

Python programozás

Jegyzet

Tartalomjegyzék

1	Helló, világ!	3
1.1	Írás konzolra	3
1.2	Kommentek	3
2	Primitív adattípusok	4
2.1	Szöveg (str)	4
2.1.1	Karakter	4
2.1.2	String-ek összefűzése	4
2.1.3	Ugyanazon érték megadása több változónak egyszerre	5
2.1.4	Többsoros string	5
2.1.5	Alapvető string műveletek	5
2.1.6	Bemenet	6
2.2	Szám (Number)	6
2.2.1	Egész szám (int)	6
2.2.2	Lebegőpontos szám (float)	6
2.2.3	Komplex szám (complex)	6
2.2.4	Aritmetikai műveletek	7
2.2.5	Értékadó operátorok	7
2.3	Logikai (Boolean)	8
2.3.1	Igazságtábla	8
2.3.2	Összehasonlítás (érték alapján)	8
2.3.3	Logikai műveletek	9
2.3.4	Objektumok összehasonlítása	9
2.4	Hogyan tárolódnak a memóriában a változók?	10

1 Helló, világ!

1.1 Írás konzolra

```
1 print("Helló, világ!")
```

```
>>> "Helló, világ!"
```

1.2 Kommentek

```
1 # Ez egy komment
2
3 # Több soros komment
4 """
5 Több
6 soros
7 komment
8 """
```

2 Primitív adattípusok

Másnéven, előre meghatározott (alapvető) adattípusok.

Értékkadás szintaxisa: <változó> = <érték>

- Szöveg (Text): str (string)
- Szám (Number): int, float, complex
- Logikai (Boolean): bool
- Értékhiány (None): NoneType

2.1 Szöveg (str)

```
1      greeting = "Helló"
2      name = "XYZ"
```

```
>>> greeting = "Helló"
>>> name = "XYZ"
```

2.1.1 Karakter

```
1      character = "X"
```

```
>>> character = "X"
```

2.1.2 String-ek összefűzése

```
1      greeting_user = greeting + ", " + name
```

```
>>> greeting_user = "Helló, XYZ"
```

2.1.3 Ugyanazon érték megadása több változónak egyszerre

```
1 friend_name = "XYZ"
2 friend1 = friend2 = friend3 = friend_name
```

```
>>> friend_name = "XYZ"
>>> friend1 = friend2 = friend3 = "XYZ"
```

2.1.4 Többsoros string

```
1 multi_line_string = """Ez egy
2 többsoros
3 string"""
```

```
>>> multi_line_string = """Ez egy
... többsoros
... string"""
```

2.1.5 Alapvető string műveletek

```
1 message = "Ez egy üzenet!"
2 upper_message = message.upper() # Nagybetűssé alakítás
3 lower_message = upper_message.lower() # Kisbetűssé alakítás
4 remove_whitespace = message.strip() # VEZETŐ és VÉG szóközők eltávolítása
5 replace_message = message.replace("üzenet", "szöveg") # Csere
```

```
>>> message = "Ez egy üzenet!"
>>> upper_message = "EZ EGY ÜZENET!"
>>> lower_message = "ez egy üzenet!"
>>> remove_whitespace = "Ez egy üzenet!"
>>> replace_message = "Ez egy szöveg!"
```

2.1.6 Bemenet

`input()` függvény: A felhasználótól vár bemenetet, és azt string-ként adja vissza.

```
1 user_input = input("Kérem, adj meg a nevét: ")
2 greeting_message = greeting + ", " + user_input
```

```
>>> user_input = input("Kérem, adj meg a nevét: ")
Kérem, adj meg a nevét: XYZ
>>> greeting_message = "Helló, XYZ"
```

2.2 Szám (Number)

2.2.1 Egész szám (int)

```
1 integer_number = 10
```

```
>>> integer_number = 10
```

2.2.2 Lebegőpontos szám (float)

```
1 float_number = 10.5
```

```
>>> float_number = 10.5
```

2.2.3 Komplex szám (complex)

$a + bj$, ahol a és b valós számok, j pedig az imaginárius egység.

```
1 complex_number = 1 + 2j
```

```
>>> complex_number = 1 + 2j
```

2.2.4 Aritmetikai műveletek

```
1      addition = 10 + 5 # Összeadás
2      subtraction = 10 - 5 # Kivonás
3      multiplication = 10 * 5 # Szorzás
4      division = 10 / 5 # Osztás
5      modulo = 10 % 5 # Maradékos osztás
```

```
>>> addition = 15
>>> subtraction = 5
>>> multiplication = 50
>>> division = 2.0
>>> modulo = 0
```

2.2.5 Értékadó operátorok

```
1      number = 10
2      number += 5 # number = number + 5
3      number -= 5 # number = number - 5
```

```
>>> number = 10
>>> number += 5
>>> number -= 5
```

2.3 Logikai (Boolean)

Két értéke lehet: `True` (1) vagy `False` (0).

2.3.1 Igazságtábla

A	B	A ÉS B
1	1	1
1	0	0
0	1	0
0	0	0

A ÉS B csak akkor igaz, ha
A és B is igaz.

A	B	A VAGY B
1	1	1
1	0	1
0	1	1
0	0	0

A VAGY B akkor igaz, ha
legalább az egyik igaz.

2.3.2 Összehasonlítás (érték alapján)

```
1     is_equal = 10 == 10
2     is_not_equal = 10 != 5
3     is_greater = 10 > 5
4     is_less = 10 < 5
5     is_greater_or_equal = 10 >= 10
6     is_less_or_equal = 10 <= 5
```

```
>>> is_equal = True
>>> is_not_equal = True
>>> is_greater = True
>>> is_less = False
>>> is_greater_or_equal = True
>>> is_less_or_equal = False
```


2.3.3 Logikai műveletek

Lásd: Igazságtábla.

```
1 number = 5
2 and_result = number == 5 and number != 2 # True, ha mindkettő igaz
3 or_result = number == 10 or number == 5 # True, ha legalább az egyik igaz
4 not_result = not True # True, ha az érték hamis
```

```
>>> and_result = True
>>> or_result = True
>>> not_result = False
```

2.3.4 Objektumok összehasonlítása

```
1 x = 9
2 y = x
3 is_result = x is y
4 is_not_result = x is not y
```

```
>>> is_result = True
>>> is_not_result = False
```

2.4 Hogyan tárolódnak a memóriában a változók?

```
1 value = "Hello, World!"
2 print(hex(id(value)))
3 print(type(value))
4
5 print("\n")
6
7 value = 10
8 print(hex(id(value)))
9 print(type(value))
10
11 other_value = value
12 print(hex(id(other_value)))
13 print(type(other_value))
14
15 value = 11
16 print(hex(id(value)))
17 print(type(value))
18
19 print("\n")
20 del value # Névtérből törlődik csak
21 # print(value)
```

