

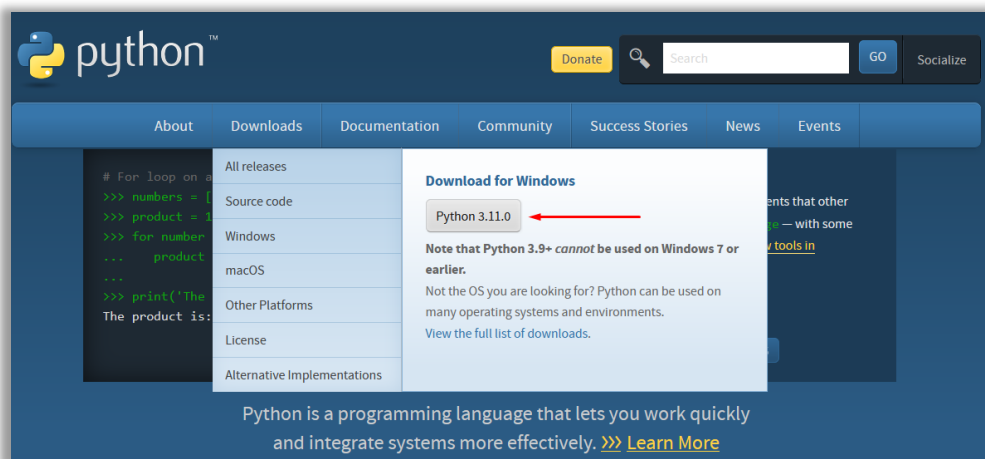
## Prvé kroky s programovacím jazykom Python

Najťažšie pri práci s akoukoľvek technológiou je začať. V rámci tohto krátkeho blogu bude ukázané, ako začať pracovať s programovacím jazykom Python od úplného začiatku (inštalácie) až po vytvorenie jednoduchej desktopovej aplikácie, ktorá bude schopná vypočítať dĺžku prepony pravouhlého trojuholníka na základe zadania dĺžok jeho odvesien.

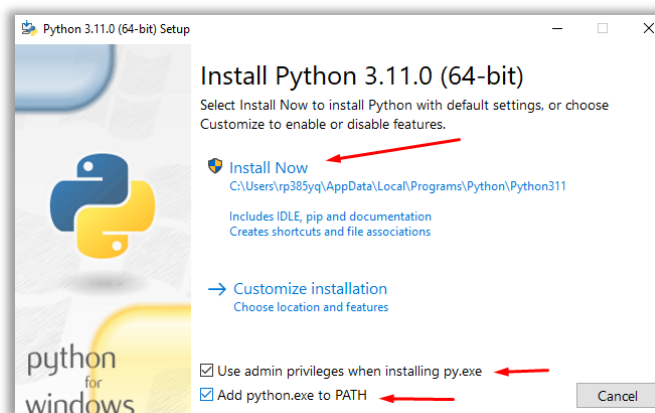
### Inštalácia jazyka Python

Návod inštalácie bude ukázaný pre používateľov OS Windows:

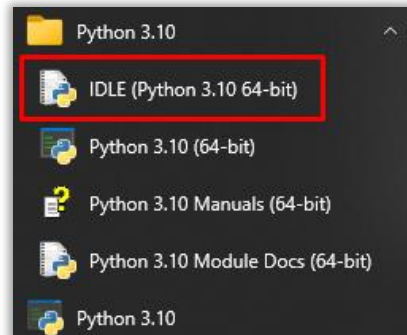
1. Využitím webového prehliadača vyhľadajte stránku: <https://www.python.org>
2. V sekcii **Download** zvolte stiahnutie inštalačného súboru najnovšej verzie jazyka Python (3.11.0), čo znázorňuje nasledujúci Obr.:



3. Po stiahnutí inštalačného súboru na tento súbor kliknite, zaškrtnite možnosť inštalácie aj napriek tomu, že je možné Python inštalovať aj z Store a následne v inštalačnom wizarde zaškrtnite políčko pre Add Python to PATH a kliknite na tlačidlo Install Now, čo znázorňuje nasledujúci Obr.:



4. V tejto chvíli stačí prejsť štandardným postupom – NEXT – NEXT - ... až po ukončenie inštalácie.
5. Po úspešnej inštalácii je možné v ponuke **Štart** vyhľadať nainštalované Python aplikácie, pričom pre nás bude dôležité vyhľadať aplikáciu Python IDLE, čo znázorňuje nasledujúci Obr. (Názov aplikácií sa bude líšiť podľa nainštalovanej verzie Pythonu):



### Prvý program – Hello World

Po spustení aplikácie Python IDLE sa spustí základný Python Interpreter, do ktorého je možné zadávať jednotlivé Python príkazy (zdrojový kód), ktoré sa začnú okamžite vykonávať, čo znázorňuje nasledujúci obrázok:

```
Python 3.10.6 (tags/v3.10.6:9c7b4bd, Aug 1 2022, 21:53:49) [MSC v.1932 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> x=3
>>> y=10
>>> print("Súčet dvoch čísel je: ", x+y)
Súčet dvoch čísel je: 13
```

Takto písať kód ale v prípade väčších programov nie je vhodné. Odporúčaný postup je vytvorenie prázdneho súboru, do ktorého sa napíše zdrojový kód a následne sa tento zdrojový kód spustí a vykoná. Pre vytvorenie nového súboru stačí zvoliť možnosť (v okne Python IDLE) **File -> New File**, čo automatický otvorí jednoduchý textový editor, kde je možné začať písať zdrojový kód. Po napísaní zdrojového kódu je potrebné vytvorený súbor uložiť (**File -> Save As**). Po uložení stačí vytvorený skript spustiť stlačením tlačidla **Run -> Run Module**. Výstup vykonania programu bude zobrazený v Python IDLE. Nasledujúci obrázok znázorňuje zdrojový kód pre vypísanie textu **Hello World**.

```
#vypísanie textu Hello World
print("Hello World")
```

Blahoželám, práve ste si vytvorili Váš prvý Python kód.

## Vytvorenie jednoduchovej desktopovej aplikácie

Pre zvýšenie záujmu o programovanie je vhodné vyvíjať aplikácie, ktoré budú mať aj vlastné používateľské rozhranie. V tejto sekcii bude ukázané, ako je možné vytvoriť jednoduchú desktopovú aplikáciu pre výpočet dĺžky prepony pravouhlého trojuholníka. Pre vytvorenie takejto aplikácie potrebujeme vykonať niekoľko jednoduchých krokov, ktoré sú vysvetlené v nasledujúcom postupe:

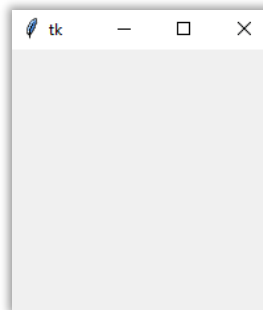
1. Pre vytvorenie okna aplikácie potrebujeme vytvoriť objekt, ktorý toto okno bude reprezentovať. Existuje viacero knižníc, no v rámci tohto manuálu bude ukázaná práca s knižnicou Tkinter. Nasledujúci Obr. znázorňuje základný zdrojový kód pre importovanie knižnice, vytvorenie objektu pre okno a otvorenie okna:

```
#importovanie knižnice tkinter
from tkinter import *

#vytvorenie objektu pre reprezentáciu okna
window = Tk()

#spustenie slučky, aby sa okno nezavrelo
window.mainloop()
```

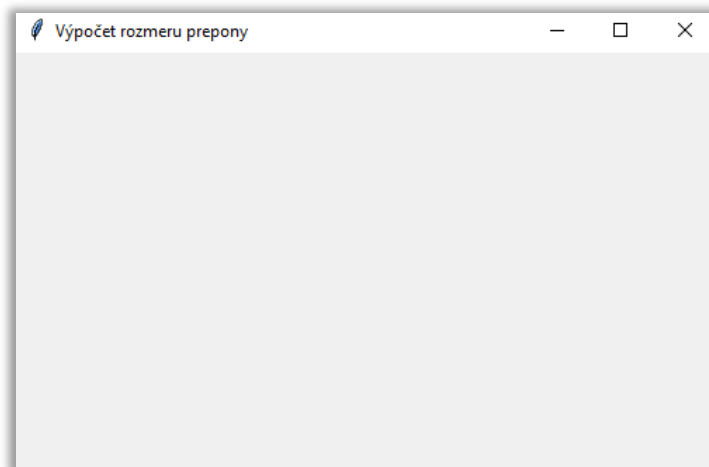
Po spustení tohto zdrojového kódu sa otvorí jednoduché prázdne okno vytvárajúcej desktopovej aplikácie, ktorej vzhľad je možné vidieť na nasledujúcom Obr.:



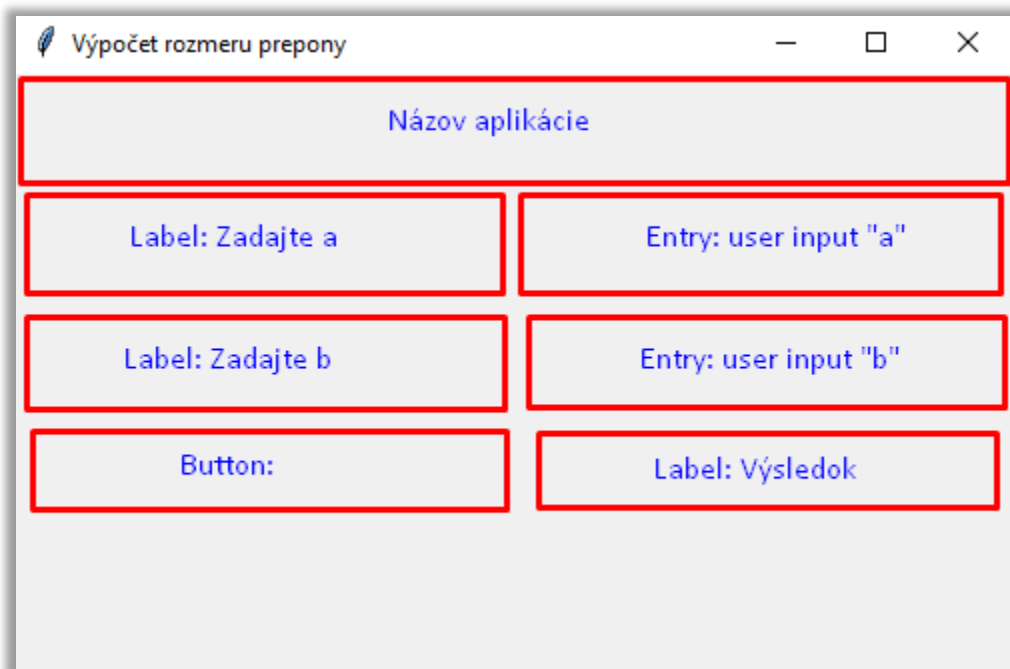
2. Ďalším krokom je nastavenie základných parametrov okna aplikácie. Nasledujúci zdrojový kód uvádza, ako je možné zmeniť parametre objektu pre okno (v kóde uložené v premennej **Window**), ktorými je možné nastaviť pomenovanie okna a jeho rozmery:

```
#nastavenie atribútov okna aplikácie
window.geometry("500x300")
window.title("Výpočet rozmeru prepony")
```

Uvedený zdrojový kód je potrebné vložiť za riadok, kde bol objekt pre okno vytvorený, no pred volanie funkcie **mainloop()**. Nasledujúci Obr. znázorňuje upravený vzhľad okna vytvárajúcej aplikácie:



3. V tejto chvíli môžeme začať vytvárať obsah používateľského rozhrania. Obsah rozhrania sa vytvára jednoducho pridávaním objektov reprezentujúcich jednotlivé elementy, medzi ktoré budú patriť: **Label** (element reprezentujúci textovú značku/text, ktorý chceme vypísať), **Entry** (element, cez ktorý bude možné získať vstup od používateľa) a **Button** (element, ktorý bude reprezentovať tlačidlo, po stlačení ktorého sa spustí nejaká obslužná funkcia).
4. Pred vkladáním elementov si je vhodné na papier zakresliť návrh používateľského rozhrania. Rozdeliť okno do mriežky (riadky a stĺpce) a následne vkladať elementy používateľského rozhrania na príslušný riadok a stĺpec. Ak chceme nejaký element centrovať – napr. vložiť názov aplikácie, tak je možné daný element vložiť cez viaceré stĺpce/riadky. Nasledujúci Obr. znázorňuje navrhnuté rozloženie vytvárajanej aplikácie:

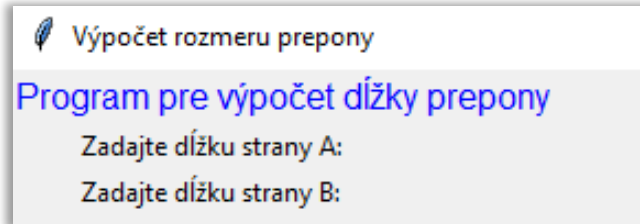


Ako je možné vidieť, tak používateľské rozhranie pre vytváranú aplikáciu bude vyžadovať 4 riadky a 2 stĺpce. Indexovanie je štandardné od nuly (zhora nadol, zľava doprava).

5. Najskôr vložíme elementy pre vypísanie textu (Label). Pri vytváraní týchto elementov stačí definovať okno, do ktorého budú vložené, ich obsah (samotný text) a ich umiestnenie. Z pohľadu textu je tiež možné nastaviť font, veľkosť, farbu ... Nasledujúci Obr. znázorňuje vloženie značiek pre Názov aplikácie a vyžiadanie zadania dĺžky odvesien (strana A a strana B):

```
#vloženie elementov pre vypísanie textu
Label(window, text="Program pre výpočet dĺžky prepony", font=12, fg="blue").grid(columnspan=2, row=0)
Label(window, text="Zadajte dĺžku strany A: ").grid(column=0, row=1)
Label(window, text="Zadajte dĺžku strany B: ").grid(column=0, row=2)
```

Očakávaný výstup aplikácie po vložení značiek je zobrazený na nasledujúcom Obr.:

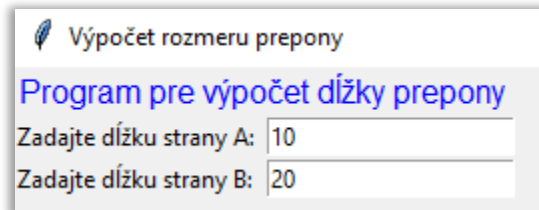


6. Ďalším krokom je vloženie elementu pre používateľský vstup (element **Entry**). Tento vstup je potrebné previazať s nejakou premennou, do ktorej bude ukladaný obsah používateľského vstupu. Nasledujúci Obr. znázorňuje zdrojový kód vloženia elementov Entry a previazanie s premennými pre uloženie získaného vstupu:

```
#vloženie elementov pre získanie používateľského vstupu
strana_A = StringVar(window, "")
Entry(window, textvariable=strana_A).grid(column=1, row=1)

strana_B = StringVar(window, "")
Entry(window, textvariable=strana_B).grid(column=1, row=2)
```

Očakávané rozhranie upravenej aplikácie je znázornené na nasledujúcom Obr.:

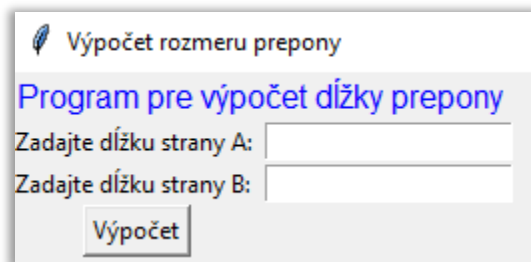


7. Posledným elementom, ktorý je potrebné pridať do vytváratej aplikácie je tlačidlo, po stlačení ktorého sa spustí obslužná funkcia, ktorá spracuje používateľský vstup a vypíše do okna aplikácie výsledok výpočtu. Nasledujúci Obr. znázorňuje kód pre vloženie tlačidla a vytvorenie obslužnej funkcie s názvom compute\_C:

```
#vytvorenie obslužnej funkcie
def compute_C():
    pass

#vloženie tlačidla
Button(window, text="Výpočet", command=compute_C).grid(column=0, row=3)
```

Vzhľad programu po úprave zdrojového kódu je znázornený na nasledujúcom Obr.:



V tejto chvíli, po stlačení tlačidla sa zatiaľ nevykoná nič. Obslužná funkcia je prázdna.

8. Obslužná funkcia má zobrať vstup od používateľa a vykonať matematický výpočet

$$c = \sqrt{a^2 + b^2}$$

Pozn.: na výpočet druhej odmocniny je možné použiť funkciu **sqrt()** a na výpočet druhej mocniny je možné použiť funkciu **pow()**, pričom prvý argument tejto funkcie je číslo, ktoré chceme umocňovať – v našom prípade to bude hodnota dĺžok odvesien. Druhým argumentom funkcie pow je hodnota exponentu – v našom prípade 2, keďže chceme vypočítať odvesnu na druhú.

Túto funkcionálnosť znázorňuje nasledujúci zdrojový kód:

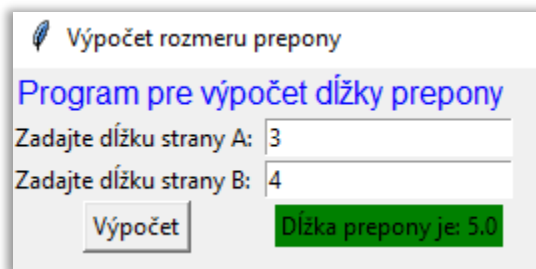
```
#importovanie knižnice pre prácu s matematickými operáciami
from math import sqrt

#vytvorenie obslužnej funkcie
def compute_C():
    #získanie vstupu a jeho prevod zo stringu na int
    a = int(strana_A.get())
    b = int(strana_B.get())

    #vypocet dĺžky prepony
    c = sqrt(pow(a,2) + pow(b,2))

    #vypísanie výsledku
    Label(window, text="Dĺžka prepony je: "+ str(c), bg="green").grid(column=1, row=3)
```

Nasledujúci Obr. znázorňuje finálnu verziu aplikácie, ktorá po zadaní vstupu a stlačení tlačidla vypočíta dĺžku prepony a vypíše túto informáciu do okna aplikácie:



9. Kompletný zdrojový kód vytvorenej aplikácie je uvedený na nasledujúcom Obr.:

```
#importovanie knižnice tkinter
from tkinter import *
#importovanie knižnice pre prácu s matematickými operáciami
from math import sqrt

#vytvorenie objektu pre reprezentáciu okna
window = Tk()

#nastavenie atribútov okna aplikácie
window.geometry("500x300")
window.title("Výpočet rozmeru prepony")

#vlozenie elementov pre vypísanie textu
Label(window, text="Program pre výpočet dĺžky prepony", font=12, fg="blue").grid(columnspan=2, row=0)
Label(window, text="Zadajte dĺžku strany A: ").grid(column=0, row=1)
Label(window, text="Zadajte dĺžku strany B: ").grid(column=0, row=2)

#vlozenie elementov pre získanie používateľského vstupu
strana_A = StringVar(window, "")
Entry(window, textvariable=strana_A).grid(column=1, row=1)

strana_B = StringVar(window, "")
Entry(window, textvariable=strana_B).grid(column=1, row=2)

#vytvorenie obslužnej funkcie
def compute_C():
    #získanie vstupu a jeho prevod zo stringu na int
    a = int(strana_A.get())
    b = int(strana_B.get())

    #vypocet dĺžky prepony
    c = sqrt(pow(a,2) + pow(b,2))

    #vypísanie výsledku
    Label(window, text="Dĺžka prepony je: "+ str(c), bg="green").grid(column=1, row=3)

#vlozenie tlačidla
Button(window, text="Výpočet", command=compute_C).grid(column=0, row=3)

#spustenie slučky, aby sa okno nezavrelo
window.mainloop()
```

## Záver

Ako je možné vidieť, tak tvorba jednoduchovej desktopovej aplikácie je priamočiara a pozostáva z mála elementárnych krokov. Vytvorenú aplikáciu je možné jednoduchými zmenami rozšíriť na výpočet ďalších typov úloh.