

03

Kintamieji ir užduotys



print

funkcija atspausdina nurodytą pranešimą į ekraną arba kitą standartinį išvesties įrenginį (mūsų atveju į *konsolę*)

`print(object(s))`

- `print` - funkcijos pavadinimas;
- `()` - vieta argumentams;
- `object(s)` - objektai, objektas, kintamasis, kintamieji kurių reikšmes norime atspausdinti į ekraną.

* `print` funkcija turi ir kitų argumentų, bet kol kas jų nesimokysime.

Naudosime iki kol išmoksime kitus būdus atvaizduoti duomenims.

01 pavyzdys

```
print("hello world")
```

- `print` - funkcijos pavadinimas;
- `()` - vieta argumentams;
- `"hello world"` - *string* tipo kintamasis, kurio vertė "hello world".

02 pavyzdys

```
print(2)
```

- `print` - funkcijos pavadinimas;
- `()` - vieta argumentams;
- `2` - *int* tipo kintamasis, kurio vertė 2.

01 Kintamieji | apibrėžimas

Kintamieji yra duomenų įverčių/reikšmių saugojimo talpyklos.



```
# variable_name = value
# v1, v2, v3 = value
# Value can be other variable
myname = "Vilius"
my_name = "Vilius"
_my_name = "Vilius"
myName = "Vilius"
MYNAME = "Vilius"
myname2 = "Vilius"
```

02 Kintamieji | vardai

- Kintamojo vardas **turi** prasidėti su raide arba apatiniu brūkšniu („_“);
 - `name;`
 - `_name;`
 - Kintamojo vardas **negali** prasidėti su skaičiumi;
 - `10years;`
 - Kintamojo varde **gali būti tik** raidžių-skaičių (alpha-numeric) ženklai ir apatiniai brūkšniai (A-z, 0-9, and _);
 - `yea.rs;`
 - `months_per_year;`
 - `profit_per_10_months;`
 - Kintamojo varduose **paistomi** simbolių dydžiai (mažosios ir didžiosios raidės) (case-sensitive);
 - `age;`
 - `Age;`
 - `AGE;`
 - Kintamojo vardas negali būti vienas iš *Python* raktažodžių;
 - `and;`
 - `continue;`
 - `...`
-

03 Kintamieji | skaičiai

```
# integers
integer_1 = 1
integer_2 = 35656222554887711
integer_3 = -3255522
```

```
# float
float_1 = 1.10
float_2 = 1.0
float_3 = -35.59
```

- Sveikieji; \mathbb{Z}
- Slankiojo kablelio; \mathbb{R}
- Kompleksiniai; \mathbb{C}

```
# Complex
complex_1 = 3+5j
complex_2 = 5j
complex_3 = -5j
```

04 Kintamieji | tekstas

Eilutė (tekstas) yra apsuptos arba viengubomis, arba dvigubomis kabutėmis.

```
print("Hello")  
  
# is same as  
print('Hello')
```

```
name="Vilius"  
name='vilius'
```

Method	Description
<code>capitalize()</code>	Converts the first character to upper case
<code>casefold()</code>	Converts string into lower case
<code>center()</code>	Returns a centered string
<code>count()</code>	Returns the number of times a specified value occurs in a string
<code>encode()</code>	Returns an encoded version of the string
<code>endswith()</code>	Returns true if the string ends with the specified value
<code>expandtabs()</code>	Sets the tab size of the string
<code>find()</code>	Searches the string for a specified value and returns the position of where it was found
<code>format()</code>	Formats specified values in a string
<code>format_map()</code>	Formats specified values in a string
<code>index()</code>	Searches the string for a specified value and returns the position of where it was found
<code>isalnum()</code>	Returns True if all characters in the string are alphanumeric
<code>isalpha()</code>	Returns True if all characters in the string are in the alphabet
<code>isascii()</code>	Returns True if all characters in the string are ascii characters
<code>isdecimal()</code>	Returns True if all characters in the string are decimals
<code>isdigit()</code>	Returns True if all characters in the string are digits
<code>isidentifier()</code>	Returns True if the string is an identifier
<code>islower()</code>	Returns True if all characters in the string are lower case
<code>isnumeric()</code>	Returns True if all characters in the string are numeric
<code>isprintable()</code>	Returns True if all characters in the string are printable
<code>isspace()</code>	Returns True if all characters in the string are whitespaces
<code>istitle()</code>	Returns True if the string follows the rules of a title
<code>isupper()</code>	Returns True if all characters in the string are upper case
<code>join()</code>	Joins the elements of an iterable to the end of the string
<code>ljust()</code>	Returns a left justified version of the string
<code>lower()</code>	Converts a string into lower case
<code>lstrip()</code>	Returns a left trim version of the string
<code>maketrans()</code>	Returns a translation table to be used in translations
<code>partition()</code>	Returns a tuple where the string is parted into three parts
<code>replace()</code>	Returns a string where a specified value is replaced with a specified value
<code>rfind()</code>	Searches the string for a specified value and returns the last position of where it was found
<code>rindex()</code>	Searches the string for a specified value and returns the last position of where it was found
<code>rjust()</code>	Returns a right justified version of the string
<code>rpartition()</code>	Returns a tuple where the string is parted into three parts
<code>split()</code>	Splits the string at the specified separator, and returns a list
<code>rstrip()</code>	Returns a right trim version of the string
<code>splitter()</code>	Splits the string at the specified separator, and returns a list
<code>splittimes()</code>	Splits the string at line breaks and returns a list
<code>startswith()</code>	Returns true if the string starts with the specified value
<code>strip()</code>	Returns a trimmed version of the string
<code>swapcase()</code>	Swaps cases, lower case becomes upper case and vice versa
<code>title()</code>	Converts the first character of each word to upper case
<code>translate()</code>	Returns a translated string
<code>upper()</code>	Converts a string into upper case
<code>zfill()</code>	Fills the string with a specified number of 0 values at the beginning

05 Kintamieji | loginės reikšmės

Loginės reikšmės reiškia vieną iš dviejų verčių: **True** or **False**.

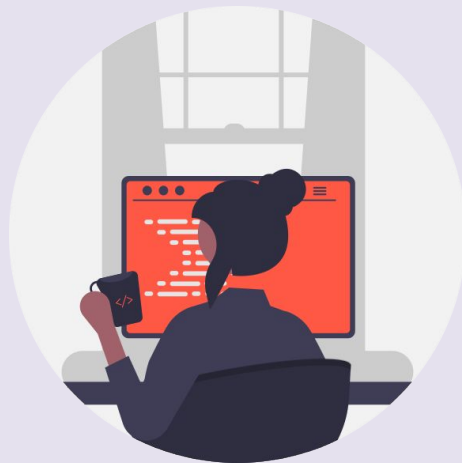
```
is_door_opened=True  
is_raining=False  
is_integer=False  
is_hungry=True
```

01 Operatoriai

simbolis arba raktažodis, naudojamas konkrečioms operacijoms su vienu ar daugiau operandų atlikti.

- Aritmetiniai;
- Priskyrimo*;
- Palyginimo;
- Loginiai;
- Tapatybės*;
- Priklausomumo*;
- Bitų*;

*sužinosim ateityje.



02 Operatoriai | aritmetiniai

Operatorius	Pavadinimas	Matematika	Pavyzdys
+	Sudėtis	$a+b$	$x + y$
-	Atimtis	$a-b$	$x - y$
*	Daugyba	$a \cdot b$	$x * y$
/	Dalyba	$a:b$	x / y
%	Dalybos modulis (liekana nustatymas) (angl. <i>modulo</i>)	$a \bmod b$	$x \% y$
**	Kėlimas laipsniu (eksponentizacijos)	a^b	$x ** y$
//	Dalyba be liekanos (angl. <i>flooring</i>)	$\lfloor a : b \rfloor$	$x // y$

03 Operatoriai | palyginimo

Operatorius	Pavadinimas	Pavyzdys
==	Lygu	<code>x == y</code>
!=	Nelygu	<code>x != y</code>
>	Daugiau už	<code>x > y</code>
<	Mažiau už	<code>x < y</code>
>=	Daugiau už arba lygu	<code>x >= y</code>
<=	Mažiau už arba lygu	<code>x <= y</code>

Operacijų su palyginimo operatoriais rezultatas yra loginė reikšmė (True arba False).

04 Operatoriai | loginiai

Operatorius	Pavadinimas	Pavyzdys
and	Grąžina True jeigu abu jei abu teiginiai (operandai) yra teisingi.	<code>x < 5 and x < 10</code>
or	Grąžina True jeigu vienas iš teiginių (operandų) yra teisingas.	<code>x < 5 or x < 4</code>
not	Apverčia rezultatą, grąžina False, jeigu rezultatas true	<code>x > y</code>

Operacijų su loginiais operatoriais rezultatas yra loginė reikšmė (True arba False).

```
# user is online
x = True
# user hasn't sent a message recently
y = False
# DND mode is off
z = False
# user is playing a game
a = True
# battery level is sufficient
b = False

if x and (y or a) and not z and not b:
    print("Send the notification!")
else:
    print("Don't send the notification.")
```

01 Užduotys | skaičiai

1. Suskaičiuoti **a**, kai $a = \frac{5 \cdot 10^5 - 6 \cdot (-699)^2}{25}$
2. Ištraukti kvadratinę šaknį iš 25, 99;
3. Suskačiuoti **c**, kai $c = \sqrt[5]{13111}$
4. Suskaičiuoti **b**, kai $b = \frac{c^6}{2.5}$
5. Patikrinti, ar **a** daugiau arba lygu už **b**;
6. Patikrinti, ar **a** lygu **b**;

- Visus atsakymus reikia pateikti su print(...).
 - Kintamieji **a**, **b**, **c** turi būti sukurti.
-

02 Užduotys | tekstas

1. Sukurti kintamąjį ***name***, kurio reikšmė yra jūsų vardas;
2. Sukurti kintamąjį ***python***, kurio reikšmė yra "python";
3. Patikrinti, ar jūsų vardas yra ilgesnis už žodį python;
4. Atspausdinti pirmąją savo vardo raidę, naudojant kintamąjį ***name***;
5. Patikrinti, ar pirmoji vardo raidė sutampa su "V".
6. Sukurti kintamąjį, kuris rodytų, ar jūsų vardas yra ilgesnis už žodį "python" ir pirmoji vardo raidė nesutampa su "A".

- Visus atsakymus reikia pateikti su print(...).
-