**Vyšší odborná škola a Střední průmyslová škola elektrotechnická Olomouc,**

**Božetěchova3**

**PRAKTICKÁ ZKOUŠKA Z ODBORNÝCH PŘEDMĚTŮ**

Tutoriál na WxPython

2020 Filip Plachý

Prohlašuji, že jsem praktickou zkoušku z odborných předmětů vypracoval samostatně a všechny prameny jsem uvedl v seznamu použité literatury.

……………………………  
 jméno a příjmení žáka

Chtěl bych vyslovit poděkování panu Ing. Markovi Nožkovi za odborné konzultace a poskytnuté informace.

……………………………  
 jméno a příjmení žáka

Prohlašuji, že nemám námitek proti půjčování nebo zveřejňování mé práce nebo její části se souhlasem školy.

……………………………  
 jméno a příjmení žáka

# Abstrakt

Úkolem diplomové práce je vytvořit sbírku tutoriálů popisující základy knihovny wxWidgets v programovacím jazyce Python. Samotná knihovna pythonu podporující wxWidgets se nazývá wxPython. Jedná se o alternativu k vytváření aplikací s uživatelským grafickým rozhraním.

Sbírka je určena pro studenty programování začínající s GUI, kteří již mají za sebou základní zkušenosti s pythonem. Součástí tutoriálů je úvod do objektově orientovaného programování, které se při tvorbě grafického rozhraní využívá. Po dokončení základních tutoriálů je součástí sbírky i sada názorných příkladů, řešící základní jednoduché aplikace.

# Obsah

Obsah se vkládá až po napsání celé práce pomocí menu Odkazy / Obsah. Při jakékoliv změně nadpisů v práci je třeba obsah aktualizovat – pravé tlačítko / Aktualizovat pole / celá tabulka.

[Obsah 5](#_Toc525402557)

[Úvod 6](#_Toc525402558)

[1. Teoretická část 7](#_Toc525402559)

[1.1 Python 7](#_Toc525402560)

[1.1.1 Úvod do jazyka Python 3](#_Toc525402564)

[1.1.2 Historie 3](#_Toc525402564)

[1.2 wxPython 3](#_Toc525402561)

[1.2.1 Historie wxPythonu 3](#_Toc525402564)

[1.2 wxWidgets 3](#_Toc525402561)

[1.2.1 Historie WxWidgets 3](#_Toc525402564)

[1.3 Jupyter notebook 3](#_Toc525402562)

[1.3.1 Úvod 3](#_Toc525402564)

[1.3.2 Historie 3](#_Toc525402564)

[1.3.3 Projekt Jupyter 3](#_Toc525402564)

[1.3.4 IPython 3](#_Toc525402564)

[1.4 Objektově orientované programování 3](#_Toc525402563)

[1.4.1 Úvod 3](#_Toc525402564)

[1.4.2 Funkcionalita v Pythonu 3](#_Toc525402565)

[2. Pokyny pro psaní textu 3](#_Toc525402566)

[2.1 Nejčastější nedostatky 3](#_Toc525402567)

[2.2 Druhá podkapitola 3](#_Toc525402568)

[2.3 Třetí podkapitola 3](#_Toc525402569)

[Závěr 3](#_Toc525402570)

[Seznam použité literatury 3](#_Toc525402571)

[Seznam obrázků a tabulek 3](#_Toc525402572)

# Úvod

Práce se rozděluje na 2 části. Teoretická a praktická. Teoretická obsahuje dopodrobna popsané nástroje a funkce, které byly při tvorbě tutoriálu využity. V praktické části si rozebereme samotný tutoriál knihovny wxPython. Tutoriál je určen pro studenty s již základními zkušenosti v programovacím jazyce python. Při práci s wxPython je potřeba mít i znalost Objektově orientovaného programování (OOP), tudíž součástí sady tutoriálů je i lekce do OOP. Cílem práce je vytvořit představu a znalostní základ, se kterým student zvládne vytvářet aplikace s grafickým uživatelským rozhraním.

# Teoretická část

Teoretická část obsahuje základní informace všech programů a nástrojů, které byly při práci využity.

## Python

Cílem této podkapitoly je obecný úvod do programovacího jazyka Python.

### Úvod do jazyka python

Python je univerzální programovací jazyk[[1]](#footnote-2).

Jazyk paří mezi tzv. "interpretované jazyky". To znamená, že napsaný zdrojový kód v jazyce Python je převeden (interpretován) pomocí programu (interpreter/tlumočník) do jazyka, se kterým pracuje počítač. Samotný tlumočník pro Python je k dispozici na všech operačních systémech přímo na stránkách [www.python.org](https://www.python.org).

Pro práci s pythonem slouží tzv. „IDE“ (Integrated Development Environment) , česky: „editor“, který pomáhá uživateli s formátováním kódu tak, aby byl nejen přehledný pro uživatele, ale také aby se dal převést do formátu proveditelného počítačem. Nejznámějšími editory jsou: Pycharm, Pydev, Visual Studio Code, VIM, Atom/Atom-IDE, IDLE, Spyder...

Python byl navržen tak, aby se jeho veškeré funkce nacházely přímo v jádru programovacího jazyka.

Má jednodušší a méně přeplněnou syntaxi a gramatiku, např. díky využití mezer. Další jeho výhodou je rozšiřitelnost o další knihovny/moduly.

### Historie

Programovací jazyk Python byl navrhnut mezi roky 1990-1991 holaňdanem Guidem van Rossumem[[2]](#footnote-3) v národním výzkumném institutu pro matiku a informatiku (CWI) v Amsterdamu[[3]](#footnote-4). Samotné pojmenování pythonu nemá nic společného s druhem hada, neboť Van Rossum pojmenoval python po televizním pořadu anglické BBC Monty Pythonův létající kruh[[4]](#footnote-5).

První verzi kódu zveřejnil Van Rossum v únoru roku 1991 (verze 0.9.0). Již v této fázi vývoje bylo možné v pythonu pracovat s třídy a dědičnostmi. K dispozici již také byli i základní datové typy jako string, list či dict. V počátečním vydání Python již obsahoval modulový systém[[5]](#footnote-6).

Python dosáhl verze 1 v lednu 1994. Novými funkcemi byla lambda[[6]](#footnote-7) a práce s mapováním[[7]](#footnote-8), filtrováním[[8]](#footnote-9) a redukování[[9]](#footnote-10) vyšších funkcí[[10]](#footnote-11) (Funkce vyšších řádů berou jednu nebo více funkcí nebo atributů a jako výsledek vrací jednu funkci). Později Van Rossum opouští CWI a pokračuje na pythonu ve CNRI[[11]](#footnote-12) (Korporace pro národní výzkumné iniciativy) ve Virginii. Další důležitou verzí je 1.4, kde se objevili pojmenované parametry[[12]](#footnote-13) nebo podpora komplexních čísel.

Během Van Rossumova pobytu ve CNRI se institut snažil o zpřístupnění možnosti programování pro veřejnost se základním vzdělání. Tohle mělo později za následek menší "šachování" mezi licencemi pythonu 1.6, které měli za cíl získání licence svobodného softwaru[[13]](#footnote-14) (Free-software licence). Další verze 1.6.1 sice neobsahovala žádné důležité funkce, ale měla změněnou licenci na GPL/GNU[[14]](#footnote-15) (General Public License neboli bezplatná softwarová licence).

Verze 2.0 byla vydána v říjnu 2000, která představila seznamy[[15]](#footnote-16).

2.2 představila sjednocení typů Pythonu (typů napsaných v C) a tříd (typů napsaných v Pythonu) do jedné hierarchie.

2.5 představil prohlášení with[[16]](#footnote-17) umožňující otevření a zavření souboru a další funkce.

Python 2.6 byl vydán, aby se shodoval se souběžným vývojem Pythonu 3.0 a varoval hlavně o funkcích, které jsou ve verzi 3.0 odstraněny.

Podobně byl vydán i 2.7, který se shodoval s 3.1. Python 2.7 byl posledním vydáním ve druhé sérii. V listopadu 2014 byl oznámen konec podpory 2.7 do roku 2020. Uživatelé byli vyzváni, aby postupně přešli na Python 3.0.

1. ledna 2020 byl "zmražen"[[17]](#footnote-18) kód Pythonu 2.7. Konečné vydání, 2.7.18, došlo 20. dubna 2020 a zahrnovalo opravy kritických chyb a blokace vydání.

Verze 3.0 přišla 3. prosince 2008. 3.0 byla navržena, aby napravila základní konstrukční chyby jazyka. Tyhle změny avšak znemožnili zpětnou kompabilitu se staršími verzemi (tudíž došlo k samotnému oddělení verzí z 2.x na 3.0). Hlavním motem změn bylo odstranění nadbytečných, duplicitních konstrukcí a modulů. Vznikl nástroj tzv. 2to3, který dokázal přepsat automaticky Python 2 do nové verze, avšak nástroj nefunguje na 100% a některé aspekty nedokáže převést.

Hlavními úpravami byly:

* Změna print, aby se jednalo o vestavěnou funkci. V Pythonu 2.6 a 2.7 print() je k dispozici jako vestavěná funkce, avšak je maskovaná syntaxí příkazu, který lze deaktivovat zadáním „from \_\_future import print\_functionv“ v hlavičce souboru
* Odebrání inputu ve verzi 2. Z verze poté byl přebrat raw\_input, který byl přejmenován na klasický input.
* Přidání podpory pro anotace jednotlivých funkcí (Když v kódu byla nepoužívaná funkce se špatnou syntaxí, tak se program ve 2.x nespustil)
* Sjednocení str / unicode typů
* Odebrání funkcí zpětné kompatibility, včetně tříd starého stylu, výjimek řetězců a implicitních relativních importů
* Změna funkce celočíselného dělení. (ve 2.0 5 / 2 = 2, nyní 5 / 2 = 2.5. ve 3.0 vznikla náhradní syntaxe 5 // 2 = 2)

## wxPython

WxPython[[18]](#footnote-19) je obal pro multiplatformní GUI[[19]](#footnote-20) (grafické uživatelské rozhraní) aplikační programovací rozhraní wxWidgets (napsán v C++) pro programovací jazyk Python. Jedná se o otevřený (veřejný kód) rozšiřující modul Pythonu. Oficiální stránky wxPythonu: https://www.wxpython.org/

### Historie wxPythonu

WxPython byl vytvořen Robinem Dunnem[[20]](#footnote-21), když potřeboval GUI k operačnímu systému HP-UX a Windows verze 3.1 (1992-1995). Při hodnocení komečních řešení narazil na vazby Pythonu se sadou nástrojů wxWidgets.

První verze verze byly vytvořeny ručně. Avšak brzy se kódová základna velmi obtížně udržovala synchronizovaná s novými verzemi wxWidgets. Pozdější verze byly vytvořeny pomocí SWIG [[21]](#footnote-22)(nástroj pro propojení knihoven v C se skriptovacími jazyky), který výrazně snížil množství práce na aktualizaci.

První "moderní" verze 0.3 byla oznámena v roce 1998.

Aktuální verze s podtitulem „Excaping the Quarantine“ je 4.1.0.

Celý vývoj wxPython nalezneme na oficiálních stránkách:

<https://wxpython.org/pages/changes/index.html>

## wxWidgets

Samotná sada nástrojů wxWidgets je knihovna nástrojů pro vytváření graficky uživatelských rozhraní napříč všemi platformami. WxWidgets umožňuje kódu GUI kompilovat a spouštět na několika počítačových platformách s žádnými nebo minimálními změnami kódu.

Jedná se o bezplatný a otevřený software distribuovaný za podmínek licence WxWidgets[[22]](#footnote-23), která je obdobná GPL/GNU u Pythonu. Stránky wxWidgets: <https://www.wxwidgets.org/>

### Historie wxWidgets

WxWidgets (původně wxWindows) zahájil v roce 1992 Julian Smart[[23]](#footnote-24) z Edinburské Univerzity[[24]](#footnote-25). V roce 2004 došlo k přejmenování wxWindows v důsledku požadavků společnosti Microsoft pro distribuci v UK.

Hlavní verze byla vydána 6. ledna 2004. Samotná verze 3.0 byla vydána 11. listopadu 2013.

# Pokyny pro psaní textu

* Další kapitola začíná vždy na nové stránce.
* Mezera mezi slovy se dělá jen jedna. Chybně je to takto .
* Před tečkou na konci věty se mezera nedělá. Mezi větami se dělá jedna mezera.
* Horní a dolní indexy je nutné psát takto m3, H2O. Používáme k tomu přímo funkci indexy (ikonky X2, X2).
* Psaní číslovek. Napíšete-li 3. (tj. s tečkou) znamená to třetí. Ale napíšete-li 3 (bez tečky) znamená to tři.
* Procento má jiný význam píše-li se s mezerou nebo bez mezery. Napíšeme-li 12 %, znamená to dvanáct procent. Napíšeme-li 12%, potom to je dvanáctiprocentní.
* Pomlčka se odděluje z obou stran mezerami, např. zápas Sparta – Slavia.
* Závorky a uvozovky přiléhají k začátku a konci textu bez mezer, od ostatního textu se oddělí mezerami (Je to „maskot“ družstva.).
* Značky jednoduchých měřících jednotek se píší s mezerou mezi hodnotou a jednotkou (tj. 10 mm, +6 V, 8,77 kg, 7 x 5 m, 10:00 h, 12 min, -15 ºC, 300 kW, 20 VA). Pokud mezeru neuděláte, znamenalo by to číslovku řadovou tedy, 10mm znamená desetimilimetrový.
* Mezní úchylky, 5 ±1, 12 ±0,05, pokud jsou uvedeny měrné jednotky, píšou se s číslem rozměru do závorky: (44 ±0,1) mm, (-13 ±2) ºC.

## Nejčastější nedostatky

* V práci jsou gramatické chyby (interpunkce, skloňování, časování aj.) Je možno použít internetovou jazykovou příručku http://prirucka.ujc.cas.cz
* V práci jsou neopravené chyby (např. překlepy, dělení slov aj.) Využijte kontrolu pravopisu menu Revize / Pravopis a gramatiky.
* Je zřejmé, že autor před odevzdáním práci nekontroloval.
* Nevhodná úprava textů, obrázků a tabulek (např. nezarovnaný text, různé typ písma, nejednotné okraje, nevhodné zvýrazňování aj.)
* Jsou použity různé druhy písma. To svědčí jen o tom, že autor text jen kopíroval.
* Obrázky, tabulky, vzorce aj. nejsou očíslovány a označeny názvem.
* Věty jsou buď příliš dlouhé, nebo se naopak používají jen holé věty.
* Používají se hovorové a slangové výrazy atp.
* Nejsou jasné způsoby výpočtu výsledků a údajů.

## Druhá podkapitola

Takto pokračujeme dále až do konce. Opět styl odstavce Nadpis 2.

## Třetí podkapitola

# Závěr

V této kapitole práce se autor věnuje zhodnocení celé práce. Je vhodné hodnotit práci podle bodů zadání. Uveďte, co jste měli udělat, jak jste to udělali a s jakými výsledky, poznatky, úspěchy či neúspěchy, vyjádřit se k splnění cíle práce. Je nutné zaujmout konkrétní stanovisko k jednotlivým výstupům práce.

Závěr je psán v první osobě jednotného čísla, v minulém čase. Rozsah závěru je jedna až dvě strany.

Nestačí napsat, že se „něco vytvořilo“, ale konkrétně s jakými výsledky, nedokonalosti je potřeba zdůvodnit, uvést možné nápravy, náměty na další práci, výhledy do budoucna atd.

# Seznam použité literatury

[1] Wikipedie:

<https://cs.wikipedia.org/wiki/Python>

<https://cs.wikipedia.org/wiki/Programovac%C3%AD_jazyk>

https://en.wikipedia.org/wiki/Guido\_van\_Rossum

# Seznam obrázků a tabulek

Zde se vkládá jí seznamy obrázků a tabulek. Vložíte to v menu Odkazy / Vložit seznam obrázků, kde vyberete v Obecných / Popisek titulu Obrázek č. a potom Tabulka č.

[Obrázek č. 1 Elektromagnetický ventil 3](#_Toc509907309)

[Obrázek č. 2 Graf spotřeby plynu v letech 2010, 2011 a 2012 3](#_Toc509907310)

[Obrázek č. 3 Nevhodná fotografie – nevhodné pozadí, nečitelné popisky. 3](#_Toc509907311)

[Obrázek č. 4 Vhodně upravená fotografie 3](#_Toc509907312)

[Tabulka č. 1: Měření zesílení operačního zesilovače 3](#_Toc450839160)

# Přílohy

V přílohách se uvádějí složitější obrázky, grafy, schémata zapojení, schémata desek plošných spojů, výpisy programů, fotografie a podobně, které není vhodné dávat do základního textu práce.

Každá příloha je číslovaná a pojmenovaná stejně jako obrázky, včetně zdrojů.

Příloha č. 1: Poster k maturitní práci.

Zde bude **povinně vložen obrázek vytvořeného posteru** ve formátu A4, barevný, orientovaný na výšku popř. i šířku.

1. https://cs.wikipedia.org/wiki/Programovac%C3%AD\_jazyk [↑](#footnote-ref-2)
2. https://en.wikipedia.org/wiki/Guido\_van\_Rossum [↑](#footnote-ref-3)
3. https://en.wikipedia.org/wiki/Centrum\_Wiskunde\_%26\_Informatica [↑](#footnote-ref-4)
4. https://en.wikipedia.org/wiki/Monty\_Python%27s\_Flying\_Circus [↑](#footnote-ref-5)
5. https://en.wikipedia.org/wiki/Modular\_programming [↑](#footnote-ref-6)
6. https://www.w3schools.com/python/python\_lambda.asp [↑](#footnote-ref-7)
7. https://en.wikipedia.org/wiki/Map\_%28higher-order\_function%29 [↑](#footnote-ref-8)
8. https://en.wikipedia.org/wiki/Filter\_%28higher-order\_function%29 [↑](#footnote-ref-9)
9. https://en.wikipedia.org/wiki/Fold\_%28higher-order\_function%29 [↑](#footnote-ref-10)
10. https://en.wikipedia.org/wiki/Higher-order\_function [↑](#footnote-ref-11)
11. https://en.wikipedia.org/wiki/Corporation\_for\_National\_Research\_Initiatives [↑](#footnote-ref-12)
12. https://en.wikipedia.org/wiki/Named\_parameter [↑](#footnote-ref-13)
13. https://cs.qaz.wiki/wiki/Free-software\_license [↑](#footnote-ref-14)
14. https://en.wikipedia.org/wiki/GNU\_General\_Public\_License [↑](#footnote-ref-15)
15. https://en.wikipedia.org/wiki/List\_comprehension [↑](#footnote-ref-16)
16. https://en.wikipedia.org/wiki/Python\_syntax\_and\_semantics#With\_statements [↑](#footnote-ref-17)
17. https://en.wikipedia.org/wiki/Freeze\_%28software\_engineering%29 [↑](#footnote-ref-18)
18. https://en.wikipedia.org/wiki/WxPython [↑](#footnote-ref-19)
19. https://en.wikipedia.org/wiki/Graphical\_user\_interface [↑](#footnote-ref-20)
20. https://wiki.wxpython.org/RobinDunn [↑](#footnote-ref-21)
21. https://cs.qaz.wiki/wiki/SWIG [↑](#footnote-ref-22)
22. https://en.wikipedia.org/wiki/WxWidgets#License [↑](#footnote-ref-23)
23. http://www.anthemion.co.uk/julian.htm [↑](#footnote-ref-24)
24. https://en.wikipedia.org/wiki/University\_of\_Edinburgh [↑](#footnote-ref-25)