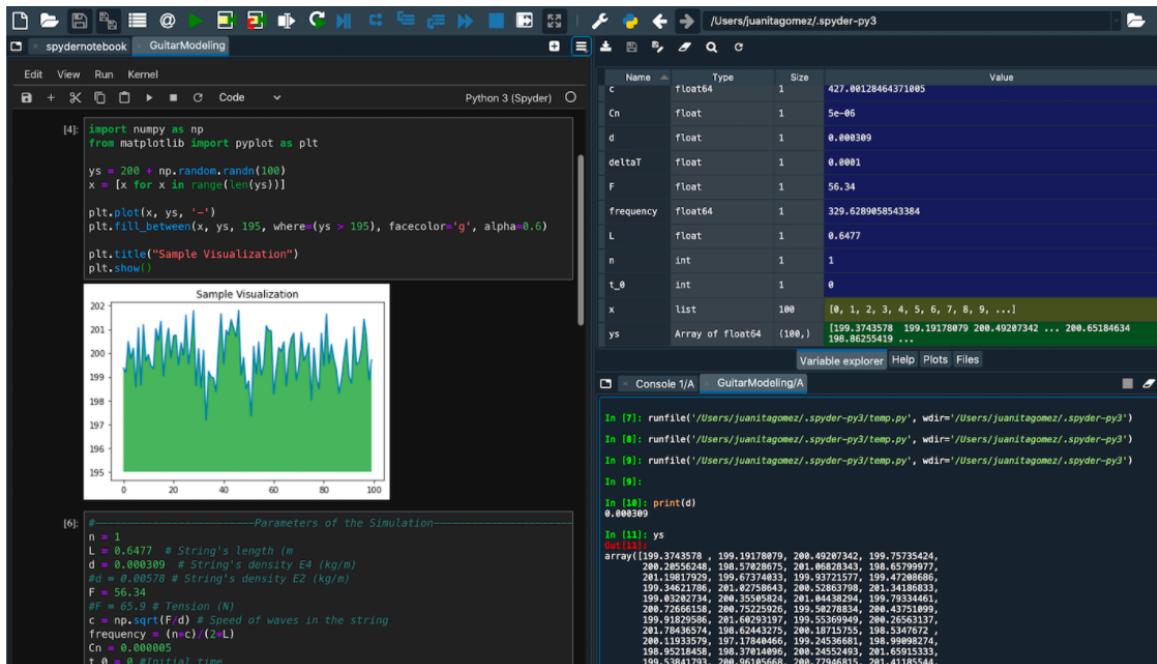
FAQ

# Spyder Notebook

## Warning

Currently, this plugin is still being ported to Spyder 5, and will likely not yet work or experience serious issues on this version of Spyder. A compatible version is expected soon. Thanks for your patience!

**Spyder-notebook** is a plugin that allows you to open, edit and interact with Jupyter Notebooks right inside Spyder.



conda install spyder-notebook -c spyder-ide

[Edit this page](#)[On this page](#)[Installing the Notebook](#)[Using the Notebook](#)[Connecting an IPython Console](#)[Additional Options](#)[OPEN CHAT](#)

## Útržky kódu

Filtrovať útržky kódu

Adding form fields →

Camera Capture →

Cross-output communication →

display.Javascript to execute Jav... →

Downloading files or importing da... →

Adding form fields

Vložit'

Forms example

Forms support multiple types of fields with type checking including sliders, date pickers, input fields, dropdown menus, and dropdown menus that allow input.

```
#@title Example form fields
#@markdown Forms support many types of fields with type checking including sliders, date pickers, input fields, dropdown menus, and dropdown menus that allow input.

no_type_checking = '' #@param
string_type = 'example' #@param
slider_value = 142 #@param {type: "number", min: 0, max: 200}
number = 102 #@param {type: "number", min: 0, max: 200}
date = '2010-11-05' #@param {type: "date", min: "2010-01-01", max: "2020-12-31"}
pick_me = "monday" #@param [ 'monday', 'tuesday', 'wednesday', 'thursday', 'friday', 'saturday', 'sunday' ]
select_or_input = "apples" #@param {type: "select", options: [ "apples", "oranges", "bananas" ]}
```

[Zobrazit zdrojový zápisník](#)

+ Kód + Text Kopírovať na Disk

```
tiene = True
# r g b, c m y k, w
farby_vlastne = ["black","pink", "b", "#CCCC00"]

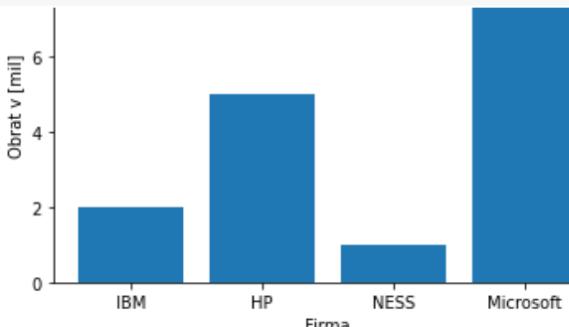
print(type(y))
print(y)

plt.pie(y, labels = menovky, startangle = 0, explode = vykrojenie, shadow = tiene, colors = farby_vlastne)
plt.legend(title = "Produkty ABC s.r.o.")
plt.title("Analyza predaja produktov za Q1-2022")
plt.show()

x1 = np.array(["IBM", "HP", "NESS", "Microsoft"])
y1 = np.array([2, 5, 1, 9])

plt.bar(x1, y1)
plt.title("Porovnanie IT firiem 2022")
plt.xlabel("Firma")
plt.ylabel("Obrat v [mil]")
plt.show()

nahoda = np.random.normal(100, 10, 200)
print(nahoda)
plt.hist(nahoda)
plt.show()
```



```
[ 79.76069196  99.4155264  114.29926387 101.33767141  88.49106384
 111.70288892  91.32702578 102.53587004 108.38846479 114.34889501
 98.79114202 117.40488367 89.26174251  94.12100639 101.96805716]
```

C:\Users\miros

▼ ↑

Python\_I\_Za...

Edit View Run Kernel

Python 3 (Spyder)

Name Type Size Value

## Kurz Python - 1. deň

Miroslav Reiter | miroslav.reiter@it-academy.sk | <https://www.linkedin.com/in/miroslav-reiter/>

Kurz Python | <https://www.it-academy.sk/kurz/python/> | <https://github.com/miroslav-reiter>

## Komentáre, kódovanie, tlač a docstringy

```
[1]: # -*- coding: utf-8 -*-
# Toto je komentár (jednoriadkový)
"""Toto je docstring (document string)"""

# Pozor na nespravne zalomovanie riadku (Enter)
# SyntaxError: EOL while scanning string Literal
# Emotikony https://emojipedia.org/
print("Python je fajnovy jazyk!")
print(__doc__)
print("🎲 🎲 🎲")
print("Co bolo skorej? --> ", min(['\N{CHICKEN}', '\N{EGG}']))
```

Python je fajnovy jazyk!
Toto je docstring (document string)
🎲 🎲 🎲
Co bolo skorej? --> 🎲

## Premenné a typy

•••

Milujem Python
Milujem Python
Milujem Python

Nazov produktu je: Hypoteka pre mladych 2021
Splatka vasej hypoteky je: 600 Eur
Uroková sadzba je: 1.5 % p.a.

Variable explorer Help Plots Files

Console 1/A

Python 3.8.5 (default, Sep 3 2020, 21:29:08) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 7.19.0 -- An enhanced Interactive Python.

In [1]:

Editor Notebook

LSP Python: ready Kite: ready (no index) conda: base (Python 3.8.5) main [86] Line 8, Col 1 UTF-8 CRLF RW Mem 47% CPU 17% 09:26



C:\Users\miros\AppData\Local\Temp\kite\_tutorial.py

temp.py kite\_tutorial.py

```

1 # Welcome to...
2 #
3 #      `hmy+.      ://:
4 #      .mMMMMMNho: ` NMMm
5 #      :NMMMMMMMMMdS/.` NMMm      :ss:
6 #      +NMMMMMMMMMMMMmy+ NMMm      -MMM-   ---
7 #      `oMMMMMMMMMMMMMMMo NMMm      /ss/   :MM+
8 #      `yMMMMMMMMNshmNMMMN` NMMm      /MM+
9 #      .dMMMMMMMMm/hmhssydmMM+ NMMm      `/yhy. shhy ohmMMMhhhh. ./ydmmdho-
10 #     omMMMMMd/mMMMMMhsosy` NMMm      .omMMmo. mMMN odmMMMdsss. omMNsoshNMNy
11 #     .+dMMMy/mMMMMMMMMMd- NMMm-yNMMy/` mMMN /MM+ sMMN: `:NMMy
12 #     `ymo/NMMMMMMMMMd NMMmNMNMMN` mMMN :MM+ MMNdddNNMMN
13 #     `hMMMMMMMMMM: NMMm+mMMNs. mMMN :MM+ MMN///////////
14 #     `:yNMMMMMMMMh NMMm `/dMMNy- mMMN :MM+ ` sMMNo` `:
15 #     .+mMMMMMd- NMMm `/dMMNy- mMMN .MMNddNN/ +NMNdhydNNMs
16 #     `:yMMMy yhs      `/hhh shhs :ymmmdho: `/sdmmmmhs:`
17 #     `om.
18
19 """
20 Kite is your Python programming copilot. Kite will try to show you the
21 right information at the right time as you code to prevent you from context
22 switching out of your current line of thought.
23
24 This tutorial will teach you how to use all of Kite's core features. You
25 should be able to learn everything in 5 minutes.
26
27 If you get stuck at any point, please visit https://help.kite.com/ or file
28 an issue at https://github.com/kiteco/issue-tracker.
29 """
30
31
32
33
34 """
35 PART 0: BEFORE WE START =====
36
37 Spyder will by default try to start the Kite backend when the editor first
38 starts. You can change this behavior by opening settings, clicking on
39 "Completion and linting", "Advanced", and then changing Kite's "Start Kite
40 Engine on editor startup" setting.
41
42 Look for the Kite indicator in the bottom left corner of Spyder's status
43 bar – It will tell you if Kite is ready and working. If the indicator reads
44 "not running", then you'll have to start the Kite Engine manually before
45 proceeding with the rest of this tutorial.
46
47
48
49

```

↓ ☰ 🔍 C

| Name | Type | Size | Value |
|------|------|------|-------|
| a    | int  | 1    | 5     |
| b    | str  | 1    | Karol |

Variable explorer Help Plots Files

Console 1/A

Python 3.8.5 (default, Sep 3 2020, 21:29:08) [MSC v.1916 64 bit (AMD64)]  
Type "copyright", "credits" or "license" for more information.

IPython 7.19.0 -- An enhanced Interactive Python.

In [1]: runfile('C:/Users/miros/.spyder-py3/temp.py', wdir='C:/Users/miros/.spyder-py3')

In [2]: runfile('C:/Users/miros/.spyder-py3/temp.py', wdir='C:/Users/miros/.spyder-py3')  
5  
Karol

In [3]:



[For Teams](#)    [Download](#)

# Code Faster. Stay in Flow.

Kite adds AI powered code completions to your code editor, giving developers superpowers.



[Download for Free](#)

```
1 import os
2 import sys
3
4 def count_py_files_in_repos(dirname):
5     if os.path.exists(os.path.join(dirname, '.git')):
6         count = 0
7         for root, dirs, files in os.walk(dirname):
8             count += len([f for f in files if f.endswith('.py')])
9         print('{} has {} Python files'.format(dirname, count))
10        for name in os.listdir(di)
```



dirname  
dirs  
dict

kite.com

```
1 import os
2 import sys
3
4 def count_py_files_in_repos(dirname):
5     i|
```

```
C:\Windows\System32\cmd.exe - pip install matlib
Microsoft Windows [Version 10.0.16299.785]
(c) 2017 Microsoft Corporation. Všetky práva vyhradené.

C:\Windows\System32>cd C:\Program Files (x86)\Python37-32\Scripts

C:\Program Files (x86)\Python37-32\Scripts>pip instal xlwt
ERROR: unknown command "instal" - maybe you meant "install"

C:\Program Files (x86)\Python37-32\Scripts>pip install xlwt
Collecting xlwt
  Downloading https://files.pythonhosted.org/packages/44/48/def306413b25c3d01753603b1a222a011b8621aed27cd7f89cbc27e6b0f4/xlwt-1.3.0-py2.py3-none-any.whl (99kB)
    100% |██████████| 102kB 826kB/s
Installing collected packages: xlwt
Could not install packages due to an EnvironmentError: [WinError 5] Access is denied: 'c:\\\\program files (x86)\\\\python37-32\\\\Lib\\\\site-packages\\\\xlwt'
Consider using the `--user` option or check the permissions.

You are using pip version 10.0.1, however version 18.1 is available.
You should consider upgrading via the 'python -m pip install --upgrade pip' command.

C:\Program Files (x86)\Python37-32\Scripts>pip install matlib
Collecting matlib
```

PIP a easy install



Seems like I've  
installed wrong version  
of Python...

| sys Variables  |                        | String Methods         |                         | Datetime Methods                               |  |
|--|------------------------|------------------------|-------------------------|--|--|
| argv   | Command line args      | capitalize() *         | lstrip()                | today()  | fromordinal(ordinal)   |
| builtin_module_names   | Linked C modules       | center(width)          | partition(sep)          | now(timezoneinfo)                              | combine(date, time)  |
| byteorder  | Native byte order      | count(sub, start, end) | replace(old, new)       | utcnow()                                       | strftime(date, format)   |
| check_interval   | Signal check frequency | decode()               | rfind(sub, start ,end)  | fromtimestamp(timestamp)                       | utcfromtimestamp(timestamp)  |
| exec_prefix  | Root directory         | encode()               | rindex(sub, start, end) |  |  |
| executable   | Name of executable     | endswith(sub)          | rjust(width)            |  |  |
| exitfunc   | Exit function name     | expandtabs()           | rpartition(sep)         |  |  |
| modules  | Loaded modules         | find(sub, start, end)  | rsplit(sep)             |  |  |
| path   | Search path            | index(sub, start, end) | rstrip()                | replace()                                      | utcoffset()  |
| platform   | Current platform       | isalnum() *            | split(sep)              | isoformat()                                    | dst()  |
| stdin, stdout, stderr  | File objects for I/O   | isalpha() *            | splittlines()           | __str__()                                      | tzname()   |
| version_info   | Python version info    | isdigit() *            | startswith(sub)         | strftime(format)                               |  |
| winver   | Version number         | islower() *            | strip()                 |  |  |
| <b>sys.argv for \$ python foo.py bar -c qux --h</b>                      |                        |                        |                         |  |  |
| sys.argv[0]  | foo.py                 | isspace() *            | swapcase() *            | <b>Date Formatting (strftime and strptime)</b> |  |
| sys.argv[1]  | bar                    | istitle() *            | title() *               | %a Abbreviated weekday (Sun)                   |  |
| sys.argv[2]  | -c                     | isupper() *            | translate(table)        | %A Weekday (Sunday)                            |  |
| sys.argv[3]  | qux                    | join()                 | upper() *               | %b Abbreviated month name (Jan)                |  |
| sys.argv[4]  | --h                    | ljust(width)           | zfill(width)            | %B Month name (January)                        |  |
|  |                        | lower()                |                         | %c Date and time                               |  |
|  |                        |                        |                         | %d Day (leading zeros) (01 to 31)              |  |
|  |                        |                        |                         | %H 24 hour (leading zeros) (00 to 23)          |  |
|  |                        |                        |                         | %I 12 hour (leading zeros) (01 to 12)          |  |
|  |                        |                        |                         | %j Day of year (001 to 366)                    |  |
|  |                        |                        |                         | %m Month (01 to 12)                            |  |
|  |                        |                        |                         | %M Minute (00 to 59)                           |  |
|  |                        |                        |                         | %p AM or PM                                    |  |
|  |                        |                        |                         | %S Second (00 to 61*)                          |  |
|  |                        |                        |                         | %U Week number <sup>1</sup> (00 to 53)         |  |
|  |                        |                        |                         | %W Week number <sup>2</sup> (0 to 6)           |  |
|  |                        |                        |                         | %x Date  |  |
|  |                        |                        |                         | %X Time  |  |
|  |                        |                        |                         | %y Year without century (00 to 99)             |  |
|  |                        |                        |                         | %Y Year (2008)                                 |  |
|  |                        |                        |                         | %Z Time zone (GMT)                             |  |
|  |                        |                        |                         | %% A literal "%" character (%)                 |  |
| os Variables   |                        | List Methods           |                         |  |  |
| altsep   | Alternative sep        | append(item)           | pop(position)           |  |  |
| curdir   | Current dir string     | count(item)            | remove(item)            |  |  |
| defpath  | Default search path    | extend(list)           | reverse()               |  |  |
| devnull  | Path of null device    | index(item)            | sort()                  |  |  |
| extsep   | Extension separator    | insert(position, item) |                         |  |  |
| linesep  | Line separator         |                        |                         |  |  |
| name   | Name of OS             |                        |                         | <b>File Methods</b>                            |  |
| pardir   | Parent dir string      |                        |                         |  |  |
| pathsep  | Patch separator        |                        |                         |  |  |
| sep  | Path separator         |                        |                         |  |  |
| <b>Note</b>  |                        | close()                | readlines(size)         |  |  |
| Registered OS names: "posix", "nt", "mac", "os2", "ce", "java", "riscos" |                        | flush()                | seek(offset)            |  |  |
|  |                        | fileno()               | tell()                  |  |  |
|  |                        | isatty()               | truncate(size)          |  |  |
|  |                        | next()                 | write(string)           |  |  |
|  |                        | read(size)             | writelines(list)        |  |  |
|  |                        | readline(size)         |                         |  |  |
| <b>Class Special Methods</b>   |                        |                        |                         |  |  |
| __new__(cls)   | __lt__(self, other)    |                        |                         | 1.   | Sunday as start of week. All days in a new year preceding the first Sunday are considered to be in week 0. |
| __init__(self, args)   | __le__(self, other)    |                        |                         | 2.   | 0 is Sunday, 6 is Saturday.  |
| __del__(self)  | __gt__(self, other)    |                        |                         | 3.   | Monday as start of week. All days in a new year preceding the first Monday are considered to be in week 0. |
| __repr__(self)   | __ge__(self, other)    |                        |                         | 4.   | This is not a mistake. Range takes account of leap and double-leap seconds.                                |
| __str__(self)  | __eq__(self, other)    |                        |                         |  |  |
| __cmp__(self, other)   | __ne__(self, other)    |                        |                         |  |  |
| __index__(self)  | __nonzero__(self)      |                        |                         |  |  |
| __hash__(self)   |                        |                        |                         |  |  |
| __getattr__(self, name)  |                        |                        |                         |  |  |
| __getattribute__(self, name)   |                        |                        |                         |  |  |
| __setattr__(self, name, attr)  |                        |                        |                         |  |  |
| __delattr__(self, name)  |                        |                        |                         |  |  |
| __call__(self, args, kwargs)   |                        |                        |                         |  |  |
| <b>Indexes and Slices (of a=[0,1,2,3,4,5])</b>                           |                        |                        |                         |  |  |
|  |                        | len(a)                 | 6                       |  |  |
|  |                        | a[0]                   | 0                       |  |  |
|  |                        | a[5]                   | 5                       |  |  |
|  |                        | a[-1]                  | 5                       |  |  |
|  |                        | a[-2]                  | 4                       |  |  |
|  |                        | a[1:]                  | [1,2,3,4,5]             |  |  |
|  |                        | a[:5]                  | [0,1,2,3,4]             |  |  |
|  |                        | a[:-2]                 | [0,1,2,3]               |  |  |
|  |                        | a[1:3]                 | [1,2]                   |  |  |
|  |                        | a[1:-1]                | [1,2,3,4]               |  |  |
|  |                        | b=a[:]                 | Shallow copy of a       |  |  |

PC Settings X

Search

> Appearance & Behavior

Keymap

Editor

- > General
- Font
- > Color Scheme
- > Code Style
- Inspections
- File and Code Templates
- File Encodings
- Live Templates**
- File Types
- > Emmet
- Images
- Intentions
- Language Injections
- Spelling
- TextMate Bundles
- TODO

Plugins

> Version Control

> Project: test1

> Build, Execution, Deployment

> Languages & Frameworks

> Tools

Editor > Live Templates

By default expand with **Tab** ▼

>  **Python**

- compd (Dict comprehension)
- compdi (Dict comprehension with 'if')
- compg (Generator comprehension)
- compgi (Generator comprehension with 'if')
- compl (List comprehension)
- compli (List comprehension with 'if')
- comps (Set comprehension)
- compsi (Set comprehension with 'if')
- iter (Iterate (for ... in ...))
- itere (Iterate (for ... in enumerate))
- main (if \_\_name\_\_ == '\_\_main\_\_')
- prop (Property getter)
- props (Property getter/setter)
- propsd (Property getter/setter/deleter)
- super ('super(...)' call)

>  **R**

>  **React**

No live templates are selected

OK Cancel Apply

|                | Cross Platform | Commercial/Fre e | Auto Code Completion | Multi-Langua ge Support | Integrat ed Python Debugging | Error Markup | Source Control Integrati on | Smart Indent | Bracket Matchin g | Line Numbering | UML Editing / Viewing | Code Folding | Code Templat es | Unit Testing | GUI Designe r (Qt, Eric, etc) | Integrat ed DB Support | Rapid Application | Development                |
|----------------|----------------|------------------|----------------------|-------------------------|------------------------------|--------------|-----------------------------|--------------|-------------------|----------------|-----------------------|--------------|-----------------|--------------|-------------------------------|------------------------|-------------------|----------------------------|
| Atom           | Y              | F                |                      |                         | Y                            | Y            | Y                           | Y            | Y                 | Y              |                       | Y            | Y               |              |                               |                        |                   |                            |
| BlackAdder     | Y              | C                |                      |                         |                              |              |                             |              | Y                 |                |                       | Y            |                 |              |                               |                        |                   |                            |
| BlueFish       | L              |                  |                      |                         |                              |              |                             |              |                   |                |                       |              |                 |              |                               |                        |                   |                            |
| ConTEXT        | W              | C                |                      |                         |                              |              |                             |              |                   |                |                       |              |                 |              |                               |                        |                   |                            |
| DABO           | Y              |                  |                      |                         |                              |              |                             |              |                   |                |                       |              |                 |              |                               |                        |                   |                            |
| DreamPie       |                | F                | Y                    |                         |                              |              | Y                           |              |                   |                |                       |              |                 |              |                               |                        |                   |                            |
| Dr.Python      |                | F                |                      |                         | Y                            |              |                             |              |                   |                |                       |              |                 |              |                               |                        |                   |                            |
| Editra         | Y              | F                | Y                    | Y                       |                              |              | Y                           | Y            | Y                 | Y              |                       | Y            |                 |              |                               |                        |                   |                            |
| Emacs          | Y              | F                | Y                    | Y                       | Y                            | Y            | Y                           | Y            | Y                 | Y              | Y                     | Y            | Y               | Y            |                               |                        |                   |                            |
| Eric Ide       | Y              | F                | Y                    |                         | Y                            | Y            |                             | Y            |                   | Y              |                       | Y            |                 |              | Y                             |                        |                   |                            |
| E-Texteditor   | W              |                  |                      |                         |                              |              |                             |              |                   |                |                       |              |                 |              |                               |                        |                   |                            |
| Geany          | Y              | F                | Y*                   | Y                       |                              |              |                             | Y            | Y                 | Y              |                       | Y            |                 |              |                               |                        |                   | *very limited              |
| Gedit          | Y              | F                | Y <sup>1</sup>       | Y                       |                              |              |                             | Y            | Y                 | Y              |                       |              | Y <sup>2</sup>  |              |                               |                        |                   | 1with plugin; 2sort of     |
| Idle           | Y              | F                | Y                    |                         | Y                            |              |                             | Y            | Y                 |                |                       |              |                 |              |                               |                        |                   |                            |
| JEdit          | Y              | F                |                      | Y                       |                              |              |                             |              | Y                 | Y              |                       | Y            |                 |              |                               |                        |                   |                            |
| KDevelop       | Y              | F                |                      | Y                       |                              |              | Y                           | Y            | Y                 | Y              |                       | Y            |                 |              |                               |                        |                   |                            |
| Komodo         | Y              | CF               | Y                    | Y                       | Y                            | Y            | Y                           | Y            | Y                 | Y              |                       | Y            | Y               | Y            | Y                             | Y                      |                   |                            |
| NetBeans*      | Y              | F                | Y                    | Y                       | Y                            |              | Y                           | Y            | Y                 | Y              | Y                     | Y            | Y               | Y            | Y                             | Y                      | Y                 | *pre-v7.0                  |
| NotePad++      | W              | F                |                      | Y                       |                              |              |                             | Y*           |                   | Y              |                       |              |                 |              |                               |                        |                   | *with plugin               |
| Pfaide         | W              | C                | Y                    | Y                       |                              |              |                             | Y            | Y                 | Y              |                       | Y            | Y               |              |                               |                        |                   |                            |
| PIDA           | LW             | F                | Y                    | Y                       |                              |              |                             | Y            | Y                 | Y              |                       | Y            |                 |              |                               |                        |                   | VIM based                  |
| PTVS           | W              | F                | Y                    | Y                       | Y                            | Y            | Y                           | Y            | Y                 | Y              |                       | Y            |                 |              | Y*                            |                        | Y                 | *WPF bsed                  |
| PyCharm        | Y              | CF               | Y                    | Y*                      | Y                            |              | Y                           | Y            | Y                 | Y              |                       | Y            |                 | Y            |                               |                        |                   | *JavaScript                |
| PyDev(Eclipse) | Y              | F                | Y                    | Y                       | Y                            | Y            | Y                           | Y            | Y                 | Y              | Y                     | Y            | Y               | Y            | Y                             | Y                      |                   |                            |
| Pyscripter     | W              | F                | Y                    |                         | Y                            | Y            |                             | Y            |                   | Y              |                       |              | Y               | Y            |                               |                        |                   |                            |
| PythonWin      | W              | F                | Y                    |                         | Y                            |              |                             | Y            | Y                 | Y              |                       | Y            |                 |              |                               |                        |                   |                            |
| SciTE          | Y              | F <sup>1</sup>   |                      | Y                       |                              | Y            |                             | Y            | Y                 | Y              |                       | Y            | Y               |              |                               |                        |                   | 1Mac version is commercial |
| ScriptDev      | W              | C                | Y                    | Y                       | Y                            | Y            | Y                           |              | Y                 | Y              | Y                     |              | Y               | Y            |                               |                        |                   |                            |
| SPE            |                | F                | Y                    |                         |                              |              |                             |              |                   |                | Y                     |              |                 |              |                               |                        |                   |                            |
| Spyder         | Y              | F                | Y                    |                         | Y                            | Y            |                             | Y            | Y                 | Y              |                       |              |                 |              |                               |                        |                   |                            |
| Sublime Text   | Y              | CF               | Y                    | Y                       |                              |              |                             | Y            | Y                 | Y              |                       | Y            | Y               | Y*           |                               |                        |                   | extensible w/Python,       |

# Ktoré GUI a prečo?



*wxPython*



# Čo sa oplatí prečítať?

## Slovensko a česko

- Albatrosmedia
- Kopp
- Grada
- Wolters Kluwer
- BEN
- Veda

## Zahraničie

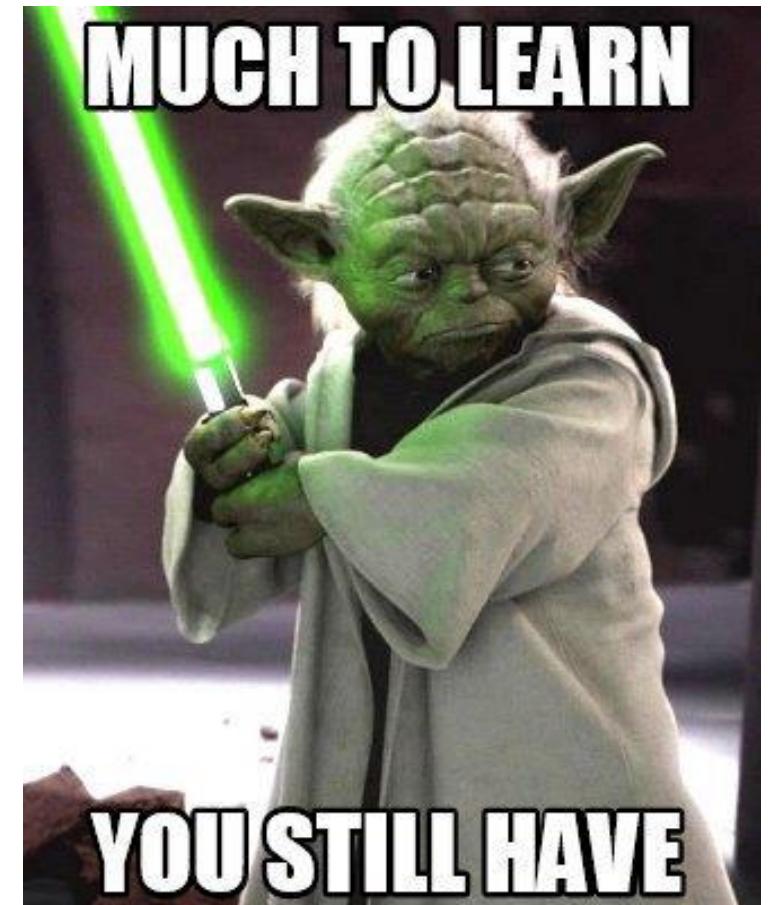
- O'Reilly
- Manning
- Packt
- Apress
- Wiley
- No Starch Press

## YouTube tutoriály

Packt Publishing

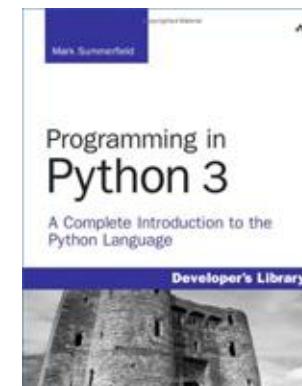
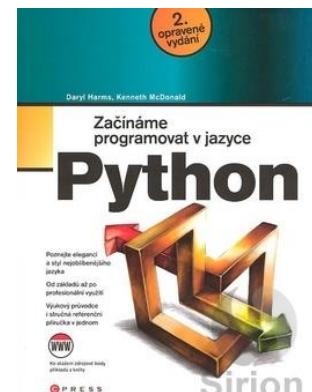
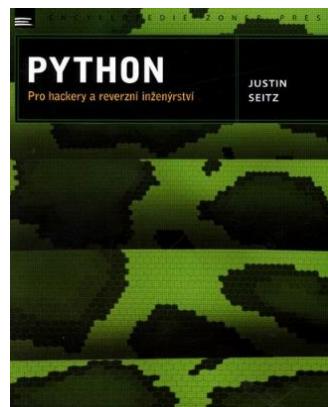
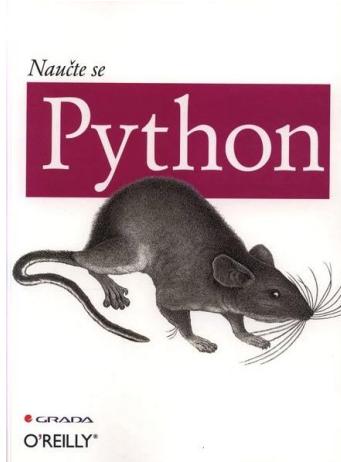
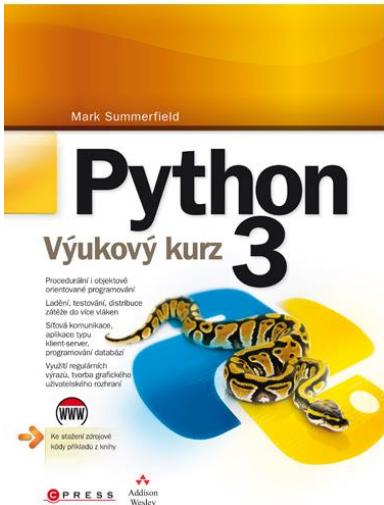
# Čo ti odporúčam si pozrieť?

1. <https://docs.python.org/3/>
2. <https://realpython.com/tutorials/best-practices/>
3. <https://google.github.io/styleguide/pyguide.html>
4. <https://docs.python.org/3/>
5. <http://python2013.input.sk/19prednaska>
6. <https://realpython.com/python3-object-oriented-programming/>
7. <https://jeffknupp.com/blog/2014/06/18/improve-your-python-python-classes-and-object-oriented-programming/>
8. <https://overiq.com/python-101/inheritance-and-polymorphism-in-python/>
9. <https://www.javatpoint.com/python-oops-concepts>
10. <https://www.programiz.com/python-programming/object-oriented-programming>



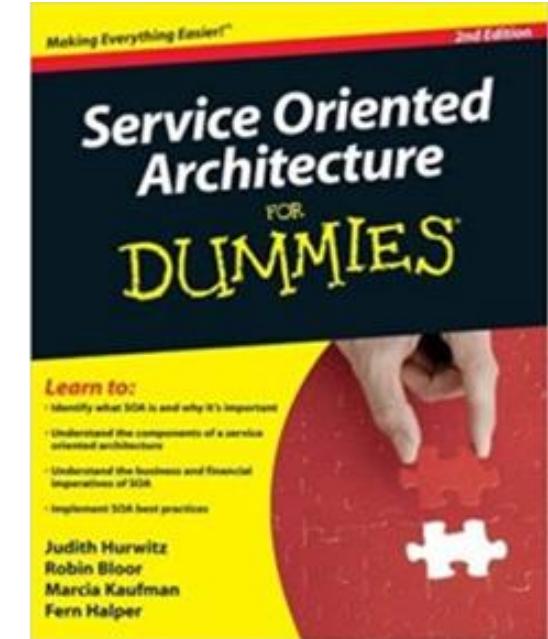
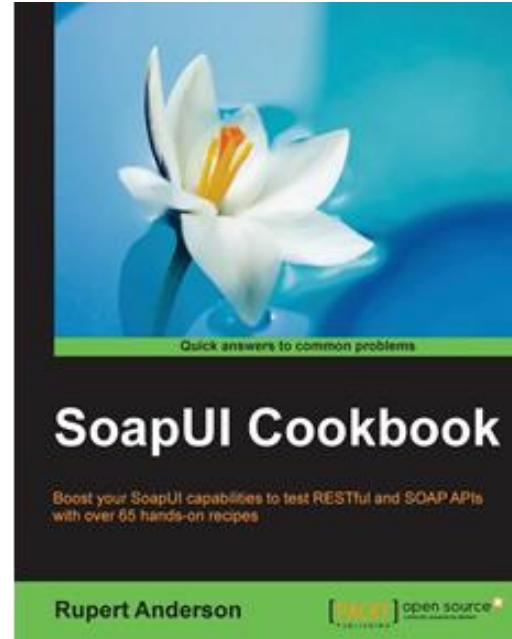
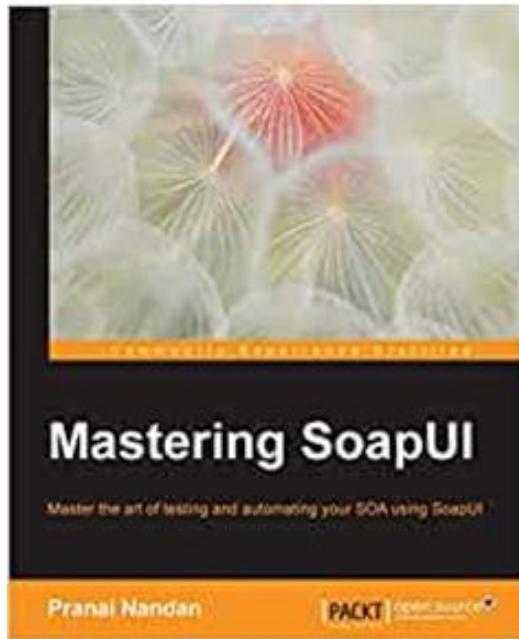
Šup do záložiek

# Čo sa oplatí/neoplatí prečítať SK/CZ?



Mark Pilgrim

# Čo sa oplatí/neoplatí prečítať EN?



# I am programmer



I have Life



I have  
stackoverflow



IT ACADEMY

Home

PUBLIC

Questions

**Tags**

Users

COLLECTIVES

Explore Collectives

FIND A JOB

Jobs

Companies

TEAMS

Create free Team

# Tags

A tag is a keyword or label that categorizes your question with other, similar questions. Using the right tags makes it easier for others to find and answer your question.

[Show all tag synonyms](#)

python

**python**

Python is a multi-paradigm, dynamically typed, multi-purpose programming language. It is designed to be quick to learn, understand, and...

1870168 questions 695 asked today, 6387 this week

**python-3.x**

USE ONLY IF YOUR QUESTION IS VERSION-SPECIFIC. For questions about Python programming that are specific to version 3+ of the language...

303562 questions 93 asked today, 836 this week

**python-2.7**

Python 2.7 is the last major version in the 2.x series, and is no longer maintained since January 1st 2020. Use the generic [python] tag on all Python...

94965 questions 24 asked this week, 106 this month

**python-requests**

USE ONLY FOR THE PYTHON REQUESTS LIBRARY. Requests is a full-featured Python HTTP library with an easy-to-use, logical API.

18697 questions 8 asked today, 57 this week

**python-imaging-library**

The Python Imaging Library (PIL) provides the Python language with a de-facto standard foundation for image work. PIL's API is lightweight but...

7883 questions 5 asked today, 38 this week

**wxpython**

wxPython is a Python wrapper for the cross-platform C++ GUI API wxWidgets.

7047 questions 7 asked this week, 14 this month

**ipython**

IPython is a feature-rich interactive shell for Python, and provides a kernel for frontends such as IPython Notebook and Jupyter Notebook.

6886 questions 5 asked this week, 26 this month

**python-3.6**

Version of the Python programming language released in December 2016. For issues specific to Python 3.6. Use more generic [python] and [python-3....

5602 questions 11 asked this week, 24 this month

**python-asyncio**

to be used for the asyncio Python package which provides mechanisms for writing single-threaded concurrent code. The asyncio package provides...

5492 questions 29 asked this week, 125 this month

**python-import**

For questions about importing modules in Python

5119 questions 11 asked this week, 47 this month

**python-multiprocessing**

multiprocessing is a package that supports spawning processes using an API similar to the threading module in python programming language.

4036 questions 12 asked this week, 46 this month

**python-3.7**

Version of the Python programming language released in June 27, 2018. For issues that are specific to Python 3.7. Use the more generic [python] and...

4034 questions 5 asked this week, 21 this month

[Popular](#) [Name](#) [New](#)

# Efektívne používanie klávesnice

**Špeciálne znaky, kde ich nájst' na klávesnici**

The diagram shows a standard QWERTY keyboard with additional labels and icons:

- Top Row:** Special characters (e.g., 1, 2, 3, 4, 5, 6, 7, 8, 9, 0, %, ě, ď, ľ, š, č, ľ, ž, ý, á, í, é, ó), BACK SPACE, and ENTER.
- Second Row:** TAB, Q, W, E, R, T, Y, U, I, O, P, /, (, Ú, ä, and another ENTER key.
- Third Row:** CAPS LOCK, A, ~, S, D, F, [, G, ], H, J, K, L, " \$ ! ' ) , ô, §, ' ñ.
- Fourth Row:** Up arrow, |, Y, X, #, C, &, V, @, B, {, N, }, M, ?, <, :, >, -, \*, and another SHIFT key.
- Bottom Row:** Special characters (~, !, @, #, \$, %, ^, &, \*, (, ), -, +, =, |, \, and back arrow), TAB, Q, W, E, R, T, Y, U, I, O, P, {, }, [, ], and another ENTER key.
- Bottom-Shift Row:** Z, X, C, V, B, N, M, <, >, ?, /, and another SHIFT key.

**Legend:**

| Kategória       | Znaky            | Popis  |
|-----------------|------------------|--|
| Operátory       | +, *, -, /, %, @ | Sčítavanie, Spájanie; Násobenie, Opakovanie; Odčítanie; Delenie; Zvyšok, modulo; Dekorátor, Nás. matic |
| Porovnávanie    | <, >, =, !       | Väčšie, Menšie; Rovnosť, Priradenie; Nerovnosť   |
| Retázce         | '", \            | Úvodzovky; Špeciálne znaky   |
| Oddelovače      | , . ; : ?        | Prvok; Atribútov; Blokov, Klúčov; Príkazov; Poznámky; Komentár   |
| Bitové operácie | &,  , ^, ~       | Prienik, A, AND; Zjednotenie, OR; XOR; Inverzie  |
| Zátvorky        | (,), {}, [], \$  | Zátvorky, Volanie; Slovníky, Formát; Zoznamy, Indexy; Ostatné; Súčasť mena; Nevyužité                  |

# Najdôležitejšie klávesové skratky

## Práca s IDE

- Ctrl + D Delete zmaž riadok
- **Ctrl + Space** Asistent kódu
- **Ctrl + /** Komentáre
- Ctrl + A Označ všetko
- **Alt + /** Dokonči slovo
- Ctrl + F Hľadanie a náhrady
- Ctrl + Shift + F Kompakt režim
- Ctrl + Shift + S Ulož všetko

## Práca s browserom

- Ctrl + T Vytvor nový tab
- Ctrl + W Zatvor aktuálny tab
- Ctrl + Shift + W Zatvor všetky taby
- **Ctrl + Shift + T** Otvor posledný tab
- Ctrl + Shift + J/F12 Web console
- **F11** Fullscreen

**F5 nie je spustenie, ale Refresh**

# PYCON SK 2022

Bratislava

[KÚP SI LÍSTOK](#)



Vývojári



Miroslav

Domov

Vytvoriť



## Vývojári

Verejná skupina

Informácie

## Diskusia

Oznámenia

Členovia

Podujatia

Videá

Fotky

Súbory

Hľadať v tejto skupine



Ste člen

Upozornenia

Zdieľať

... Viac



Napísat príspěvok

Pridať fotku/video

Živé video

Viac

Napište niečo...



Napište niečo...



Fotka/video



Divácka páry



Označiť priateľov



NOVÁ AKTIVITA



Roland Mondek

10 h

## POZVAŤ ČLENOV

+ Zadajte meno alebo e-mailovú adresu...



## ČLENOVIA

5 505 členov



## POPIS

Skupina softvérových vývojárov. Táto skupina by mala byť miestom... Zobraziť viac

## TYP SKUPINY

Všeobecné

## VAŠE STRÁNKY



IT Academy



VITA - Virtual It Academy

## KONTAKTY



Evka Rybárska



Jarmila Palenčárová



Stefan Orosi



Ivana Ivka Jasaňová



Hrá Word Blitz



Ivana Pavlíková



Martin Vanko



Lucia Kovačičová

4 h



Lošák Filip



Andrej Nejedlik



Gabika Zubrikova

## SKUPINOVÉ KONVERZÁCIE



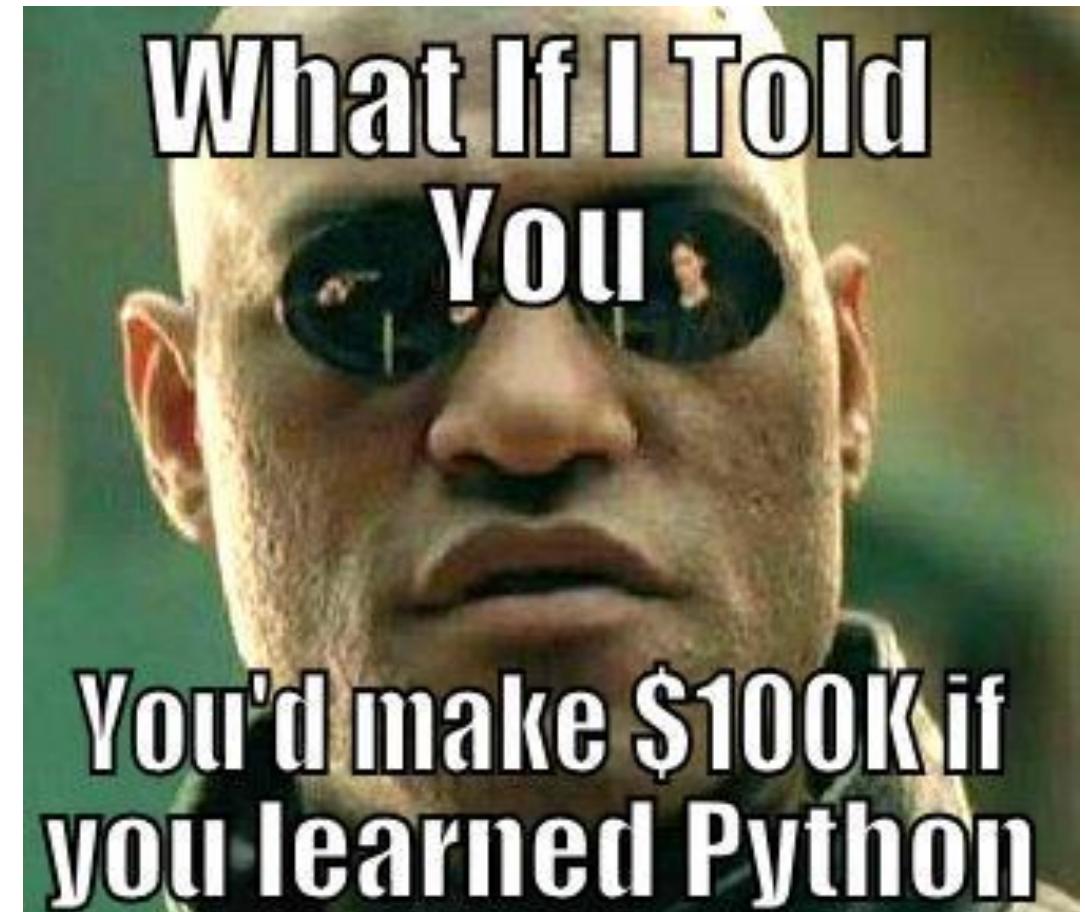
Vytvoriť novú skupinu

Hľadať kontaktov / 821



# Zen filozofia Pythonu

1. Krásny je lepší než škaredý
2. Explicitný je lepší ako implicitný
3. Jednoduchý je lepší ako zložitý
4. Zložitý je lepší ako komplikovaný
5. Plochý je lepší ako vnorený
6. Riedky je lepší ako hustý
7. Na čitateľnosti záleží
8. Praktickosť vyhráva nad čistotou



import this

# Čaká nás krásna budúcnosť

```
>>> from __future__ import braces  
SyntaxError: not a chance (<pyshell#13>, line 2)  
>>> |
```

No future {} a ;



# Inšpirácia na projekty

# Python Project Ideas

Easy



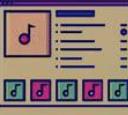
Quote Gener. Number guessing Dice Simulation YT downloader

Mid



Password Manag. Mario Party Web Crawler Email Autom.

Pro



Music Player Face Detection Twitter Clone Twitter Bot

# Čo je Grow with Google?

Grow with Google je **globálna iniciatíva**, ktorej cieľom je vytvoriť viac príležitostí pre všetkých.

Poskytuje **kurzy, nástroje a školenia** či už online alebo offline, ktoré ľuďom pomáhajú získavať tie správne zručnosti, aby si našli prácu, posunuli svoju kariéru či nakopli svoje podnikanie.

V roku 2021 sa zameriava na pomoc:

- **Malým a stredným podnikateľom**
- **Startupom**
- **Neziskovým organizáciám**
- **Ženám**
- **Mnoho ďalším ...**

Viac informácií nájdete na:  
[www.growwithgoogle.sk](http://www.growwithgoogle.sk)



# Slovensko môže patríť k digitálne najrozvinutejším ekonomikám, avšak...

136



**Strácame  
momentum**  
kvôli nedostatku  
inovácií



Až 99 % firiem  
na Slovensku sú  
**malé a stredné  
podniky**



Máme **nízky počet  
startupov**, najmä v  
regiónoch



**Digitalizácia  
podnikania** je na  
Slovensku jedna z  
**najnižších v rámci EU**

**Ekonomika na Slovensku bola  
ovplyvnená nedávnou krízou a  
technológie môžu pomôcť pri jej  
prekonávaní aj v budúcnosti...**

## ...klúčom k naštartovaniu ekonomiky a digitálnej transformácie je vzdelávanie

Od roku 2016 pomáha Grow with Google firmám, používateľom a inštitúciám na Slovensku prijímať digitálnu transformáciu bezpečným spôsobom.

### 100,000 ľudí

sme spoločne preškolili v zručnostiach potrebných pre budúcnosť

### 450 školení

sme spolu s partnermi poskytli na celom Slovensku



### 5 000 + ľudí

začalo podnikať po absolvovaní nášho vzdelávania

Pomáhame Slovensku rásť

# Grow with Google

Našim záväzkom je prispieť k oživeniu slovenskej ekonomiky a do konca roka 2021

pomôcť ďalším 50,000 slovenským firmám a jednotlivcom

lepšie využívať internet k rastu svojho podnikania, rozvoja kariéry či nájdenia novej práce.

## Online marketing

Google Analytics

Online marketing strategy

Google for Nonprofits

Shopping

YouTube

Google My Business

Google Ads

## Technology & Tools

Workspace (G Suite)

Online Security & Safety

Google for Education

AI/ML

## Business & Soft skills

Entrepreneurship

Leadership

Export

#IamRemarkable\*

Entrepreneurship / Diversity for women

Critical thinking / Media literacy

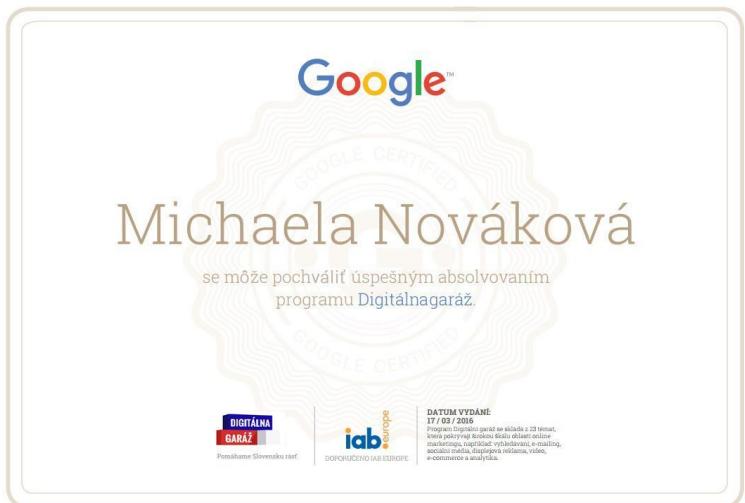
## Online platforma na výuku digitálneho marketingu a mäkkých zručností

26 tém

106 lekcií

9 hodín obsahu

Dostupné 24/7  
Zadarmo  
Na mieru



Google Digitálna garáž

1. Základy e-mailového marketingu

Téma: Využite e-mailové spojenie

Prehľad tému

Lekcie 0 / 4

1. Základy e-mailového marketingu

Sledujte lekcii 6 min

Otestujte si svoje znalosti 1 min

2. Možnosti e-mailového marketingu

3. Vytváranie skvelých marketingových e-mailov

4. Správa úspešných e-mailových kampaní

PRESKOČIŤ NA TEST

ZÁLOŽKA ZDIELAŤ

0:01 / 5:04 YouTube

ZOBRAZIŤ PREPIS

OTESTUJTE SI SVOJE ZNALOSTI

Hlavné poznatky

Zasielanie bulletinov a akčných ponúk zákazníkom prostredníctvom e-mailu môže zohrať kľúčovú úlohu vo vašom marketingovom pláne. Budujte a upevňujte vzťahy so zákazníkmi. V tomto videu sa pozrieme na:

- vytváranie zoznamu kontaktov,
- zacielenie na publikum na základe záujmov,
- budovanie vzťahov so zákazníkmi.

# Odporúčaná literatúra:

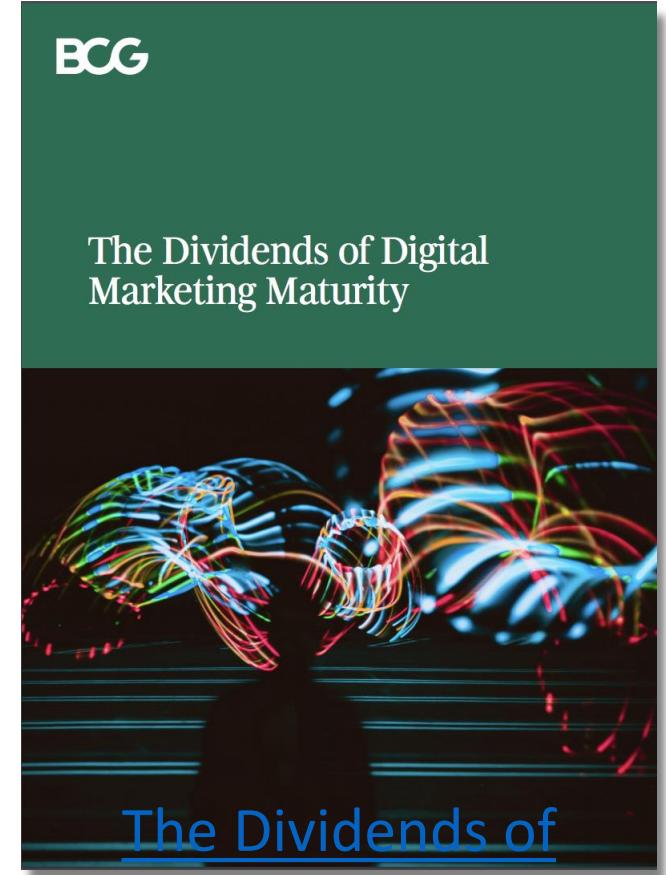
141



[The rise of Digital Challengers, 2018](#)



[Digital Challengers in the next normal, 2020](#)



[The Dividends of  
Digital Marketing  
Maturity, 2019](#)



Našim záväzkom je **prispiet' k oživeniu slovenskej ekonomiky** a do konca roka 2021

**pomôcť ďalším 50,000 slovenským firmám a jednotlivcom**

lepšie využívať internet k rastu svojho podnikania, rozvoja kariéry či nájdenia novej práce.

## Online marketing

Google Analytics

Online marketing strategy

Google for Nonprofits

Shopping

YouTube

Google My Business

Google Ads

## Technology & Tools

Workspace (G Suite)

Online Security & Safety

Google for Education

AI/ML

## Business & Soft skills

Entrepreneurship

Leadership

Export

#IamRemarkable\*

Entrepreneurship / Diversity for women

Critical thinking / Media literacy



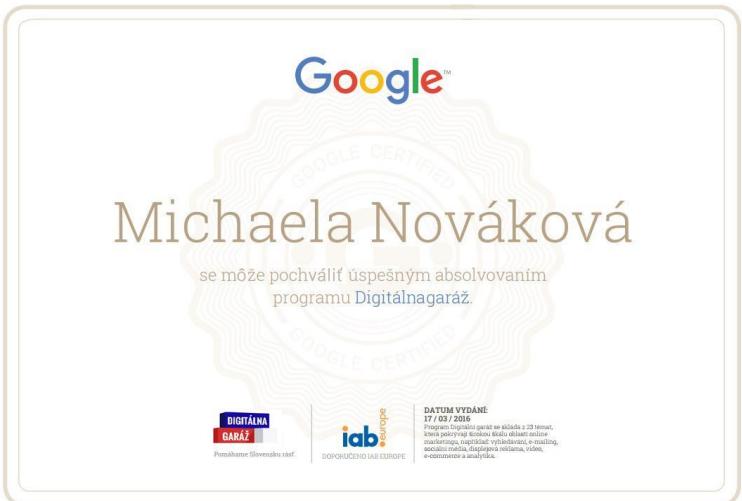
Digitálna garáž Online platforma na výuku digitálneho marketingu a mäkkých zručností

26 tém

106 lekcií

9 hodín obsahu

Dostupné 24/7  
Zadarmo  
Na mieru



≡ | Google Digitálna garáž

1. Základy e-mailového marketingu  
Téma: Využite e-mailové spojenie

☆ ZÁLOŽKA ⌂ ZDIELAŤ:

Prehľad tém

Lekcie 0 / 4

☆ 1. Základy e-mailového marketingu

Sledujte lekcii 6 min

Otestujte si svoje znalosti 1 min

☆ 2. Možnosti e-mailového marketingu

☆ 3. Vytváranie skvelých marketingových e-mailov

☆ 4. Správa úspešných e-mailových kampaní

PRESKOČIŤ NA TEST

YouTube

0:01 / 5:04

ZOBRAZIŤ PREPIS

OTESTUJTE SI SVOJE ZNALOSTI

Hlavné poznatky

Zasielanie bulletinov a akčných ponúk zákazníkom prostredníctvom e-mailu môže zohrať kľúčovú úlohu vo vašom marketingovom pláne. Budujte a upevňujte vzťahy so zákazníkmi. V tomto videu sa pozrieme na:

- vytváranie zoznamu kontaktov,
- zacielenie na publikum na základe záujmov,
- budovanie vzťahov so zákazníkmi.

# Ako skončíme?

## 1. Pridaj si ma na LinkedIn

- [www.linkedin.com/in/miroslav-reiter](http://www.linkedin.com/in/miroslav-reiter)

## 2. Materiály po prednáške

- <https://1drv.ms/p/s!AlrLrycbTQ1a19sf c1MmNNnYMaluWA?e=FTUITc>

The screenshot shows the LinkedIn SlideShare interface. At the top, there is a navigation bar with the LinkedIn logo, the word "SlideShare", a search bar, and a user profile icon. Below the navigation bar, there are tabs for "Home" and "Explore". On the left, there are buttons for "My Uploads", "My Comments", and "Analytics". A dropdown menu shows "Most Recent" and a "Select all" option. There is also a search bar for "Search Uploads...". Two presentation thumbnails are displayed:

- Automatizácia a optimalizácia Firemné procesy**  
by Ing. Mgr. Miroslav Reiter, DSc., N  
[mirek.reiter@reiter.sk](#)  
1 month ago, 11 slides
- Najväčšie faily na sociálnych sietach a Google reklame**  
by Ing. Mgr. Miroslav Reiter, DSc., N  
[mirek.reiter@reiter.sk](#)  
6 months ago, 63 slides

Below each thumbnail, there are icons for views (113 or 19), likes (0), comments (0), and downloads (0).

# Ako sa s nami spojiť?



**ADRESA:** IT Academy, s. r. o.

Budova KOLOSEO prízemie  
Tomášikova 50/A  
831 04 Bratislava



**WEB:** [www.it-academy.sk](http://www.it-academy.sk)



**E-MAIL:** [info@it-academy.sk](mailto:info@it-academy.sk)



**TELEFÓN:** +421 917 095 406



# Ako vieme pomôcť?

#Copywriting

#Školenia

#Zamestnanci

#Pomáhame

#Rast

#Projekty

#Certifikácie

#Kurzy

#Tréningy

#Vzdelávanie

#PPC Kampane

#Elearning

#Mentoring

#Konzultácie

#Online

#Programovanie

#Vývoj

#Marketing

#Reklama

#Prenájom Techniky

Mrkni na náš YouTube kanál a daj odber



[www.YOUTUBE.COM/C/IT-ACADEMYSK](https://www.youtube.com/c/IT-ACADEMYSK)



IT ACADEMY

## Spracovanie a vizualizácia dát v Pythone

Základy dátovej analýzy, spracovanie a vizualizácia dát v programovacom jazyku Python.

Prerekvizitou tohto kurzu sú základné zručnosti v programovacom jazyku Python. Pokiaľ ste absolvovali predošlý kurz s názvom „Základy programovacieho jazyka Python“ prípadne „Objektovo orientované programovanie v Pythone“, určite splňate základné prepoklady pre absolvovanie tohto kurzu.

Opäť (ako v predchádzajúcich kurzoch) budeme pracovať v prostredí Jupyter Notebook, ktorý si môžete nainštalovať aj doma na svojom osobnom počítači prostredníctom GUI Anaconda Navigator.

### 1. Základný balík NumPy:

- Nainštalovanie knižnice NumPy
- Vytvorenie NumPy polí
- Dátové typy a operácie s NumPy poľami
- Indexovanie a prechádzanie NumPy polí
- Univerzálnie NumPy funkcie
- Spracovanie a filtrovanie NumPy polí
- Zhrnutie nových znalostí

### 2. Vizualizácia dát:

- Nainštalovanie knižnice pre vizualizáciu dát
- Úvod do vizualizácie v knižničach Matplotlib a Seaborn
- Vizualizácia dát na rozličných príkladoch v spomínaných knižničach
- Prispôsobenie(Customization) výstupov grafov podľa našej potreby

### 3. Spracovanie dát:

- Nainštalovanie a import Pandas knižnice
- Vytváranie Pandas dataframov
- Načítanie súborov (.txt, .xlsx, .csv)
- Spracovanie a analýza dát zo súborov(.txt, .xlsx, .csv)
- Operácie s dátami
- Vytváranie grafov
- Zhrnutie

#### Trvanie:

10 hodín (2 dni)

#### Cena:

€0,-

#### Kategória:

Python

#### Registrácia (počet prihlásených):

18. - 19. 5. 2022 (43/50)

# Vyber si online kurz

Nauč sa programovať, tvoriť webstránky a grafiku, manažovať alebo sa zameraj na osobný rozvoj. Všetko jednoducho vďaka našim online kurzom z pohodlia tvojho domova.

Ročné  
predplatné na  
všetky online  
kurzy

~~2299.99€~~

399.99€

Priístup pre Teba do všetkých aktuálnych aj pripravovaných online kurzov

12 mesačná platnosť

 Kúpiť teraz

## Zadarmo

1. Kurzy SAV

2. Kurzy Grow with Google

3. YouTube kanál IT Academy

<https://www.youtube.com/c/IT-AcademySK>

## Platené

Moje kurzy na [www.vita.sk](http://www.vita.sk)