

# Medii integrate de proiectare interactivă

## Lucrarea de laborator nr. 1

**Tema:** Introducere în limbajul Python. Tipuri de date. Colecții de date.

Obiective:

- Instalare interpretor / IDE python
- Familiarizare cu limbajul python
  - Execuție instrucțiuni simple în interpretor
  - Implementare/executare programe simple folosind Python IDLE
- Studiul expresii, variabile, funcții.
- Instrucțiunile logice și decizionale, repetitive.
- Studiul colecțiilor de date: liste, tuple, seturi, dicționare.
- Metodele aplicabile tuple, seturi, dicționare.

Cerințe:

- Revizuiți materialul didactic de la curs
- Executați toate sarcinile de la punctul I.
- Calculați funcțiile din punctul II. Numărul variantei conform registrului.
- Executați sarcina de la punctul III varianta conform registrului.

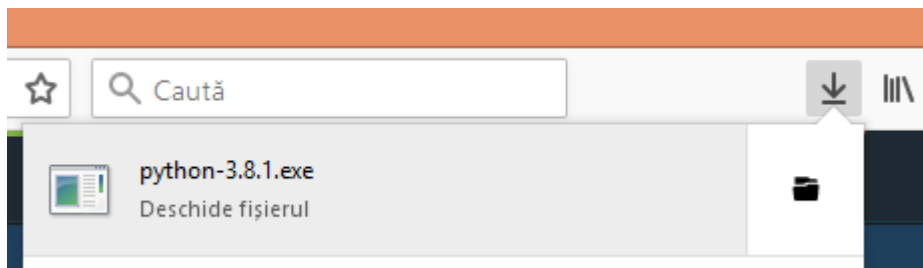
### Instalarea mediului de programare Python 3

Accesăm pagina oficială a limbajului Python, secțiunea Downloads, care are adresa URL de mai jos: <https://www.python.org/downloads/>

Alegeți ultima versiune stabilă 3.10.

În cazul din exemplu prezentat este **Python 3.8.1**.

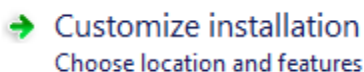
Apăsați butonul corespunzător, salvați fișierul și deschideți-l apoi:



După ce fișierul a fost lansat, va apărea un panou unde vă recomand să bifați înainte opțiunea de mai jos pentru a fi adăugate automat variabilele de sistem necesare:

- ☒ Install launcher for all users (recommended)
- ☒ Add Python 3.8 to PATH

Porniți procesul de instalare apăsând butonul:

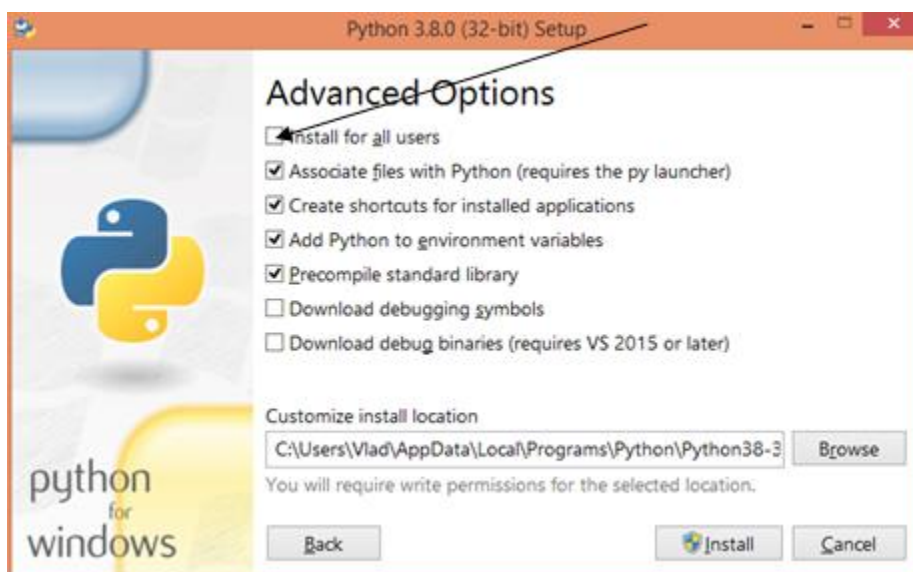


(apăsând butonul **Install Now**, Python se va instala doar pentru utilizatorul curent, iar modul de lucru va fi oarecum anevoios)

Lăsați totul bifat în prima fereastră:



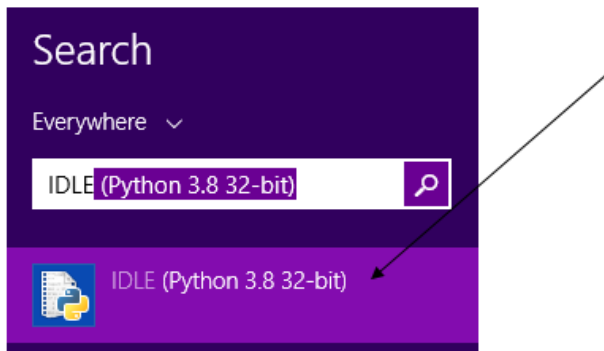
În cea de-a doua fereastră însă, bifați opțiunea **Install for all users**:



Astfel, locația de instalare va fi în locația "**C:\Program Files\Python38-32**", accesibilă tuturor utilizatorilor (în cazul meu, versiunea 3.8, sistemul de operare pe 32 de biți).

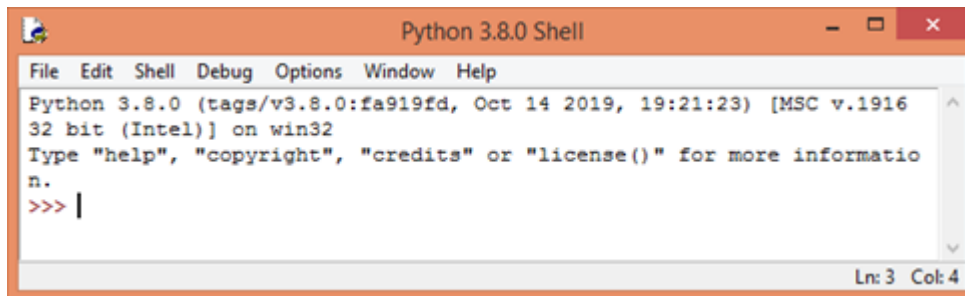
Apăsați butonul **Install**. Dacă totul a fost ok, apăsați apoi **Close**.

Mediul este instalat cu succes pe sistemul vostru, dar implicit nu aveți niciun shortcut spre **Python** pe *Desktop*, așa cum poate vă așteptați. Sistemul de operare prezentat este **Microsoft Windows**, așadar apăsăm butonul **Start**, apoi în bara de căutare scriem "**IDLE**", apoi selectăm (în cazul meu):



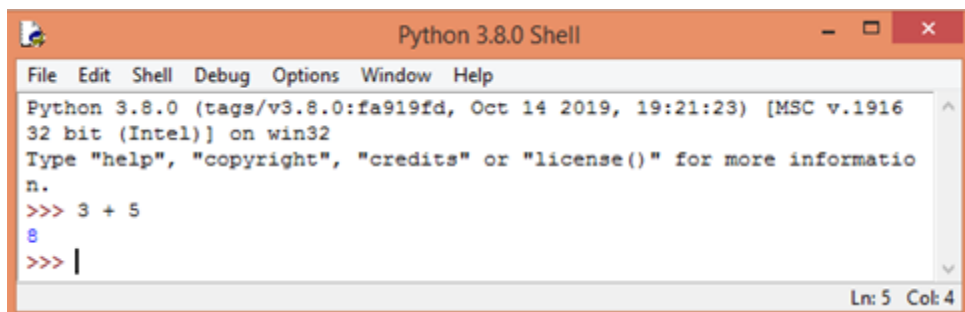
**Notă.** **IDLE** semnifică **I**ntegrated **D**evelopment and **L**earning **E**nvironment, adică un mediu integrat de dezvoltare și învățare.

Se va deschide următoarea fereastră:



De acum puteți scrie cod în Python!

Mai sus aveți linia de comandă, consola (**Interactive Python Shell**), iar poziția curentă este indicată prin cele trei caractere ">>>" și cursorul. Scrieți "**3 + 5**" și apăsați tasta **Enter**,



iar Python va afișa imediat rezultatul.  
Încercați alte operații aritmetice

## I. Rezolvați sarcinile, unde este necesar faceți explicații.

1. Să se execute în consola Python toate expresiile, să se explice rezultatele:

- |              |              |
|--------------|--------------|
| a) $2/5$     | h) $5*8.0$   |
| b) $7.9+1.1$ | i) $3.4-1.1$ |
| c) $6-2$     | j) $9 \% 2$  |
| d) $8+1$     | k) $9 \% -2$ |
| e) $9+1.1$   | l) $-9 \% 2$ |
| f) $2*7$     |              |
| g) $3.2/6$   |              |

2. Să se verifice tipul și valoarea următoarei expresii:

- a) `type(5.8e+3)`
- b) `type(1+4.0+2)`
- c) `type(None)`
- d) `type('float')`
- e) `type(4+2j)`

3. Să se compare și să se explice valorile expresiilor obținute:

- a)  $12.5/2.5$
- b) `int(12.5)/int(2.5)`
- c) `int(12.5/2.5)`

4. Să se anticipeze rezultatul fiecăreia dintre următoarele instrucțiuni, apoi să fie verificate în interpretorul Python.

- a) `str(6)*int('5')`
- b) `int('6')+float('6.1')`
- c) `str(6)*float('6.1')`
- d) `str(6/4)*2`

5. Se citește  $x$ , un număr real. Să se evalueze expresia:

$$f = \begin{cases} x, & x < 0; \\ 2 \cdot x, & 0 \leq x < 10; \\ 3 \cdot x, & 10 \leq x < 100; \\ 4 \cdot x, & x \geq 100. \end{cases}$$

10. Se citesc 4 numere întregi. Să se decidă dacă sunt distincte (adică nu există două egale între ele).

Indicație. Se compară primul număr cu toate celelalte, al doilea cu cele care îi urmează, al treilea cu al patrulea. Dacă în nici o comparație nu rezultă egalitate, se tipărește mesajul corespunzător.

## II. Variantele sarcinilor individuale:

Să se rezolve următoarele formule.

Nr. Var.	Formulele pentru calcul	Valorile datelor inițiale
----------	-------------------------	---------------------------

1	$Z = \begin{cases} (c \sin^4(x - \pi/3)) \\ (1/7 + \ln^2 y) \end{cases}; \quad v = \begin{cases} c + z^2 \\ (3 + z^3/5) \end{cases}$	x=0.345...7.45; cu pasul 0.1 y= -0,570 c=1.8
2	$y =  c^{vv/a} a^{-3} \sqrt{a/c} ; \quad w = \begin{cases} y + \sin^2 vv - a \\ 5,0001 + (y - vv)^3 \end{cases}$	c=1.037 a=0.15...1.25; cu pasul 0.1 vv=5.978
3	$a = e^{-h} j \cos(h k - j) + \sqrt{ j - k }; \quad b = a \ln(a j^4 - \sin 2h) - k.$	j= 0.1...1.0; cu pasul 0.1 k=2.7 h=0.94
4	$q = \left  \sqrt{d^2 + f} - f^2 \ln^3(d + x)/d \right ; \quad y = \cos^3 q^4 - q/\sqrt{f^2 + d^2}$	d=2.359 f=15.5 x=-0.1...2.9; cu pasul 0.1
5	$a = g^5 \lg^2(n + m)^2 + m/\sqrt{g + n}; \quad b = (ng^3 - a)/(e^{ax} - 7)$	g=1.0...1.64; cu pasul 0.1 m=2.14 n=1.34
6	$u = \sin^3(s - 1)/p - \operatorname{tg}^3(s - st); \quad o = \sqrt{u/s + \sin^4(s - p)}$	p=0.7 s=1.605 t=0.5...1,0; cu pasul 0.1
7	$a = 3 \operatorname{tg}^5(e - w^2) - \sqrt{q/w}; \quad b = a^3/q - \sin^3(a - qq)$	q=0.5...3,1; cu pasul 0.1 w=1.23 e=5.11
8	$at = \arcsin d - w^2 + \sqrt{ e - w^4 }; \quad as = at \ln(t) + \log_2 at$	d=1.711 t,e=0.1...1.0 w=2.1101
9	$af = e^{4z-d} g^3 - d z/g; \quad aa = af \operatorname{arctg}(z - d/g)^4$	d,z=4,101 g=1.0...2.0; cu pasul 0.1
10	$w = (s - \sin r^2) + \ln^3(r/q); \quad ff = e^{-w} \sqrt{ r - w } / ss;$	q=1.01 r=0.2...1.0; cu pasul 0.1 s=1.3;
11	$dd = \ln(d - v^{2r} + r)/(r - 1); \quad dr = dd^3 - d \arcsin^4(r + v)/r$	v=2.001 d=0.004...1.07; cu pasul 0.1 r=1.43
12	$ee = e^{-2r+h} \sqrt{ d - w^2 v \ln 2r }; \quad k = r^3 - \operatorname{tg} \sqrt{v^2 + ee^2}$	v=0.3...5.1; cu pasul 0.1 h=2.04; r=0.101; d=1.711 w=0.1201

13	$gg=v e^{-r} \log_2 rm / (mv-1); g=\sin^r(gg-m^{2r}-r)$	$v=1.3...7.0$ ; cu pasul 0.1 $m,r=0.87$
14	$ZZ=\frac{c\sqrt{ x-y^2 \ln 2c }}{(1/7+\ln^2 y)}; v=\frac{y^4(x-zz)^2}{(3+zz^3/5)}$	$x=0.3...4.5$ ; cu pasul 0.1 $y=-0.570$ $c=1.8$
15	$l=\log_2 v r+m^5 \arcsin^2(m); ll=\frac{1}{\sqrt{l+1}} e^{-m r}$	$m=0.4...0.5$ ; cu pasul 0.1 $r,v=1.8$

## Modulul math

```
>>> import math
```

```
>>> dir(math)
```

```
['__doc__', '__loader__', '__name__', '__package__', '__spec__', 'acos',
'acosh', 'asin', 'asinh', 'atan', 'atan2', 'atanh', 'ceil', 'comb',
'copysign', 'cos', 'cosh', 'degrees', 'dist', 'e', 'erf', 'erfc', 'exp',
'expm1', 'fabs', 'factorial', 'floor', 'fmod', 'frexp', 'fsum', 'gamma',
'gcd', 'hypot', 'inf', 'isclose', 'isfinite', 'isinf', 'isnan', 'isqrt',
'lcm', 'ldexp', 'lgamma', 'log', 'log10', 'log1p', 'log2', 'modf', 'nan',
'nextafter', 'perm', 'pi', 'pow', 'prod', 'radians', 'remainder', 'sin',
'sinh', 'sqrt', 'tan', 'tanh', 'tau', 'trunc', 'ulp']
```

```
>>> help(pow)
```

Help on built-in function pow in module builtins:

```
pow(base, exp, mod=None)
```

Equivalent to `base**exp` with 2 arguments or `base**exp % mod` with 3 arguments

Some types, such as ints, are able to use a more efficient algorithm when invoked using the three argument form.

Descrierea funcțiilor mai detaliată o găsiți pe adresa <https://docs.python.org/3/library/math.html#module-math>

## Structuri de control. Instrucțiunea if

Sintaxa instrucțiunii If este:

```
if conditie1:
    instruct
iuni elif
conditie2:
    instructiuni
else:
    instructiuni
```

Pot exista mai multe sectiuni *elif* sau nici una, iar sectiunea *else* poate apărea o singură dată sau niciodată.

Numarul 0, listele și tuplurile vide, stringurile vide și valoarea **None**, sunt considerate false daca sunt folosite în evaluarea unei condiții.

```

primul_script.py - C:/Users/Utilizator/Desktop/primul_script.py (3.9.7)
File Edit Format Run Options Window Help
n = int(input("Introduceti un numar intreg"))
if n<0:
    print("Numar negativ")
elif n==0:
    print("Zero")
else:
    print("Numar pozitiv")
Ln: 9 Col: 0

```

**III. Изучите предметную область варианта и продумайте, какую информацию следует хранить о ключевом объекте (например, для телефонного тарифа это наименование, стоимость, тип и т.д.).**

**Ориентируясь на Листинг 1, создайте небольшое приложение, которое хранит информацию об объекте выбранной предметной области и выводит ее на экран.**

Требования к данным:

- кол-во объектов (элементов словаря) должно быть не менее 3;
- кол-во атрибутов объекта должно быть не менее 5;
- по крайней мере 3 атрибута объекта должны иметь различный тип: int, float, str и др.;
- по крайней мере 1 атрибут объекта должен иметь тип list.

№ Вар ианта	Описание
1	Вуз
	Для предоставления информации абитуриентам в вузе работает система, которая хранит информацию о специальностях: наименование, описание, необходимые экзамены, проходной балл по ним и др.
2	Продуктовый магазин
	В продуктовом магазине работает терминал, который хранит информацию о продуктах: цена, количество, вес, состав (белки, жиры, углеводы) и др.
3	Кинотеатр
	Для взаимодействия с клиентами кинотеатр использует систему, которая хранит информацию о фильмах: название, список актеров, стоимость билета, список сеансов и др.
4	Книжный магазин
	В книжном магазине используется система, которая хранит информацию о книгах: название, цена, количество, список авторов, <u>ISBN</u> и др.
5	Банк



№ Варианта	Описание
	Банк предлагает клиентам электронный помощник, который хранит информацию о вкладах: сумма, валюта, срок, капитализация процентов и др.
6	Мобильный оператор
	Мобильный помощник оператора связи хранит информацию о тарифах: тип, стоимость звонка и др.
7	Фитнес-центр
	В фитнес-центре установлен терминал для клиентов, который хранит информацию о клубных картах: стоимость, тип, доступные занятия и др.
8	Агентство недвижимости
	Веб-сайт агентства недвижимости хранит информацию об арендуемых квартирах: адрес, количество комнат, площадь, цена, наличие балкона, лифта и др.
9	Магазин смартфонов
	В магазине используется система, которая хранит информацию о смартфонах: наименование, цена, количество, информация о комплектующих и др.
10	Пассажирский перевозчик
	Пассажирский перевозчик предоставляет клиентам веб-сайт, который хранит информацию о билетах: наименование, тип (безлимитный, социальный и др.), стоимость и др.

### Листинг 1 - Пример хранения и вывода информации о программистах в компании

```
# Это пример кода, он может быть произвольно усложнен
# в рамках ограничений темы
#
# Требования к данным:
# - кол-во объектов (элементов словаря) должно быть не менее 3;
# - кол-во атрибутов объекта должно быть не менее 5;
# - по крайней мере 3 атрибута объекта должны иметь различный тип:
#   int, float, str и др.;
# - по крайней мере 1 атрибут объекта должен иметь тип list.

# Замените атрибуты словаря и др. на соответствующие своему варианту

db = [
    {
        "name": "Иванов Иван",
        "birthday": "01/12/2002",
        "height": 170,
        "weight": 70.5,
        "car": True,
        "languages": ["C++", "Python"]
    },
    {
        "name": "Сергеев Сергей",
```

```

        "birthday": "01/06/2001",
        "height": 180,
        "weight": 110.4,
        "car": False,
        "languages": ["Polog", "Perl"]
    },
    ...
]

print("Содержимое базы данных ({}):".format(len(db)))

print("1.")
print("Имя: {}".format(db[0]["name"]))
print("День рождения: {}".format(db[0]["birthday"]))
print("Рост (см.): {}".format(db[0]["height"]))
print("Вес (кг.): {}".format(db[0]["weight"]))
print("Есть машина: {}".format(db[0]["car"]))
print("Языки программирования: {}".format(db[0]["languages"]))

print("2.")
print("Имя: {}".format(db[1]["name"]))
print("День рождения: {}".format(db[1]["birthday"]))
print("Рост (см.): {}".format(db[1]["height"]))
print("Вес (кг.): {}".format(db[1]["weight"]))
print("Есть машина: {}".format(db[1]["car"]))
print("Языки программирования: {}".format(db[1]["languages"]))

...

# -----
# Пример вывода:
#
# Содержимое базы данных (3):
# 1.
# Имя: Иванов Иван
# День рождения: 01/12/2002
# Рост (см.): 170
# Вес (кг.): 70.5
# Есть машина: True
# Языки программирования: ['C++', 'Python']
# 2.
# Имя: Сергеев Сергей
# День рождения: 01/06/2001
# Рост (см.): 180
# Вес (кг.): 110.4
# Есть машина: False
# Языки программирования: ["Polog", "Perl"]
#
# ...

```