

MODULY

Jmenný prostor (namespace)

1. V Pythonu je všechno objekt
2. Každý objekt má svůj jmenný prostor

Funkce `dir` vypíše obsah jmenného prostoru

```
dir()
```

```
a=5
```

```
dir()
```

```
dir(a)
```

```
import math
```

```
dir(math)
```

Jak použít („importovat“) modul

1.způsob:

```
import math
a = math.sqrt( 2 )
```

```
import math as mm
a = mm.sqrt( 2 )
```

2.způsob:

```
from math import sqrt
a = sqrt( 2 )
```

3.způsob:

```
from math import * # => NE!  
pi  
pi = sqrt( 2 )  
pi  
math.pi
```

Jak vytvořit modul

= jen uložit zdrojový kód, jako jakýkoliv jiný

Příkazy uvnitř modulu

Provádí se jen při prvním příkazu import

Proměnná `__name__`

V hlavním programu má hodnotu `'__main__'`,
jinak jméno modulu.

```
if __name__ == '__main__':
```

Proč vytvářet moduly

1. rozdělení programu na části
2. oddělení kódu, který bychom mohli použít i v jiných projektech

Světlá strana

The Python Standard Library

<https://docs.python.org/3/library/>

- součást instalace Pythonu pro Windows
- ...v Linuxu se doinstalovává obvyklými prostředky
- všestranná knihovna
- moduly napsané v Pythonu i v jazyku C

Temná strana

...později.

Zajímavé moduly

`datetime` — Basic date and time types

`math` — Mathematical functions

`random` — Generate pseudo-random numbers

`os.path` — Common pathname manipulations

= už jsme viděli, když jsme zjišťovali existenci souboru

`turtle` — Turtle graphics

`urllib.request` — Extensible library for opening URLs

`html.parser` — Simple HTML and XHTML parser

`tkinter` — Python interface to Tcl/Tk

Mnoho dalších (ne-standardních) doinstalovatelných

`matplotlib`

Balíček

Více souborů/modulů, adresářová struktura.

Soubor `__init__.py` v každém (pod)adresáři.

`__init__.py`

- může být prázdný
- příkaz

```
__all__ = ["aaa", "bbb", "ccc"]  
= seznam modulů pro příkaz  
from balik import *
```

V Programování 1 NE.

I když přece jen...

- Balíčky jsou užitečné
- Můžeme využívat toho, co už někdo naprogramoval
- Nemusíme vynalézat kolo
- The Python Package Index (PyPI) <https://pypi.org/>
204,421 projects 1,536,788 releases 2,288,324 files
- Můžeme sami přispět a ostatní mohou využívat naše balíčky

Jenomže (Temná strana)

- (Cizí) balíčky mohou vyžadovat další (cizí) balíčky
- Balíčky se vyvíjejí a mohou mít různé verze (jako všechno)
- Různé verze balíčků nemusí být zpětně kompatibilní
- = konec klidného spaní. viz "Dependency hell"

Zpětná kompatibilita

Dlouhá čísla

když nám nestačí standardní číselné typy...

Krok 1: Návrh reprezentace

- a) nezáporná x i záporná
 - a2) se znaménkem x doplněk
- b) celá x desetinná
 - b2) pevná x pohyblivá řádová čárka
- c) pole x spojový seznam
- d) odpředu x odzadu
- e) obsah jednoho prvku
- f) délka
 - f1) kolik míst navíc
 - spočítat (a ještě zkusit)