



Takács Edit

edit.takacs@gmail.com

GitHub:

https://github.com/gtakacse/CS_class_scripts



Emberi Erőforrások
Minisztériuma

AZ EMBERI
ERŐFORRÁSOK
MINISZTERIUMA ÚJ
NEMZETI KIVÁLÓSÁG
PROGRAMJÁNAK
TÁMOGATÁSÁVAL
KÉSZÜLT



Programozás nyelvészeknek (2. óra)

Ismerkedés a Spyder keretprogrammal és a
Pythonnal

Python objektumok

+ Spyder



The screenshot shows the Spyder Python IDE interface. The main window is titled 'Spyder (Python 3.5)'. The menu bar includes File, Edit, Search, Source, Run, Debug, Consoles, Tools, View, and Help. The toolbar contains icons for file operations, running, and debugging. The editor pane shows a Python script named 'temp.py' with the following code:

```
1 # -*- coding: utf-8 -*-
2 """
3 Spyder Editor
4 This is a temporary script file
5 """
6
7
8 def hello():
9     print('hello, world')
10
11 hello()
```

The interface includes several panels and callout boxes:

- Working directory helye**: Points to the address bar showing 'C:\Users\ASUS\Documents'.
- Gombok script futtatásához**: Points to the 'Run' button in the toolbar.
- Script editor**: Points to the main code editor area.
- Python Shellben futó folyamat leállítása**: Points to the 'Stop' button in the IPython console toolbar.
- Py. Shell újraindítása**: Points to the 'Restart' button in the IPython console toolbar.
- Python Shell**: Points to the IPython console output area.

The IPython console shows the following output:

```
Python 3.5.2 [Anaconda 4.1.1 (64-bit)] (default, Jul 5 2016, 11:41:13) [MSC v.1900 64 bit (AMD64)]
Type "copyright", "credits" or "license()" for more information.

IPython 4.2.0 -- An enhanced Interactive Python.
? -> Introduction and overview of IPython's features.
%quickref -> Quick reference.
help -> Python's own help system.
object? -> Details about 'object', use 'object??' for extra details.
%gui? -> A brief reference about the graphical user interface.

In [1]:
```

The status bar at the bottom shows: Permissions: RW, End-of-lines: CRLF, Encoding: UTF-8, Line: 11, Column: 1, Memory: 72 %.



Egyszerű Python objektumok



- Integer (egész számok)
- Float (lebegőpontos számok)
- Változók létrehozása
- String (karakter sor)
- Mi az a bug?
- Boolean (Igazságérték)
- None (érték hiánya)

+ Integer (Egész szám)



- Például

- 7
- 1
- -2
- 0

- `type(2)`

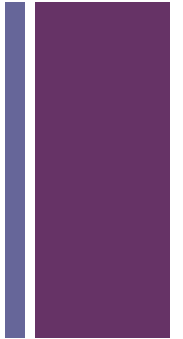


Float (Lebegőpontos szám)



- 3.14159265
- Vigyázz a szám nem egész részét **PONT** jelöli nem pedig vessző!
- CSAK közelíti a valós számokat:
 - 10/3
 - `print('%0.50f' % (10/3))`
- egy végtelen hosszú számot nem lehet pontosan tárolni a számítógép véges memóriájában

+ Python mint számológép



- Egyszerű aritmetikai operátorok (+, -, /, *)
- `2 + 2` # kifejezés, amelyet Python képes értékelni
- Egyéb operátorok
 - `**` # hatvány, 2^2 Pythonban `2**2`
 - `//` # int osztás ($9/2$) [vigyázz Python 2-ben nem így működik!!]
 - `%` # maradék visszaadás ($7\%5$)
- Zárójel () ugyanúgy működnek, mint a matekban. A számítások sorrendjét adja meg
- $(2 + 3) * 5 \neq 2 + 3 * 5$



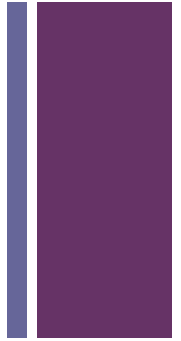
Feladat - Mi lesz az eredménye a következő kifejezéseknek?



1. $17 - 23 * 4$
2. Mi az első eredményének típusa?
3. $54/67$
4. Mi a 3. kifejezés típusa?
5. $2.3 + 7$
6. Mi az 5. eredményének típusa?
7. $(45 + 23.5) * 2 - 7$
8. $5 ** 3$
9. $6 // 5$
10. $\text{int}(6/5)$
11. $13 \% 9$

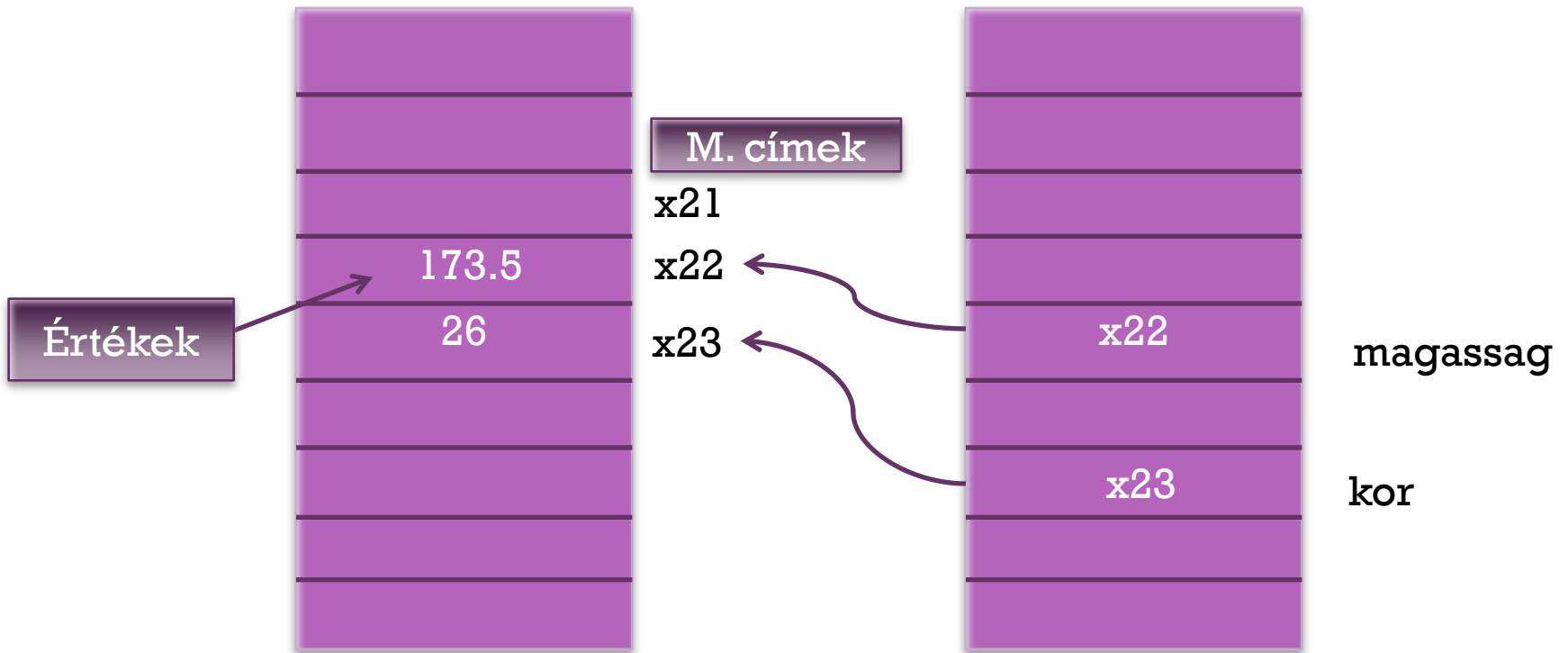


Változók elnevezése és tárolása

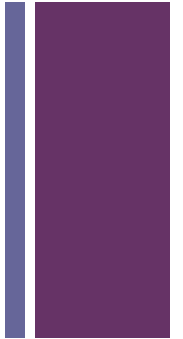


■ Memória

Változó = elnevezett hely a memóriában



+ Változók



- változó létrehozása a '=' *hozzárendelő operátorral*

- `x = 4`

- `hex(id(x))`

- felülírhatók

- `x = 7`

- `x = x + 4` # shorthand `x += 4`

- `y = 3.0`

- `type(y)`

+ String (karakter sor)



- szöveggént kezelt változó
- idézőjelek között szerepel
 - 'szöveg'
 - "és ez is szöveg"

+ String tulajdonságai

- van hosszuk >> len()

- indexelhetők

 - `x = 'hello!'`

 - `x[0]`

- / a menekülő karakter (pl. `"\"` >> a string `'`)

- speciális kombinációk: `\n` >> új sor, `\t` >> tab stb.

- `print()`

0	1	2	3	4	5	
H	E	L	L	O	!	
-6	-5	-4	-3	-2	-1	

**You know you're a
programmer when..**



you count 3 apples



Feladat - Mi a következő kifejezések értéke?



1. `x = 'Have a good day!'`

2. `x[:]`

3. `x[1:]`

4. `x[::-1]`

5. `x[18]`

6. `x[:18]`

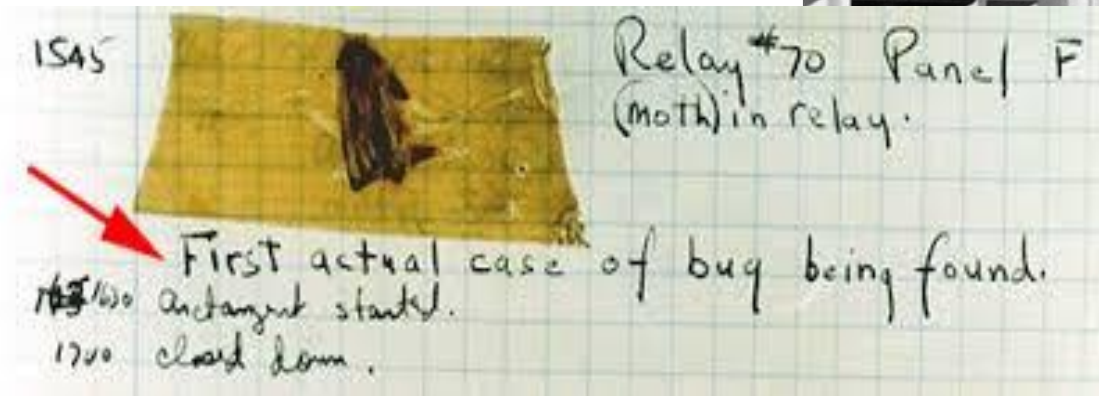
7. `x[::2]`



Mi a bug?



- Hiba a számítógépes programban, ami hatására, az nem úgy működik, mint elvárt.





Modern Bug



```
In [51]: s[18]
```

```
Traceback (most recent call last):
```

```
  File "<ipython-input-51-1c65de0552f6>", line 1, in <module>  
    s[18]
```

```
IndexError: string index out of range
```



Feladat - Írj Python kefejezést, ami a következőket hajtja végre!

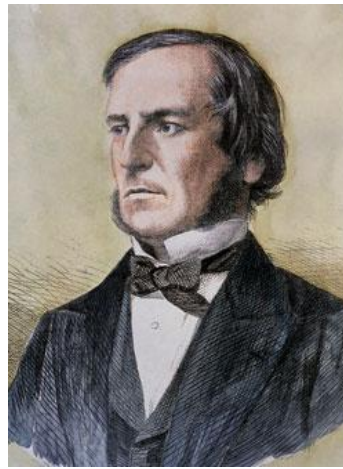


■ `x = 'hello'`

■ `city = 'Budapest'`

1. Add vissza az `X` változó utolsó két karakterét!
2. Add vissza a `CITY` változó első és utolsó karakterét!
3. Add vissza az `X` változó első, második és harmadik karakterét!
4. Add vissza a `CITY` változó 3-tól 6. karakterét.
5. Add vissza a `CITY` változó minden harmadik karakterét!
6. Hozzd létre az `X` és `CITY` változók felhasználásával a következő string-et: `'hello Budapest'`

+ Boolean



George
Boole

- Logikai művelet
- True/False (Igaz / Hamis)
- a kifejezés igazságértéke (Esik az eső? >> Igaz vagy Hamis)
- a feltételes utasításokban használt struktúra (if, while)

1. ha esik az eső:

2. vigyél esernyőt

```
>>> 3 < 2
```

```
False
```

- operátorok: <, >, ==, !=, >=, <=, and, or, not



Feladat - Milyen értéket adnak a következő kifejezések?



■ `x = hello`

■ `num = 5`

1. `len(hello) == num`

2. `num**2 >= 100`

3. `True == False`

4. `3 == 3.0`

5. `type(3) != type(3.0)`

6. `'b' <= 'a'`

7. `not(True) == False`

8. `len(x*2) == num + 4`

9. `type(False) == bool`

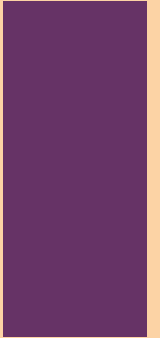
10. `type(x) != type('77')`

11. `x == num`

12. `x <= num`



Feladat - Írj olyan Python kifejezéseket, amik a következő eredményt adják!



■ `x = 4.5`

■ `y = 'ELTE'`

■ `z = False`

1. Az X változó hosszabb, mint az Y változó értéke?
2. Az Y változó típusa string?
3. Ha az Y változó 2. elemét megszorozzuk X egész szám részével, a kifejezés értéke 'EEEE' lesz?
4. *Az X típusa megegyezik Y típusával* kifejezés értéke megegyezik Z változó értékével?
5. Nem Z változó értéke megegyezik Z értékével?

+ None



- Nonetype
- az érték hiányát jelzi



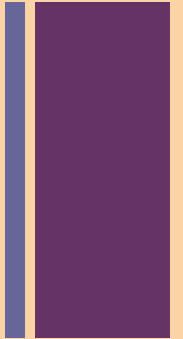
Speciális karakterek a magyar billentyűzeten



§ 0	' 1 ~	" 2 ˇ	+ 3 ^	! 4 ˇ	% 5 °	/ 6 ˇ	= 7 ˇ	(8 ˇ) 9 ˇ	Ö " ˇ	Ü " ˇ	Ó ˇ	←
↔	Q \ 	W 	E	R	T	Z —	U €	I	O " ˇ	P " ˇ	Ő ÷	Ú ×	↵
Caps Lock	A	S đ	D Đ	F [G]	H	J	K ł	L Ł	É \$	Á ß	Ű α	
↑	Í <	Y >	X #	C &	V @	B {	N }	M	? , ;	:	- *	↑	
Ctrl	Win Key	Alt								Alt Gr	Win Key	Menu	Ctrl



Otthoni Feladat



1. A https://github.com/gtakacse/CS_class_scripts/tree/master/homework címről töltsd le a `hw1_basic_data_types.py` fájlt.
2. Minden “Ide írd a kódot!” helyre illesztd be a saját megoldásodat!
3. Miután végeztél a fájlt mentsd el úgy, hogy a neve a `‘hw1_VEZETÉKNEVED_KERESZTNEVED.py’` legyen!
4. A fájlt küld el az email címemre!



Képek forrása



- BedBugs.org
- [https://en.wikipedia.org/wiki/George Boole](https://en.wikipedia.org/wiki/George_Boole)
- <https://news.digitalmediaacademy.org/tag/where-was-the-first-computer-bug-found/>
- <http://www.idgconnect.com/blog-abstract/10357/this-tech-history-september-first-bug-literally>
- <https://www.python.org/>
- plus.google.com
- <https://hu.wikipedia.org/wiki/AltGr>