

# Python

## Automatizácia



# Ako Začneme?

## 1. Maj nainštalovaný PyCharm Comm.

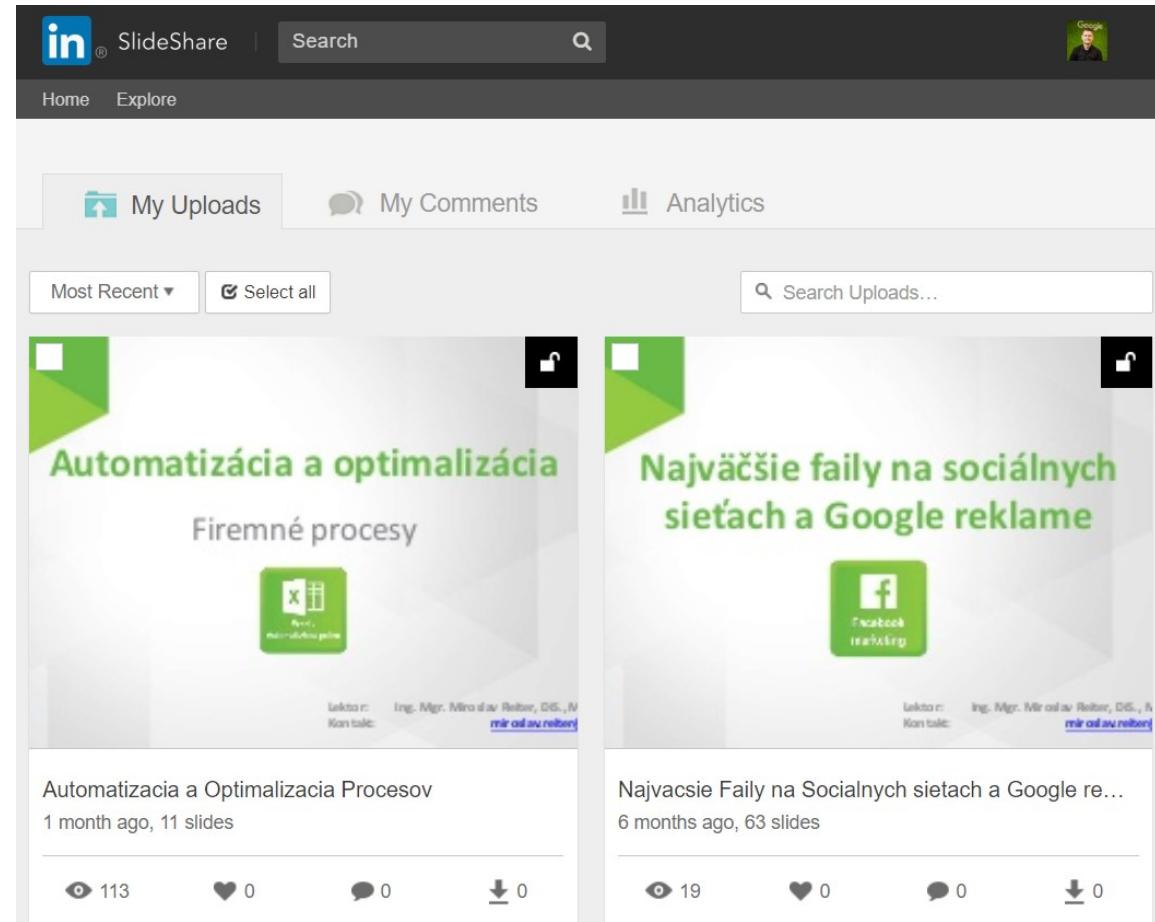
+ Anaconda Navigator

- <https://www.jetbrains.com/pycharm/>
- <https://www.anaconda.com/products/>

## 2. Pridaj si ma na LinkedIn

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

## 3. Prezentácia a materiály po prednáške



[Home](#) » Python - automatizácia úloh

## Python - automatizácia úloh

Python je jednoduchý programovací jazyk, ktorý používateľom umožňuje skriptovať automatizáciu. Výsledkom je časová úspora a jednoduchšia komunikácia a spolupráca. V tomto kurze sa môžete naučiť, ako automatizovať a zjednodušiť svoje opakujúce sa úlohy pomocou Pythonu.

### 1. Úvod do Automatizácie

- Čo je automatizácia?
- Čo a kedy automatizovať?
- Pycharm Jetbrains
- Knižnice a moduly pre automatizáciu
- Nastavenia Projektu a PIP

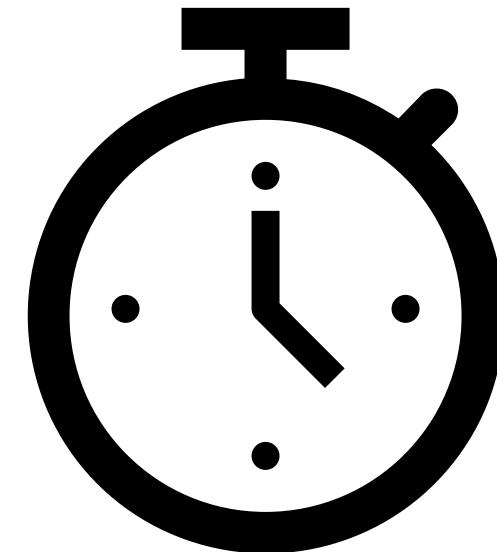
### 2. Automatizácia Úloh v Pythone

- Validácia úloh (PyInputPlus)
- Plánovanie úloh (schedule, time)
- Automatizácia v operačnom systéme Microsoft Windows (pyautogui)
- Automatizácia Dialogové Okná, Klávesnica, Myš, Screenshoty
- Automatizácia Práce so Súbormi (shutil)
- Kite a ChatGPT (AI code completions a snippets)

**Trvanie:**

# 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

115  
certifikácií

83 príručiek a  
publikácií

13 škôl

52 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



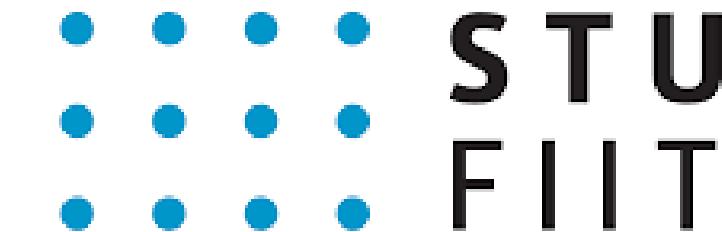
# Miroslav Reiter



1. PhDr. VŠM (Podnikový 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



# 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



392 kurzov v ponuke



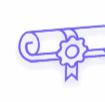
Zábavné online lekcie



Akreditované kurzy



11 rokov skúseností



Certifikovaní profesionálni lektori

Odporučame Kurzy špeciálne pre vás



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



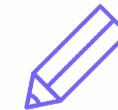
Online kurz Java I.  
Začiatočník  
67,00€ 88,40€



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



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



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



Online kurz SQL I.  
Začiatočník  
89,00€ 112,00€

# Moje začiatky

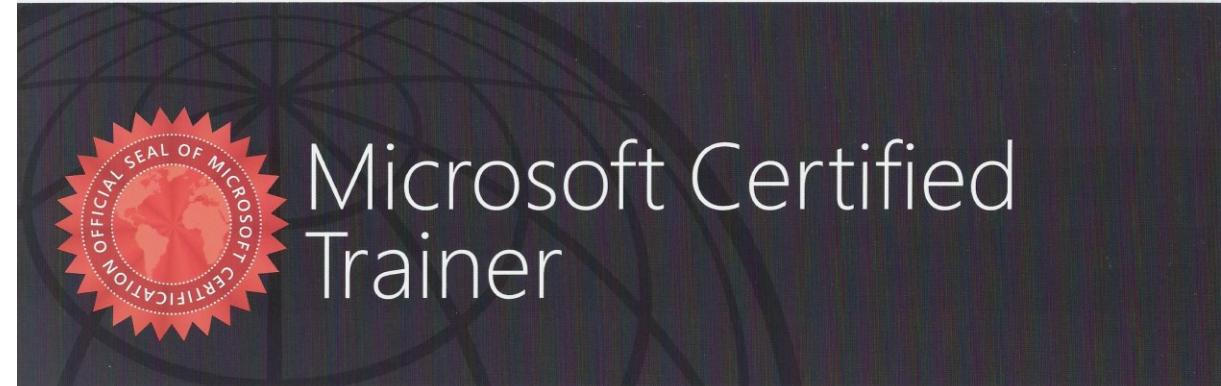


# Miroslav Reiter

získava status  
**Google Certified Trainer**

Automation

Google



MIROSLAV REITER

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

N. S. [Signature]  
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



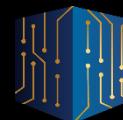
National competence centre for high performance computing  
SLOVAKIA



EURO



**EuroHPC**  
Joint Undertaking



NATIONAL  
SUPERCOMPUTING  
CENTRE



#### Vzdelávanie

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



#### Propagácia

Prednášky:  
<https://eurocc.nscc.sk/vzdelavanie/prednasky/prednasky-archiv/>



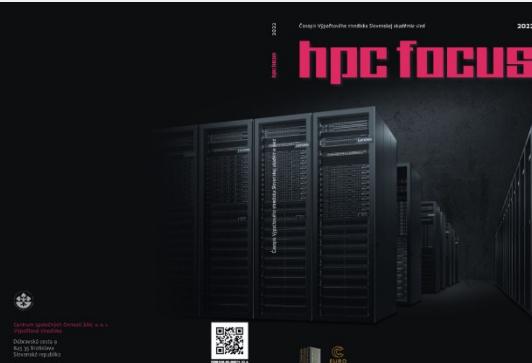
#### HPC služby

Prístup k  
výpočtovým  
prostriedkom



#### Spolupráca

Pilotné projekty  
  
Dlhodobá  
spolupráca



robíme it



Qubit  
Conference

### S kým spolupracujeme:

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



**Sledujte nás na sociálnych sietiach:**  
**#nccprehpc**



# Interaktívna prednáška

# Aktívne používanie a zapájanie sa

Participants (20)

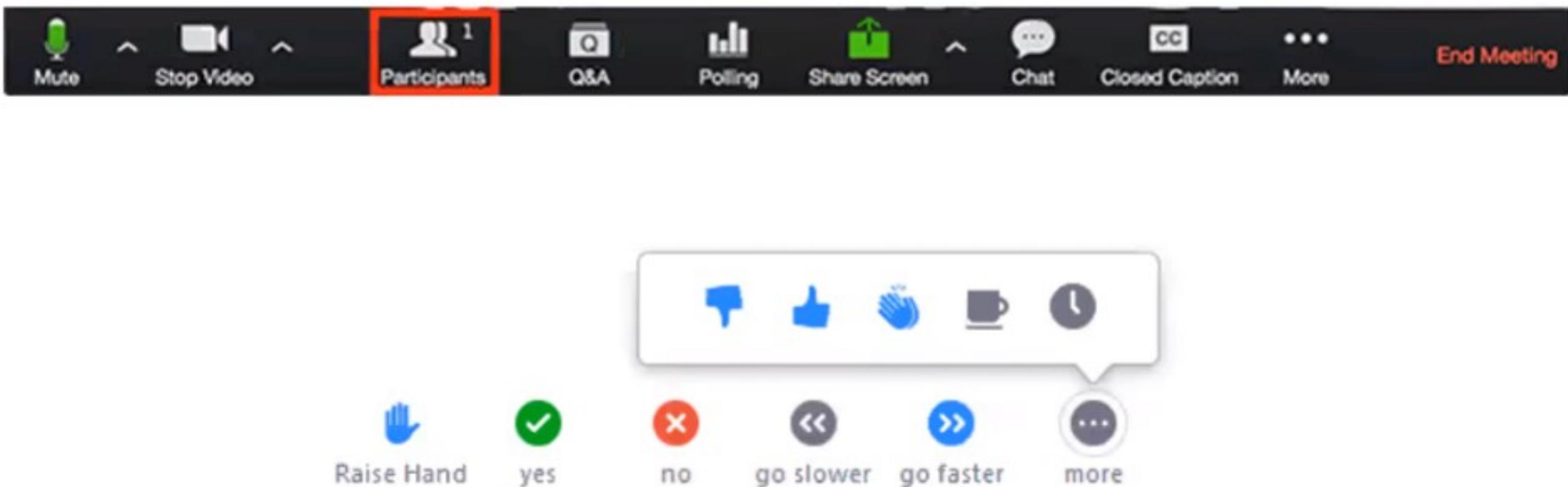
Find a participant

Participant	Color Swatches	Microphone (Green)	Video (Blue)
MR Miroslav Reiter (Me)			

Raise Hand      yes      no      go slower      go faster      more

Invite      Mute Me

# Používame Zoom



# Vaše ciele a očakávania

1. Doplniť si znalosti z jazyka Python

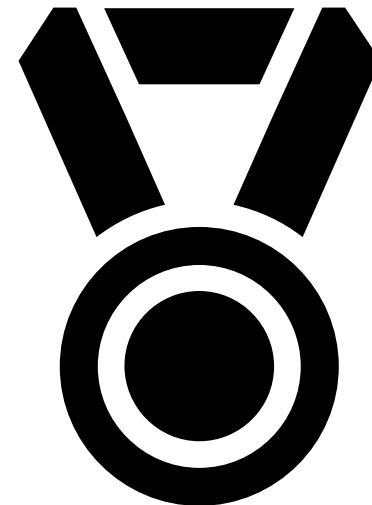
2. Základy analytického myslenia

3. Základy spracovania dát

4. Základy automatizácie

5. Doplniť si znalosti z JetBrains PyCharm

Zábava je v zaručená v každom bode :-)



# Ako automatizovať úlohy pomocou Pythonu?



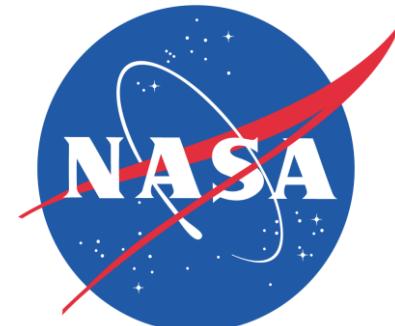
SEZNAMEM.cz

Google

Instagram



NETFLIX

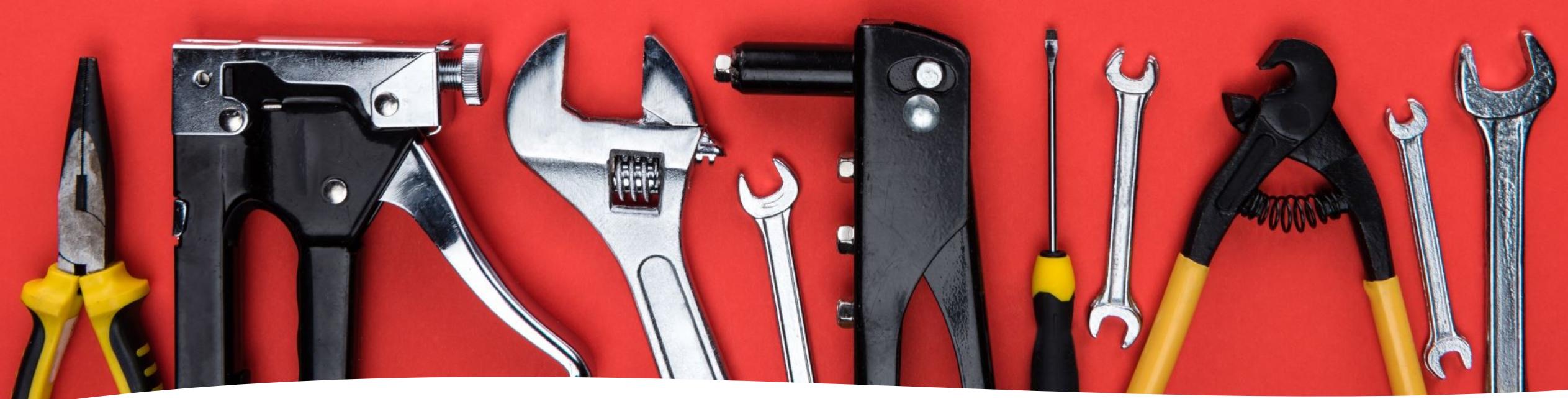


Čo majú tieto  
spoločnosti  
spoločné?

# čo je Automatizácia?

- úloh
- procesov
- práce



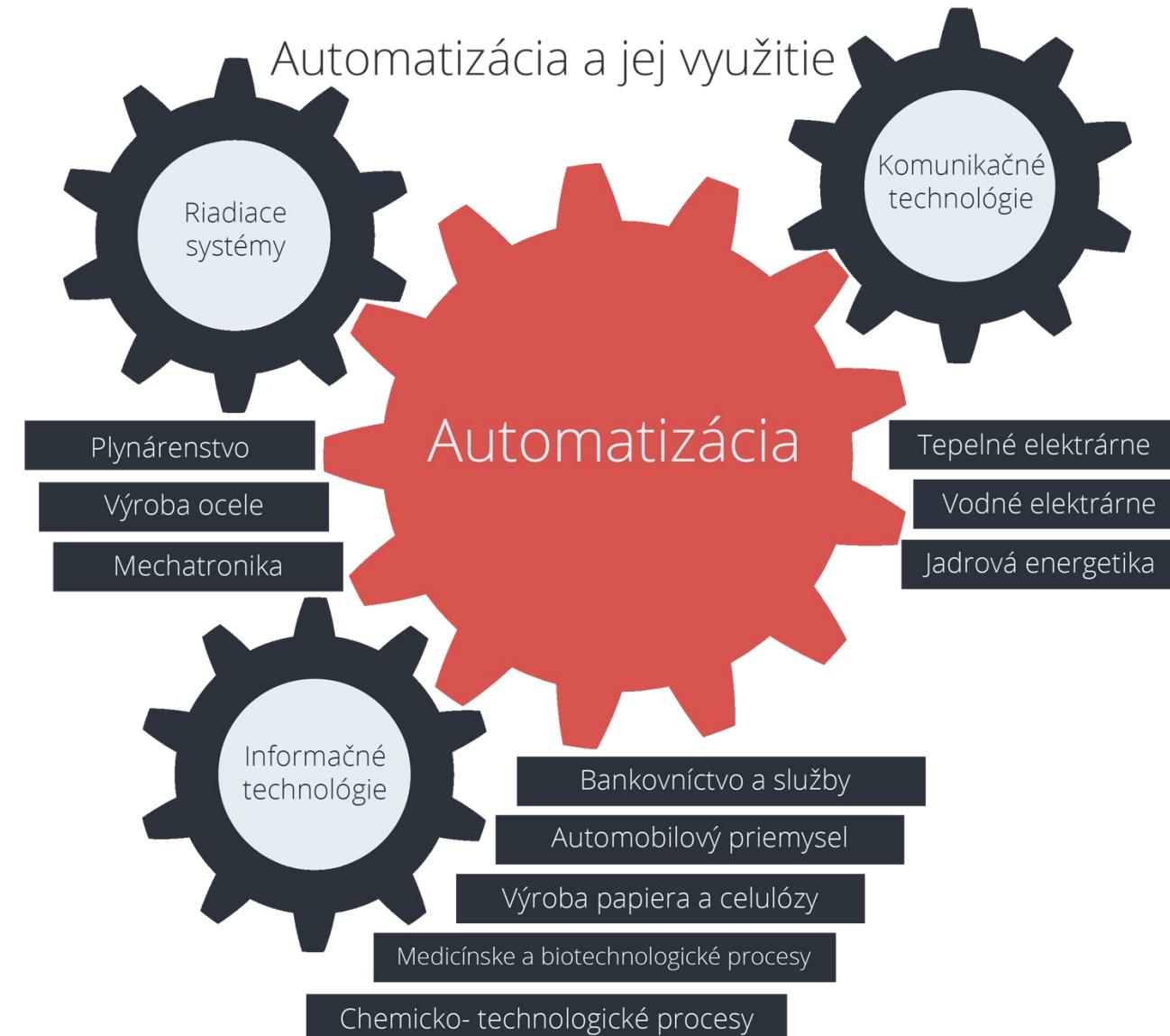


# Automatizácia

- Použitie samočinných riadiacich systémov **na riadenie technologických zariadení a procesov**.
- Z pohľadu industrializácie ide o **krok nasledujúci po mechanizácii**.
- Zatiaľ čo mechanizácia poskytuje ľuďom na prácu zariadenie, ktoré im uľahčuje prácu, **automatizácia znižuje potrebu prítomnosti človeka pri vykonávaní určitej činnosti**.
- Pri splnení ideálneho predpokladu tzv. **komplexnej automatizácie by teoreticky mohlo dôjsť až k vyradeniu človeka** z príslušného výrobného procesu.
- Moderné poňatie výrobnej automatizácie stavia **človeka** a **automatizačnú techniku** do **úlohy partnerov**: **automatizačná technika pomáha ľuďom udržiavať technologické procesy efektívne a bezpečne**.

# Automatizácia

- Je etapa rozvoja techniky, ktorá uskutočňuje **výrobné, riadiace a iné procesy bez priameho zásahu človeka**, spojená s objavom automatických výrobných liniek, automatických prevádzok a závodov, s využívaním modernej výpočtovej a riadiacej techniky
- Automatizácia je **prienikom poznatkov rôznych oborov**: elektrotechniky, strojárstva, **informatiky**
- Integruje ich a vytvára **efektívnejšie riadenie technologických postupov**



# Automatizácia Procesov

- ✓ Spočíva v **optimalizácii úloh alebo aktivít** prostredníctvom **softvéru**
- ✓ Tieto procesy musia byť **efektívne, znižovať náklady a čas realizácie**



# Ideme šetriť

## Čas

- Zbytočné prechádzanie dokumentov/kódov
- Zbytočné opakovanie úloh
- Komunikácia s klientom

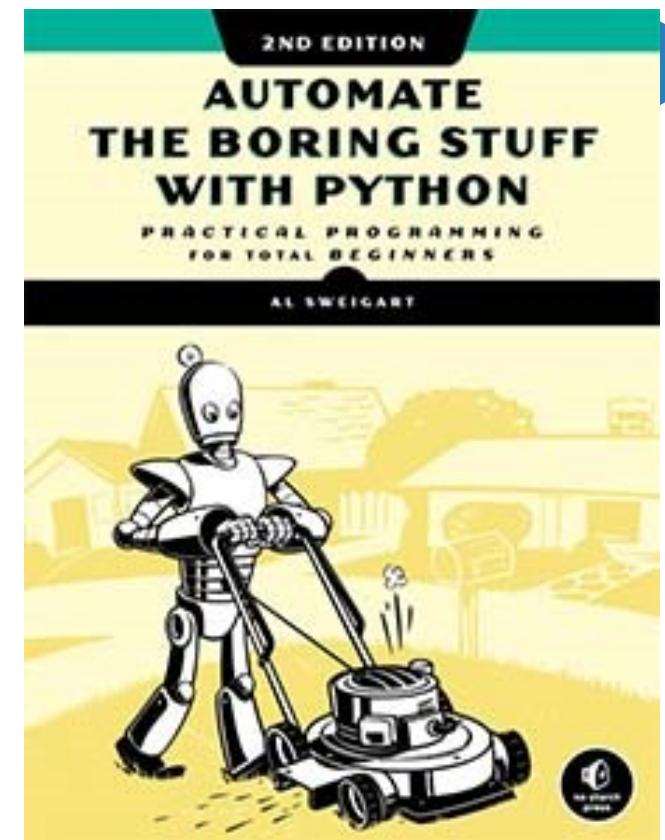
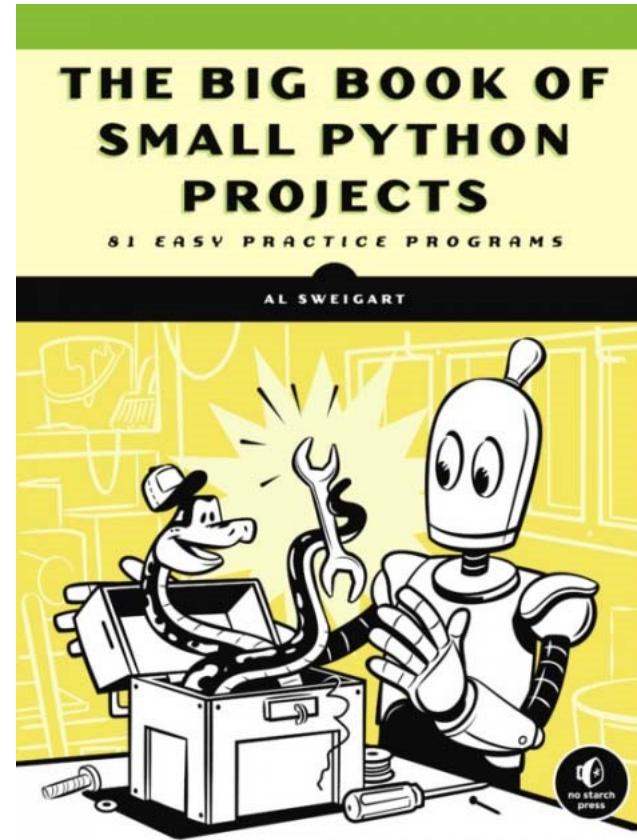
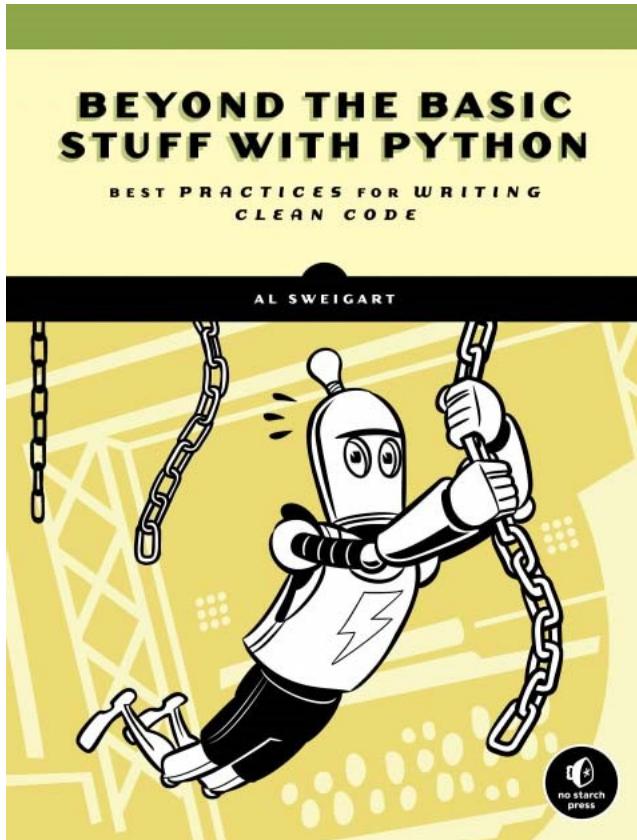
## Peniaze

- Lepšia optimalizácia
- Aj tak potrebujeme dokumentáciu
- Prístup k údajom a procesom inak nie je k dispozícii

10-15 hodín za každý týždeň

15-30 % nižšie náklady

Nervy a energiu



# Al Sweigart (Automate)

# Základná otázka k automatizácii

Ktorý z týchto **procesov** je **najlepším kandidátom na automatizáciu?**

- A. Komplexný proces, ktorému ľudia nerozumejú a robia veľa omylov
- B. Jednoduchý proces, ktorý sa čas od času praktizuje a ľudia si nepamätajú správny postup
- C. Jednoduchý proces, ktorý sa vykonáva často a správne
- D. Komplexný proces, dobre opísaný, ale veľmi ťažký

# Mám použiť automatizáciu alebo to urobiť ručne?

1. Zaberá táto úloha pravidelne viac ako hodinu?

2. Vyžaduje táto úloha len málo rozhodovacích schopností?

3. Musí sa táto úloha uskutočniť mimo otváracích/prevádzkových hodin?

4. Zodpovedá táto úloha veľkému množstvu údajov?

HOW LONG CAN YOU WORK ON MAKING A ROUTINE TASK MORE  
EFFICIENT BEFORE YOU'RE SPENDING MORE TIME THAN YOU SAVE?  
(ACROSS FIVE YEARS)



		HOW OFTEN YOU DO THE TASK					
		50/DAY	5/DAY	DAILY	WEEKLY	MONTHLY	YEARLY
HOW MUCH TIME YOU SHAVE OFF	1 SECOND	1 DAY	2 HOURS	30 MINUTES	4 MINUTES	1 MINUTE	5 SECONDS
	5 SECONDS	5 DAYS	12 HOURS	2 HOURS	21 MINUTES	5 MINUTES	25 SECONDS
	30 SECONDS	4 WEEKS	3 DAYS	12 HOURS	2 HOURS	30 MINUTES	2 MINUTES
	1 MINUTE	8 WEEKS	6 DAYS	1 DAY	4 HOURS	1 HOUR	5 MINUTES
	5 MINUTES	9 MONTHS	4 WEEKS	6 DAYS	21 HOURS	5 HOURS	25 MINUTES
	30 MINUTES		6 MONTHS	5 WEEKS	5 DAYS	1 DAY	2 HOURS
	1 HOUR		10 MONTHS	2 MONTHS	10 DAYS	2 DAYS	5 HOURS
	6 HOURS				2 MONTHS	2 WEEKS	1 DAY
	1 DAY					8 WEEKS	5 DAYS

# Benefity Automatizácie pre váš biznis aj osobný život

1. Mnoho podnikateľov uprednostňuje technológiu automatizácie na zníženie prevádzkových nákladov, zvýšenie výnosov, zlepšenie skúseností zákazníkov a zvýšenie efektívnosti
2. Automatizujte a optimalizujte rutinné procesy a úlohy, ktoré pomáhajú šetriť čas a peniaze
3. Zvyšuje prevádzkovú efektivitu a produktivitu
4. Na základe výstupov je možné rýchlejšie podniknúť rozhodnutia z kognitívnych technológií
5. Kontroluje chyby a ľudské chyby
6. Generuje kvalitné leady a vytvára zákaznícku základňu
7. Identifikuje a maximalizuje predajné príležitosti na zvýšenie výnosov
8. Ponúka inteligentné poradenstvo a predpovedá preferencie zákazníkov

File Edit View Navigate Code Refactor Run Tools VCS Window Help WIFI - main.py

WIFI > main.py

Project

WIFI C:\Users\Administrator\PycharmProjects\WIFI

venv library root

main.py

External Libraries

Scratches and Consoles

main.py

```
import subprocess
```

Code With Me: Start Session

Set permissions for new guests:

Read-only  Edit files  Full access  Custom

Guests are not allowed to make any changes on your machine.

▶ Permissions Details

Start call

Audio and video are off by default. You can activate them on the call.

Start Session Cancel

Analyzing...

Výstraha Windows Zabezpečenia

Brána Windows Defender Firewall zablokovala niektoré funkcie tejto aplikácie.

Brána Windows Defender Firewall zablokovala niektoré funkcie programu Java Chromium Embedded Framework (JCEF) Helper vo všetkých verejných, súkromných a doménových

Názov: Java Chromium Embedded Framework (JCEF) Helper

Ydavateľ: Neznáme

Cesta: C:\program files\jetbrains\pycharm 2022.3\jbr\bin\jcef\_helper.exe

Povoliť pre Java Chromium Embedded Framework (JCEF) Helper komunikáciu v týchto

Doménové siete, napríklad pracovná siet

Súkromné siete, napríklad domáca alebo pracovná siet

Verejné siete, napríklad letiská a kaviarne (neodporúča sa, pretože tieto siete sú často nedostatočne zabezpečené alebo sú nezabezpečené)

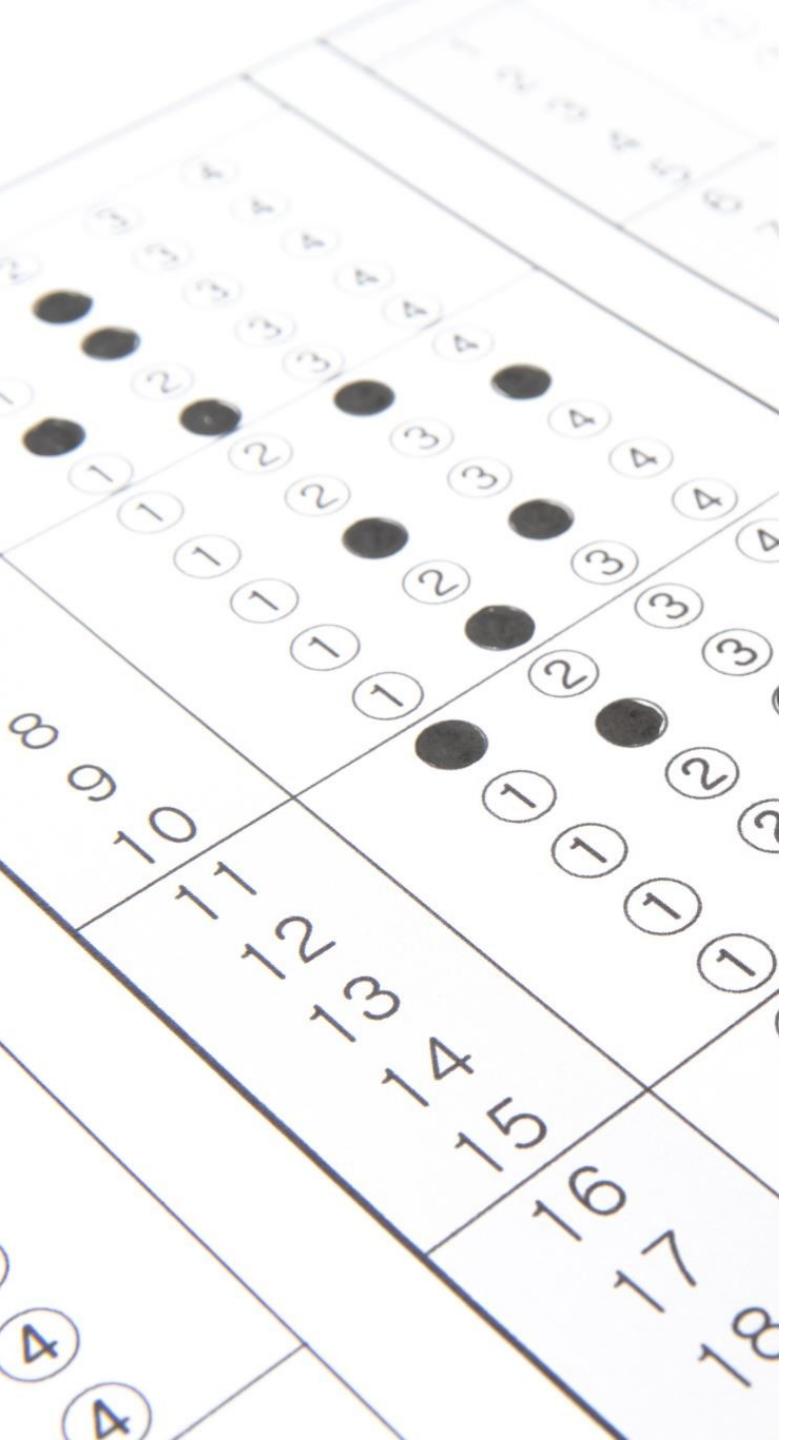
Aké sú riziká povolenia vstupu aplikácií cez bránu firewall?

Povoliť prístup Zrušiť

Version Control Python Packages TODO Python Console Problems Terminal Services

Reloading generated ske

# Príklady na Automatizáciu



# Task Scheduling

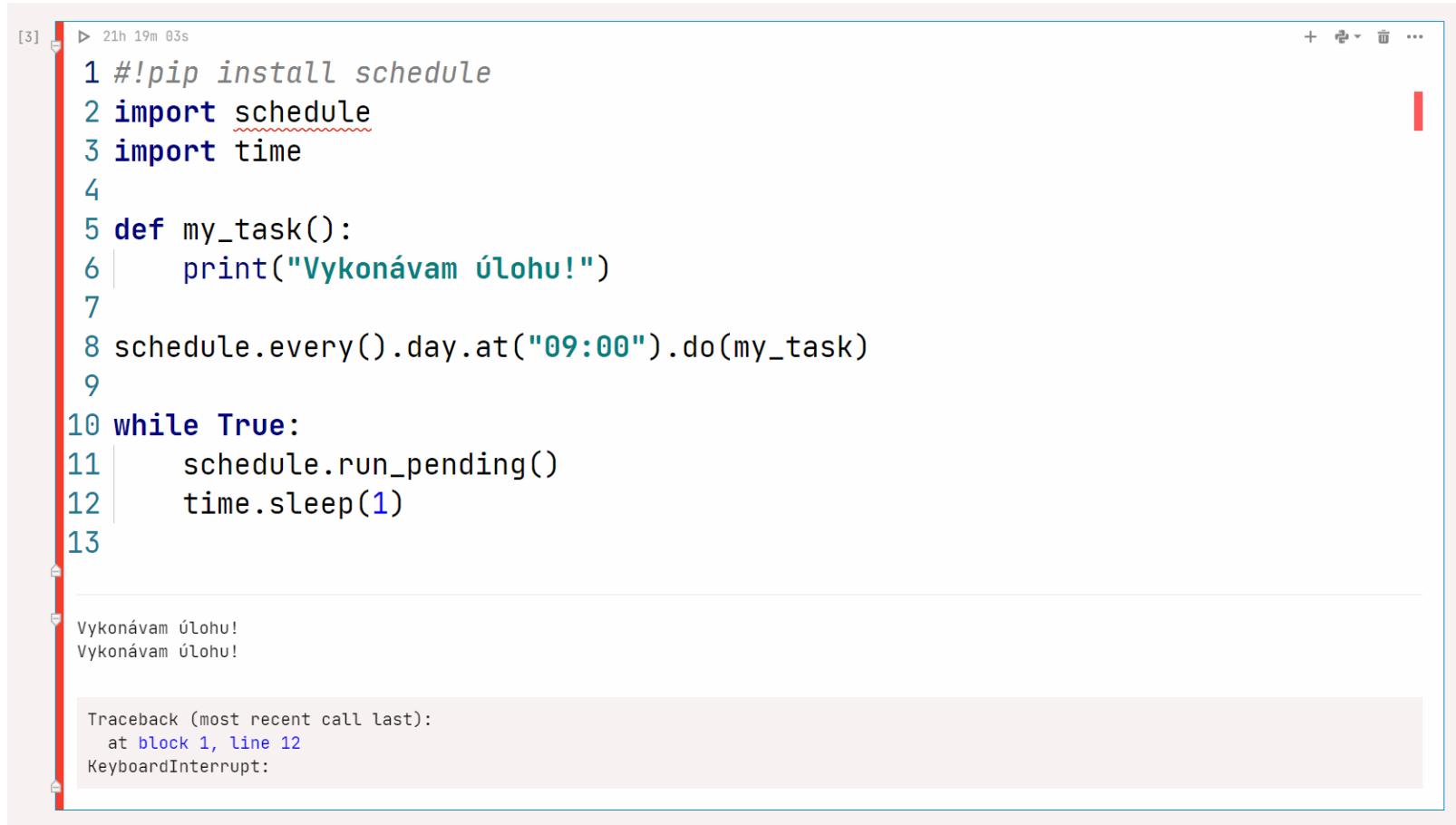
---

# Ako na Plánovanie Úloh? (Task Scheduling)

---



# Plánovanie Úloh (Task Scheduling)



The screenshot shows a terminal window with the following Python code:

```
[3] > 21h 19m 03s
1 #!pip install schedule
2 import schedule
3 import time
4
5 def my_task():
6     print("Vykonávam úlohu!")
7
8 schedule.every().day.at("09:00").do(my_task)
9
10 while True:
11     schedule.run_pending()
12     time.sleep(1)
13
```

The output of the script is displayed below the code:

```
Vykonávam úlohu!
Vykonávam úlohu!
```

A tooltip at the bottom shows the traceback:

```
Traceback (most recent call last):
  at block 1, line 12
KeyboardInterrupt:
```

pip install schedule

# Modul Schedule

- Plánovanie úloh v Pythone pre ľudí
- Pravidelné spúšťanie funkcií Pythonu (alebo akékoľvek iné volateľné) pomocou priateľskej syntaxe
- Jednoducho použiteľné API na plánovanie úloh
- Priebežný plánovač pre pravidelné úlohy
- Nie sú potrebné žiadne ďalšie procesy!
- Veľmi ľahký a bez vonkajších závislostí
- Vynikajúce pokrytie testov
- Testované na Pythone 3.6-3.9

```
# Autor: Miroslav Reiter
# Kurz: Python Automatizácia Práce

import schedule
import time

def my_task():
    print("Vykonávam úlohu!")

schedule.every().day.at("09:00").do(my_task)

while True:
    schedule.run_pending()
    time.sleep(1)
```

# Modul Schedule

- Knižnica je navrhnutá ako jednoduché riešenie pre jednoduché problémy s plánovaním
- Načo nie je stavaná:
  1. Trvanie úlohy (zapamätajte si plán medzi reštartmi)
  2. Presné načasovanie (s presnosťou na menej ako sekundy)
  3. Súbežné vykonávanie (viaceré vlákna)
  4. Lokalizácia (časové pásma, pracovné dni alebo sviatky)
- Schedule nezohľadňuje čas potrebný na vykonanie funkcie úlohy. Ak chcete zaručiť stabilný plán vykonávania, musíte presunúť dlhotrvajúce úlohy z hlavného vlákna (kde beží plánovač)

```
# Autor: Miroslav Reiter
# Kurz: Python Automatizácia Práce

import schedule
import time

def my_task():
    print("Vykonávam úlohu!")

schedule.every(5).minutes.do(my_task)
schedule.every().hour.do(my_task)
schedule.every().day.at("9:00").do(my_task)
schedule.every().monday.do(my_task)
schedule.every().wednesday.at("13:30").do(my_task)
schedule.every().minute.at(":15").do(my_task)

while True:
    schedule.run_pending()
    time.sleep(1)
```



SijmenHuizenga and chrimaho Add timezone support for .at() (#517) ...

✓ 3eac646 on Apr 23 ⏱ 381 commits

📁 .github/workflows	Split up Github Actions (#442)	last year
📁 docs	Add timezone support for .at() (#517)	8 months ago
📁 schedule	Add timezone support for .at() (#517)	8 months ago
📄 .gitignore	Log all messages as debug	2 years ago
📄 AUTHORS.rst	Release 1.1.0 (#455)	last year
📄 HISTORY.rst	Release 1.1.0 (#455)	last year
📄 LICENSE.txt	Initial code commit.	9 years ago
📄 MANIFEST.in	Release 0.4.0	6 years ago
📄 README.rst	Add timezone support for .at() (#517)	8 months ago
📄 requirements-dev.txt	Add timezone support for .at() (#517)	8 months ago
📄 setup.cfg	Add type annotations (#427)	last year
📄 setup.py	Add py.typed file (#521)	8 months ago
📄 test_schedule.py	Add timezone support for .at() (#517)	8 months ago
📄 tox.ini	Add timezone support for .at() (#517)	8 months ago

## README.rst

# schedule



Tests

passing

coverage

100%

pypi

v1.1.0

Python job scheduling for humans. Run Python functions (or any other callable) periodically using a friendly syntax.

- A simple to use API for scheduling jobs, made for humans.
- In-process scheduler for periodic jobs. No extra processes needed!
- Very lightweight and no external dependencies.
- Excellent test coverage.
- Tested on Python and 3.6, 3.7, 3.8, 3.9

Python job scheduling for humans.

🔗 [schedule.readthedocs.io/](https://schedule.readthedocs.io/)

📖 Readme

⚠ MIT license

⭐ 10.4k stars

👁 209 watching

🍴 889 forks

## Releases

🏷 24 tags

## Packages

No packages published

## Used by 21.2k



+ 21,239

## Contributors 51



+ 40 contributors

## Languages

Python 100.0%

# schedule



10,385

## Table of Contents

[Installation](#)[Examples](#)[Run in the background](#)[Parallel execution](#)[Exception Handling](#)[Logging](#)[Multiple schedulers](#)[Frequently Asked](#)[Questions](#)[Reference](#)[Development](#)[History](#)

## Useful Links

[Schedule @ GitHub](#)[Schedule @ PyPI](#)[Issue Tracker](#)

## More Python

[Follow @dbader\\_org](#)[Dan's Python Screencasts](#)[Dan's Python Newsletter](#)[Dan's Python Tutorials](#)

## Quick search

Go

# schedule

 [Tests](#)
 passing
 [coverage](#)
100%
 [pypi](#)
v1.1.0

Python job scheduling for humans. Run Python functions (or any other callable) periodically using a friendly syntax.

- A simple to use API for scheduling jobs, made for humans.
- In-process scheduler for periodic jobs. No extra processes needed!
- Very lightweight and no external dependencies.
- Excellent test coverage.
- Tested on Python 3.6, 3.7, 3.8 and 3.9

## Example

```
$ pip install schedule

import schedule
import time

def job():
    print("I'm working...")

schedule.every(10).minutes.do(job)
schedule.every().hour.do(job)
schedule.every().day.at("10:30").do(job)
schedule.every().monday.do(job)
schedule.every().wednesday.at("13:15").do(job)
schedule.every().minute.at(":17").do(job)

while True:
    schedule.run_pending()
    time.sleep(1)
```

[More Examples](#)

## When **not** to use Schedule

Let's be honest, Schedule is not a 'one size fits all' scheduling library. This library is designed to be a simple solution for simple scheduling problems. You should probably look somewhere else if you need:

v: stable



Search projects



Help

Sponsors

Log in

Register

# PyInputPlus 0.2.12

`pip install PyInputPlus` 



Latest version

Released: Oct 11, 2020

Provides more featureful versions of `input()` and `raw_input()`.

## Navigation

 Project description

 Release history

 Download files

## Project links

 Homepage

## Statistics

GitHub statistics:

 Stars: 60

 Forks: 13

 Open issues/PRs: 19

View statistics for this project via  
[Libraries.io](#), or by using our public

## Project description

A Python 2 and 3 module to provide `input()`- and `raw_input()`-like functions with additional validation features, including:

- Re-prompting the user if they enter invalid input.
- Validating for numeric, boolean, date, time, or yes/no responses.
- Timeouts or retry limits for user responses.
- Specifying regexes for whitelists or blacklists of responses.
- Specifying ranges for numeric inputs.
- Presenting menus with bulleted, lettered, or numbered options.
- Allowing case-sensitive or case-insensitive responses.

## Installation

`pip install pyinputplus`

## Example Usage

```
>>> import pyinputplus as pyip
```

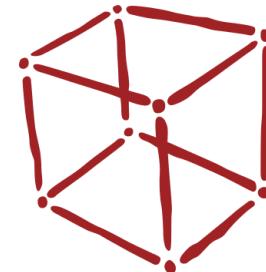


# Pizza Maker

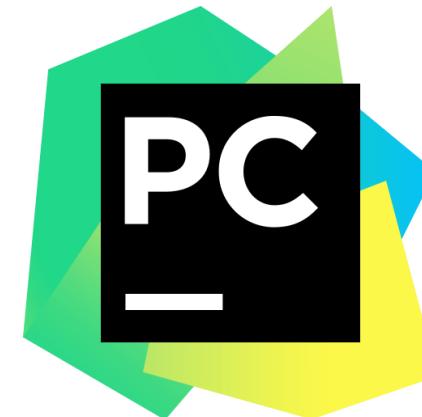
Napíšte program, ktorý sa používateľov opýta na preferencie pizze. Program by mal použiť PyInputPlus, aby zabezpečil, že zadajú platný vstup, ako napríklad:

- A. Použitie inputMenu() pre typ cesta: tenké, hrubé alebo polohrubé.
- B. Použitie inputMenu() pre typ mäsa: šunka, saláma, slanina, kuracie, alebo bez mäsa.
- C. Pomocou inputYesNo() sa spýtate, či chcú syr.
- D. Ak áno, pomocou inputMenu() požiadajte o typ syra: mozzarella, parmezán alebo eidam.
- E. Pomocou inputYesNo() sa spýtate, či chcú donášku domov.
- F. Pomocou inputInt() sa spýtate, koľko pizze chcú. Uistite sa, že toto číslo je 1 alebo viac.
- G. Zaznamenajte voľby do textového súboru.

# Aké IDE mám použiť?



## NetBeans



## WxPython



## Visual Studio



# Chceme úplne všetko!

---



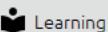
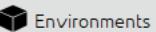
 ANACONDA NAVIGATOR[Sign in to Anaconda Cloud](#)[Home](#)

Applications on

base (root)

Channels

Refresh



Documentation

Developer Blog





**CMD.exe Prompt**  
0.1.1  
Run a cmd.exe terminal with your current environment from Navigator activated

[Launch](#)



**JupyterLab**  
1.2.6  
An extensible environment for interactive and reproducible computing, based on the Jupyter Notebook and Architecture.

[Launch](#)



**jupyter**  
Notebook  
6.0.3  
Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.

[Launch](#)



**Powershell Prompt**  
0.0.1  
Run a Powershell terminal with your current environment from Navigator activated

[Launch](#)



**Qt Console**  
4.6.0  
PyQt GUI that supports inline figures, proper multiline editing with syntax highlighting, graphical calltips, and more.

[Launch](#)



**Spyder**  
4.0.1  
Scientific PYthon Development EnviRonment. Powerful Python IDE with advanced editing, interactive testing, debugging and introspection features

[Launch](#)



**VS Code**  
1.52.1  
Streamlined code editor with support for development operations like debugging, task running and version control.

[Launch](#)



**Glueviz**  
0.15.2  
Multidimensional data visualization across files. Explore relationships within and among related datasets.

[Install](#)



**Orange 3**  
3.23.1  
Component based data mining framework. Data visualization and data analysis for novice and expert. Interactive workflows with a large toolbox.

[Install](#)



**RStudio**  
1.1.456  
A set of integrated tools designed to help you be more productive with R. Includes R essentials and notebooks.

[Install](#)

Microsoft Store

python

Filtre

# "python"

Všetky oddelenia Aplikácie Hry

Python 3.9

Aplikácie • Developer tools

★★★★★ 4

Nainštalované

Python 3.8

Aplikácie • Developer tools

Bezplatné

Python 3.10

Aplikácie • Developer tools

Bezplatné

Python 3.7

Aplikácie • Developer tools

★★★★★ 1

Bezplatné

Learn Django and Python by GoLearningBus

Aplikácie • Books & reference

Bezplatné

WiBit.Net :: Programming in Python

Aplikácie • Education

0,99 €

计算机二级 Python 考试试题库

Aplicacie • Education

Bezplatné

Data Science with Python

Aplikácie • Education

Python Programs

Aplikácie • Education

Learn Python

Aplikácie • Books & reference

Introduction to Python Programming by...

Aplikácie • Books & reference

Python Playground

Aplikácie • Developer tools

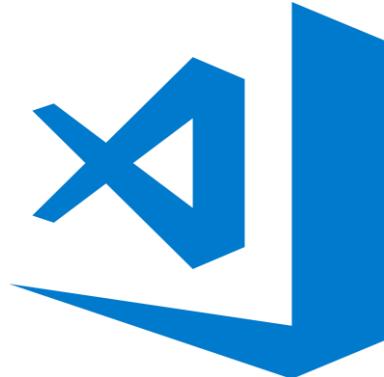
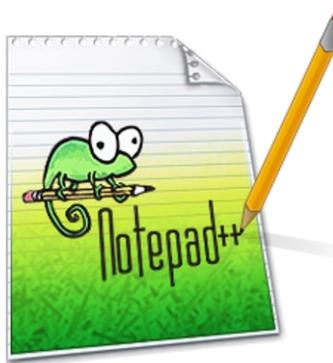
Python Programming Manual For Absolute...

Aplikácie • Utilities & tools

Python User Tutorial

Aplikácie • Utilities & tools

# Aký editor mám použiť?



```
:::  
iLE880j. :jD888880j:  
.LGitE888D.f8GjjjL8888E;  
iE :8888Et. .G8888.  
;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:**

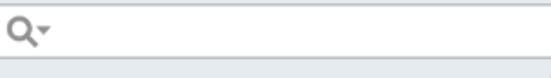
A password input field with a small circular icon containing a question mark to its right, likely indicating a password strength checker or help.

**Confirm password:**

A password input field with a small circular icon containing a question mark to its right, likely indicating a password strength checker or help.

**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)

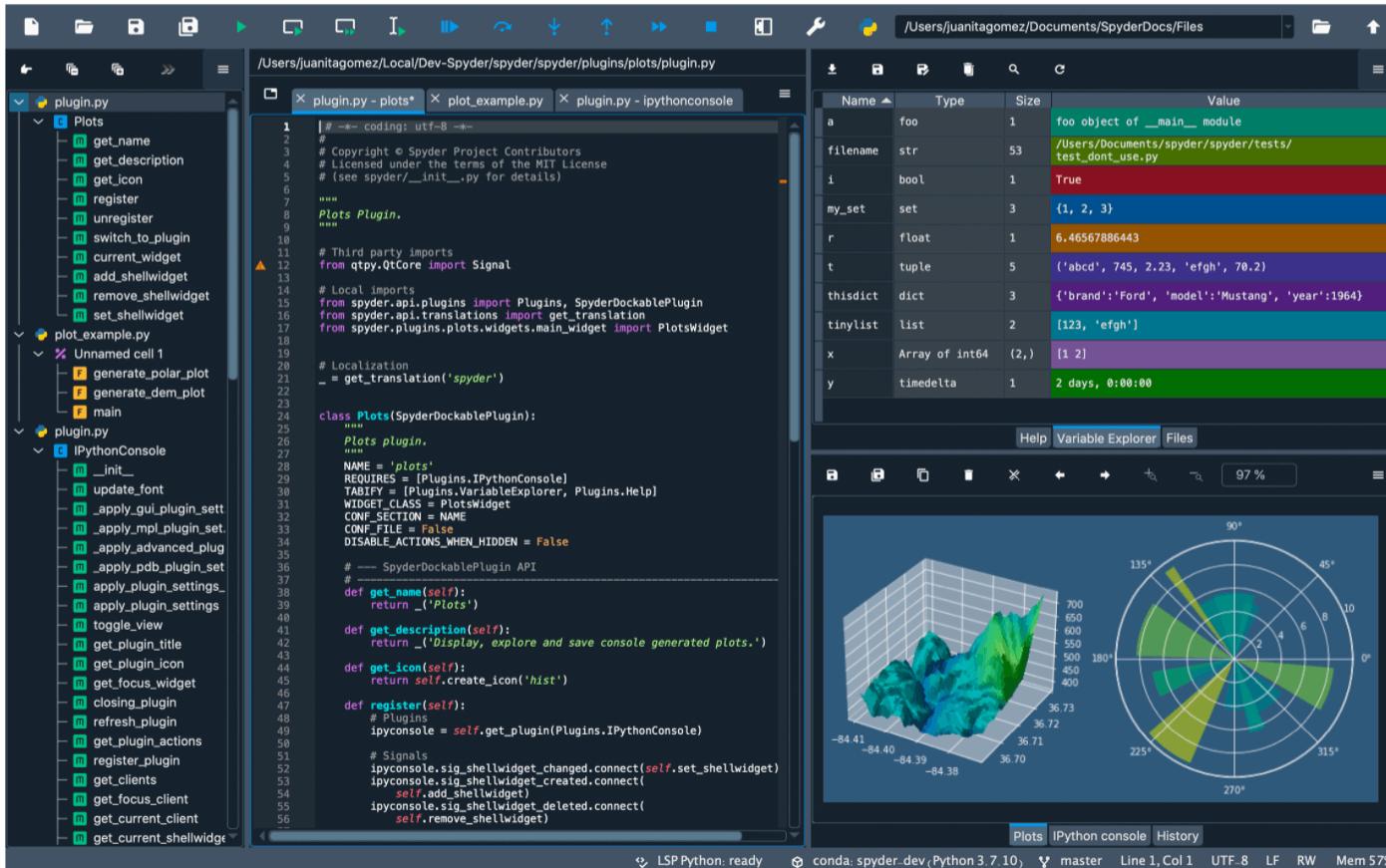
### R

### 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 radial data distribution.
- 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

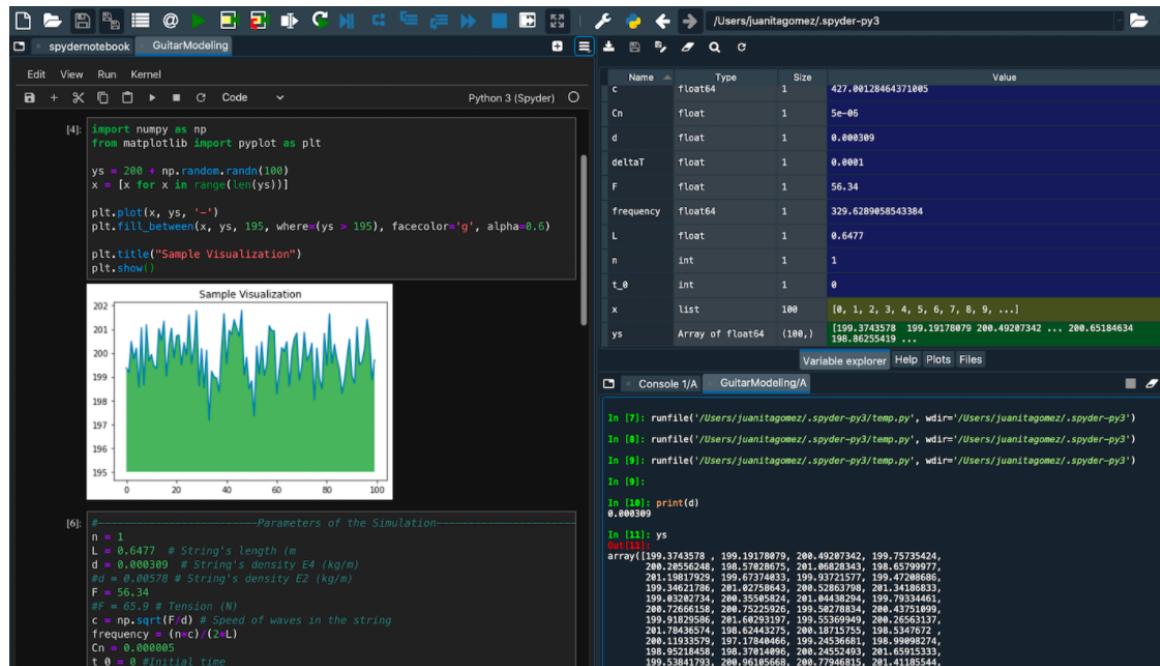
FAQ

# 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"]

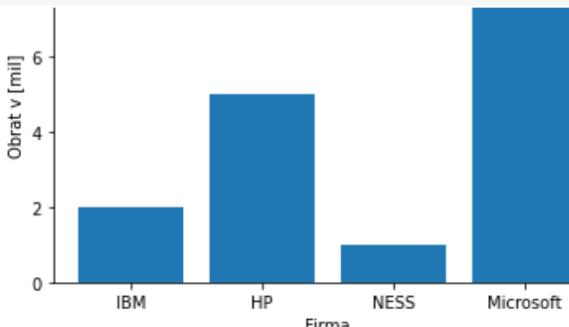
print(type(y))
print(y)

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

```

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]:

IPython console History



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() *	splines()	__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>		isspace() *	swapcase() *	<b>Date Formatting (strftime and strptime)</b>	
sys.argv[0]	foo.py	istitle() *	title() *	%a	Abbreviated weekday (Sun)
sys.argv[1]	bar	isupper() *	translate(table)	%A	Weekday (Sunday)
sys.argv[2]	-c	join()	upper() *	%b	Abbreviated month name (Jan)
sys.argv[3]	qux	ljust(width)	zfill(width)	%B	Month name (January)
sys.argv[4]	--h	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		File Methods	
altsep	Alternative sep	append(item)	pop(position)	close()	readlines(size)
curdir	Current dir string	count(item)	remove(item)	flush()	seek(offset)
defpath	Default search path	extend(list)	reverse()	fileno()	tell()
devnull	Path of null device	index(item)	sort()	isatty()	truncate(size)
extsep	Extension separator	insert(position, item)		next()	write(string)
linesep	Line separator			read(size)	writelines(list)
name	Name of OS			readline(size)	
pardir	Parent dir string				
pathsep	Patch separator				
sep	Path separator				
<b>Note</b>		Registered OS names: "posix", "nt", "mac", "os2", "ce", "java", "riscos"			
Class Special Methods		Indexes and Slices (of a=[0,1,2,3,4,5])		1.	
__new__(cls)	__lt__(self, other)	len(a)	6	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)	a[0]	0	2.	0 is Sunday, 6 is Saturday.
__del__(self)	__gt__(self, other)	a[5]	5	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)	a[-1]	5	4.	This is not a mistake. Range takes account of leap and double-leap seconds.
__str__(self)	__eq__(self, other)	a[-2]	4		
__cmp__(self, other)	__ne__(self, other)	a[1:]	[1,2,3,4,5]		
__index__(self)	__nonzero__(self)	a[:5]	[0,1,2,3,4]		
__hash__(self)		a[:-2]	[0,1,2,3]		
__getattr__(self, name)		a[1:3]	[1,2]		
__getattribute__(self, name)		a[1:-1]	[1,2,3,4]		
__setattr__(self, name, attr)		b=a[:]	Shallow copy of a		
__delattr__(self, name)					
__call__(self, args, kwargs)					

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

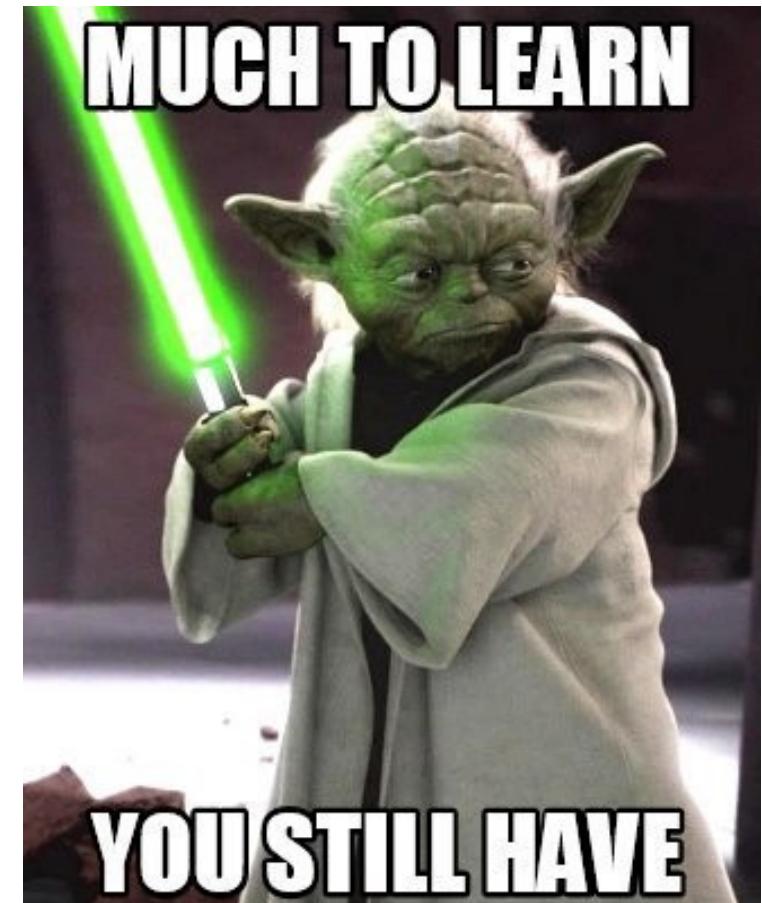
- O'Reilly
- Manning
- Packt
- Apress
- Wiley
- No Starch Press

## YouTube tutoriály

Packt Publishing

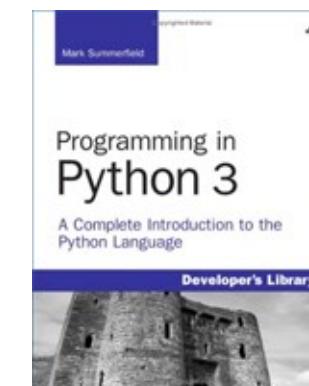
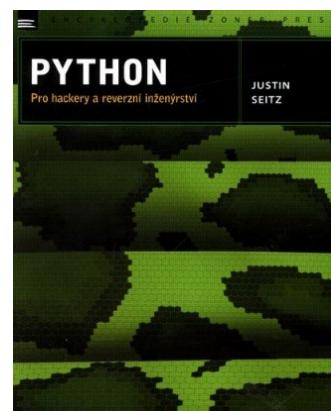
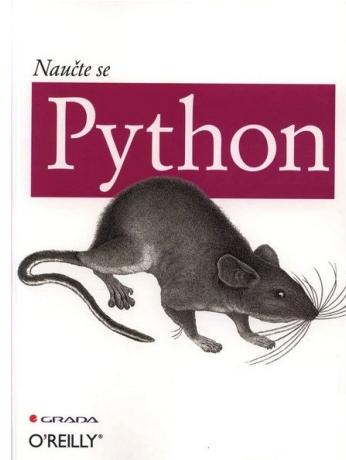
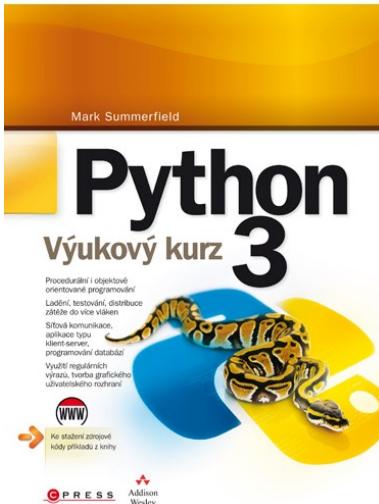
# Čo ti odporúčam si pozrieť?

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>



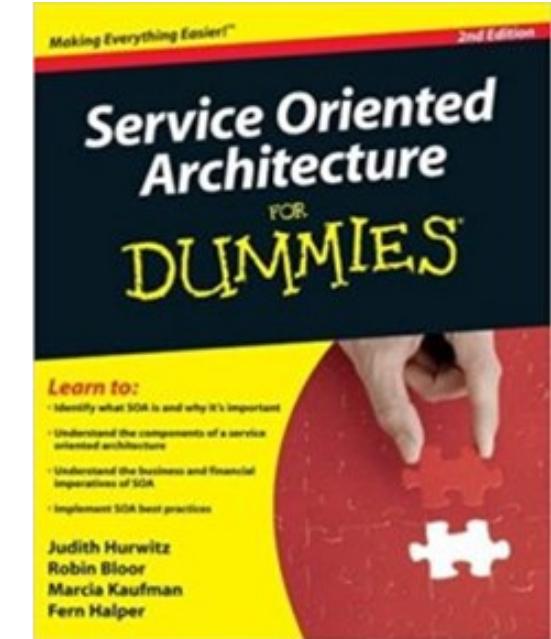
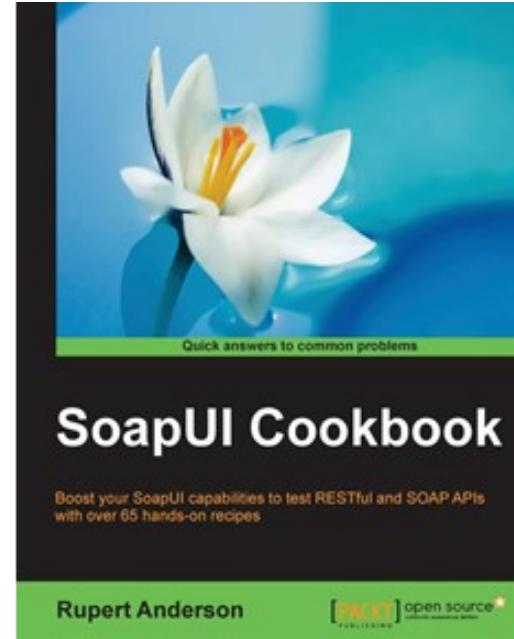
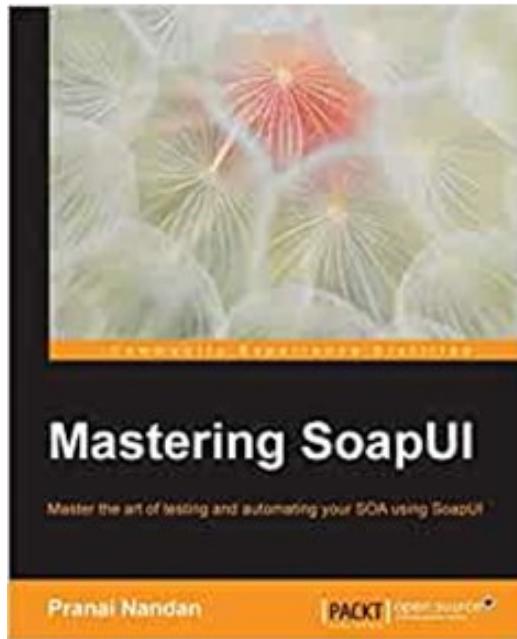
Šup do záložiek

# 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 layout with a focus on special characters. The top row of keys (Numpad 0-9, Backspace, Enter) and the F1-F12 keys are highlighted in light blue. Various special characters are overlaid on these keys, such as 'é', 'í', 'ó', 'ú', 'ä', 'ö', '§', 'ñ', '~', '!', '@', '#', '\$', '%', '^', '&', '<', '>', '<^>', '&', '|', ':', '?', '(', ')', '[', ']', '{', '}', ',', '.', '-' on the Numpad, and similar characters on the F keys.

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

**Speciálne znaky na funkčných tlačidlach**

This section highlights the function keys F1 through F12, which are used for various keyboard macros and special functions. The F keys are shown with their corresponding special characters: F1 (~), F2 (!), F3 (@), F4 (#), F5 (\$), F6 (%), F7 (^), F8 (&), F9 (< >), F10 (<^>), F11 (&|:?), and F12 (<^>[{}]), all overlaid on the original F key labels.

# 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...

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



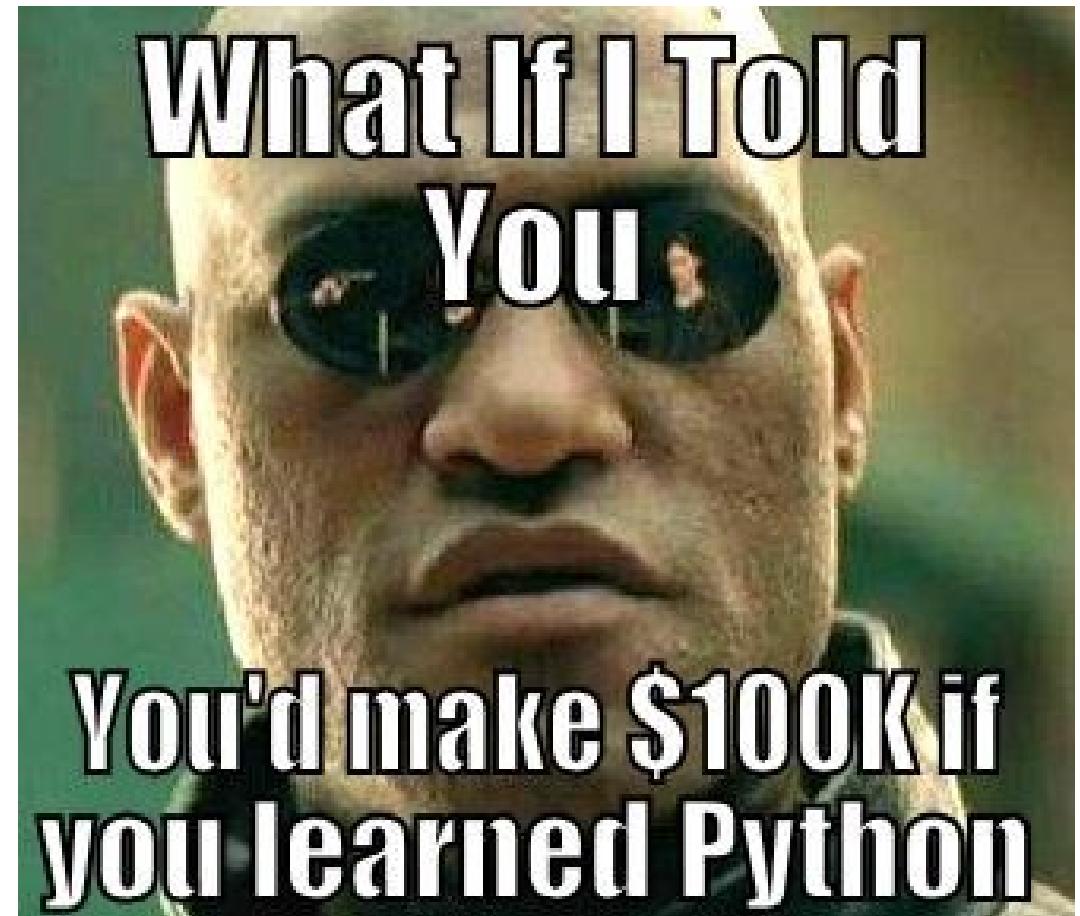
Vytvoriť novú skupinu

Hľadať



# 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



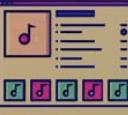
Quote Gener. Number guessing Dice Simulation YT downloader

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

V roku 2022 sa zameriava na pomoc:

- malým a stredným podnikateľom
- startupom
- neziskovým organizáciám
- ženám
- mnoho ďalším...



# Ako ti môžeme pomôcť?

Pre firmy

Pre jednotlivcov

Pre školy

Pre startupy

Väčšina

Získaj novú prácu

Získaj nové zručnosti

Začni podnikat

Pracuj lepšie na diaľku

Bud' v online svete v bezpečí

## Odporúčame

Objav naše tipy a odporúčania pre jednotlivcov



### NÁSTROJ

#### Získaj nové zručnosti potrebné pre rast tvojej kariéry

Uč sa svojím vlastným tempom, s flexibilnými a personalizovanými vzdelávacími kurzami navrhnutými tak, aby ti pomohli posunúť sa ďalej.

[Zistit viac](#)

[growwithgoogle.sk](http://growwithgoogle.sk)

**growwithgoogle.sk**

/prefirmy

Pomáha malým a stredným firmám získať digitálne zručnosti, ktoré sú **pre rast podnikania kľúčové**.

/preskoly

Ponúka rady a odporúčania na nástroje, ktoré môžu učiteľom, študentom, ale aj rodičom **pomôcť**.

/preneziskovky

Sumarizuje vzdelávacie materiály a videá, ktoré vám pomôžu byť viditeľný, **budovať komunity**, či riadiť tímy a pracovať efektívnejšie.

# Google Digitálna garáž

## Ako začať?



### Vyber si zručnosť

Či už si študent, alebo firma, na začiatku alebo v strede svojej cesty, môžeš si vybrať z lekcií tie, ktoré sú pre teba najvhodnejšie.



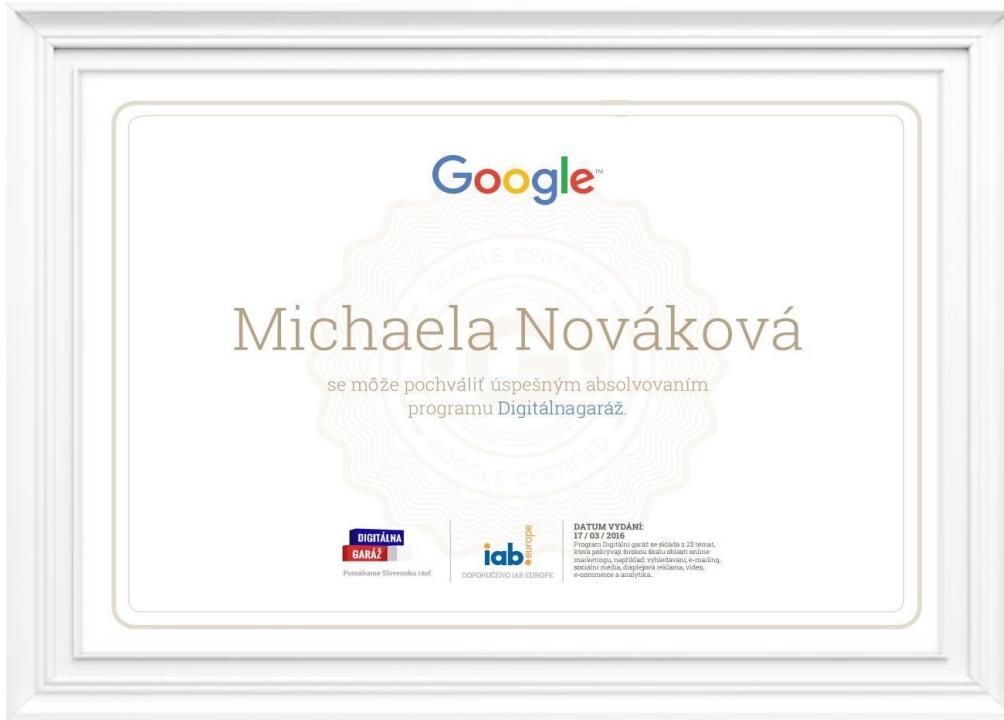
### Uč sa vlastným tempom

Vytvor si vlastný plán vzdelávania vo svojom počítači, smartfóne alebo tablete, podľa toho ako ti to bude najlepšie vyhovovať.



### Získaj certifikáciu

Sleduj svoj pokrok a získaj medzinárodne akreditovaný certifikát, ktorý obohatí tvoj životopis a profil v službe LinkedIn.



**26** téma

**106** lekcií

**9** hodin obsahu



**86 %**  
účastníkov našlo vďaka metóde Design Thinking nové řešenie na existujúci problém.

**63 %**  
účastník uvádza, že sú po absolvovaní workshopu viac kreatívni.

**56 %**  
účastník uvádza, že po absolvovaní workshopu lepšie spolupracujú v tíme.

Zúčastnite sa bezplatných workshopov New Generation of Founders a rozvíjajte svoje nápady a projekty!



**Design Thinking** je metóda riešenia problémov zameraná na používateľa. Začína pochopením jeho potrieb, pokračuje skúmaním inovatívnych riešení a končí testovaním rýchlych prototypov.

V rámci programu **New Generation of Founders** stretnete nových ľudí s rôznymi skúsenosťami a záujmami. Prejdete všetkými krokmi dizajnu s malým teoretickým základom a množstvom praktických cvičení.

**Máte už podnikateľský nápad alebo problém, ktorý chcete vyriešiť?** Skvelé! Na našom workshopu pochopíte, čo za týmto problémom stojí, ako sa vočiť do koze používateľa, a ako testovať a vylepsovať produkt.

# 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 dashboard. 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 three main tabs: "Home", "Explore", and "Analytics". The "Home" tab is selected. On the left side, there are three buttons: "My Uploads" (selected), "My Comments", and "Analytics". Below these buttons, there are two filters: "Most Recent" and "Select all". A search bar labeled "Search Uploads..." is located on the right. The main content area displays two presentation cards. The first card, titled "Automatizácia a optimalizácia Firemné procesy", has a green thumbnail featuring a white play button icon. It was uploaded by "Ing. Mgr. Miroslav Reiter, DIIS, N" (with a link to their profile) 1 month ago. The second card, titled "Najväčšie faily na sociálnych sietach a Google reklame", also has a green thumbnail with a white play button icon. It was uploaded by the same user 6 months ago. Both cards show download statistics at the bottom: 113 and 19 respectively.

# Ako sa s nami spojiť?



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

Budova KOLOSEO prízemie  
Tomášikova 50/A  
831 04 Bratislava



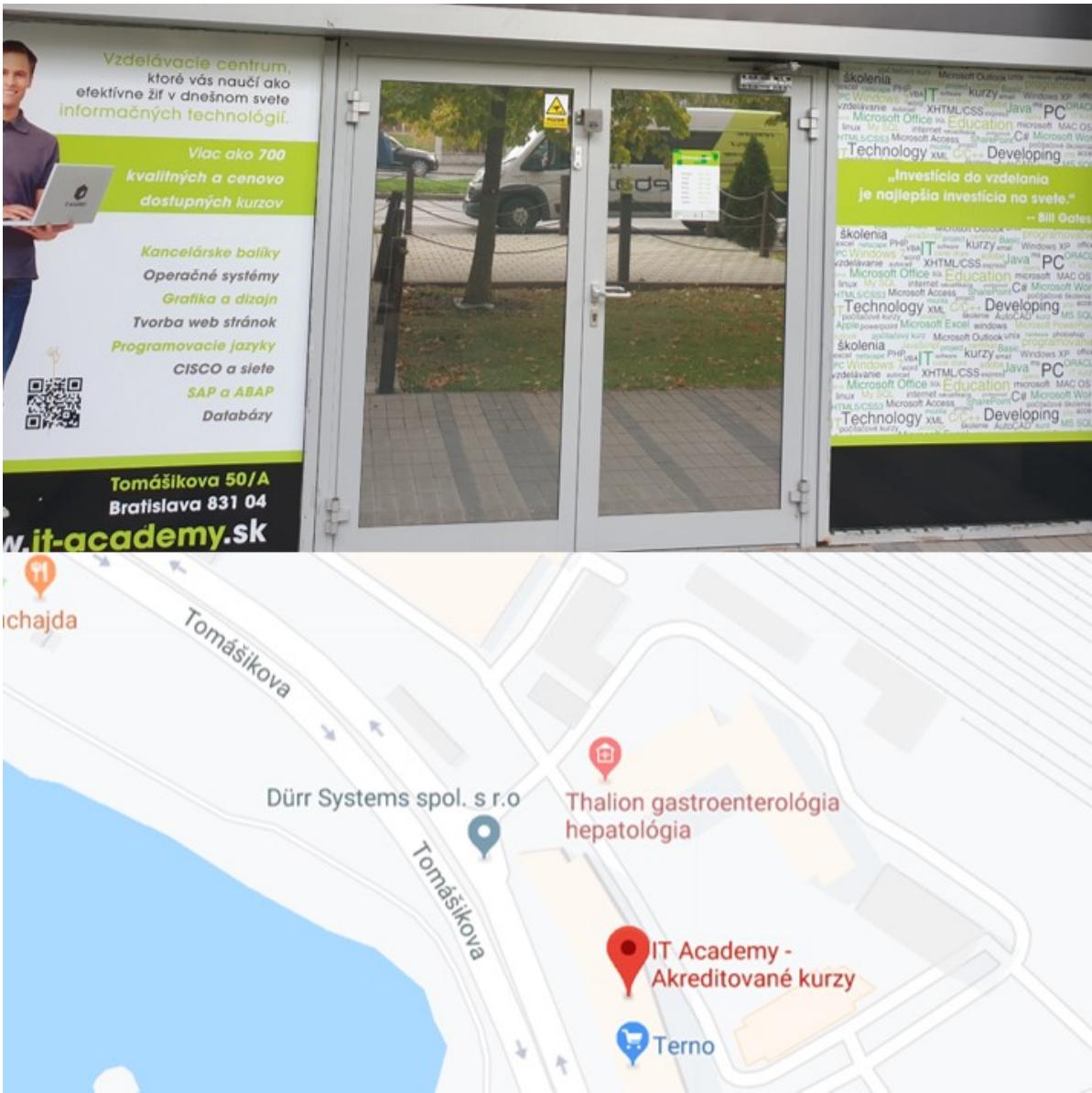
**WEB:** [www.it-academy.sk](http://www.it-academy.sk)



**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

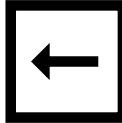
#Vývoj

#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) 

# 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€~~

299.99€

Prístup pre Teba do všetkých  
aktuálnych aj pripravovaných  
online kurzov

12 mesačná platnosť

Kúpiť teraz

## Zadarmo

- A. Ďalšie kurzy NSPC
- B. Grow with Google

1. NESTRAŤ PRÁCU (Zamestnaní, SZČO,  
na materskej)
2. KOMPAS/REPAS (Nezamestnaní)
3. YouTube Kanál (IT Academy)

## Platené

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