

Введення тексту в змінні або в список

Для введення тексту використовується оператор `input()` Приклад:

```
a = input()
b = input()
```

Якщо необхідно дві частини тексту, що розділені пропуском ввести у відповідні змінні, то синтаксис наступний:

```
a, b = input().split()
```

При необхідності введення великої кількості текстових даних іноді зручніше скористатися списком:

```
lst = list(input().split())
```

Робота з текстом – конкатенація (склеювання):

```
a = 'Зима'
b = 'Літо'
print (a+b)
```

Результат: ЗимаЛіто

Робота з текстом – повторення:

```
a = 'Літо'
print (a*10)
```

Результат: ЛітоЛітоЛітоЛітоЛітоЛітоЛітоЛітоЛітоЛіто

Визначаємо довжину рядка

```
len('Добрий день шановна Маріє Іванівно!')
```

Результат: 35

Доступ по індексу

```
s = 'Привіт!'
print(s[0])
```

Результат: 'П'

```
print(s[2])
```

Результат: 'и'

```
print(s[-1])
```

Результат: '!

Отримання зрізу

Оператор отримання зрізу: [X:Y:Z]. X – початок зрізу, а Y – закінчення, Z - крок; символ с номером Y в зріз не входить. По замовчуванню перший індекс рівний 0, а другий – довжині рядка.

```
s = 'котик Мурчик любить рибу'
```

```
print(s[6:12])
```

Результат: 'Мурчик'

```
print(s[:5])
```

Результат: 'котик'

```
print(s[20:-2])
```

Результат: 'ри'

```
print(s[1:])
```

Результат: 'отик Мурчик любить рибу'

```
print(s[:])
```

Результат: 'котик Мурчик любить рибу'

Можна задати крок і напрямок зрізу

```
print(s[6:18:4])
```

Результат: 'Мию'

```
print(s[::-1])
```

Результат: 'убир ьтибюл кичруМ киток'

```
print(s[11:5:-1])
```

Результат: ' кичруМ'

```
print(s[6:12][::-1])
```

Результат: ' кичруМ'

Конвертація числа в текст

```
a = 5
b = str(a)
```

Конвертація тексту в число

```
a = '5'
b = int(a)
```

Підсумкова таблиця: «Функції і методи рядків»

```
x = 'Микола'
y = 'Оленка'
z = 'Степан'
```

<i>Функція чи метод</i>	<i>Призначення</i>	<i>Приклад</i>	<i>Результат</i>
result = x + y	Конкатенація (склеювання рядків)	result = x + y	'МиколаОленка'
result = x * 2	Повтор рядка	result = x * 2	'МиколаМикола'
result = x[i]	Звернення по індексу	result = x[0]	'М'
result = x[[start]:[end]:[step]]	Отримання зрізу	result = x[1:3]	'ик'
result = len(x)	Визначення довжини рядка	result = len(x)	6
x.lstrip([chars])	Видалення символів пропуску (або символів chars) на початку рядка	' Коля'.lstrip()	'Коля'
x.rstrip([chars])	Видалення символів пропуску (або символів chars) в кінці рядка	'Коля '.rstrip()	'Коля'
x.strip([chars])	Видалення символів пропуску (або символів chars) на початку і в кінці рядка	' Коля '.strip()	'Коля'
x.find(str, [start],[end])	Пошук підрядка в рядку. Повертає номер першого входження або -1	result = x.find('я')	-1
x.rfind(str, [start],[end])	Пошук підрядка в рядку. Повертає номер останнього входження або -1	result = x.rfind('л')	4
x.replace(шаблон, заміна)	Заміна шаблону	result = x.replace('икол', 'урк')	'Мурка'
x.isdigit()	Чи складається рядок з цифр	result = x.isdigit()	False
x.isalpha()	Чи складається рядок з букв	result = x.isalpha()	True
x.isalnum()	Чи складається рядок з букв або цифр	result = x.isalnum()	True
x.islower()	Чи складається рядок виключно з символів в нижньому регістрі	result = x.islower()	False
x.isupper()	Чи складається рядок виключно з символів в верхньому регістрі	result = x.isupper()	False
x.isspace()	Чи включає рядок символи, що не відображаються (пробіл, символи '\f' '\n' '\r' '\t' '\v')	result = x.isspace()	False
x.istitle()	Чи починаються слова в рядку з великої літери (навіть якщо їх декілька)	result = x.istitle()	True
z.upper()	Перетворення рядка до верхнього регістру	result = z.upper()	'СТЕПАН'
z.lower()	Перетворення рядка до нижнього регістру	result = z.lower()	'степан'
z.startswith(шаблон)	Чи починається рядок з шаблону	result = z.startswith('Ст')	True
z.endswith(шаблон)	Чи закінчується рядок шаблоном	result = z.endswith('ан')	True
z.join(список)	Збирання рядка зі списку з розділювачем	result = '+'.join(z)	'С+т+е+п+а+н'

<i>Функція чи метод</i>	<i>Призначення</i>	<i>Приклад</i>	<i>Результат</i>
ord(символ)	Символ в його код Unicode	ord('Я')	1071
chr(число)	Код Unicode в символ	chr(1071)	'Я'
x.capitalize()	Переводить перший символ рядка в верхній регістр, а всі інші в нижній	result = 'вАсЯ'.capitalize()	'Вася'
x.center(width, [fill])	Повертає відцентрований рядок, по краях якого стоїть символ fill (пробіл за замовчуванням)	'Оленка'.center(10, '+')	'++Оленка++'
x.ljust(width, [fill])	Повертає рядок, довжиною не меншою width, в разі потреби заповнюючи останні символи символом fill (пробіл за замовчуванням)	'Оленка'.ljust(10, '+')	'Оленка++++'
x.rjust(width, [fill])	Повертає рядок, довжиною не меншою width, в разі потреби заповнюючи перші символи символом fill (пробіл за замовчуванням)	'Оленка'.rjust(10, '+')	'++++Оленка'
x.title()	Першу букву кожного слова переводить в верхній регістр, а всі інші в нижній	'добрИй дЕнь'.title()	'Добрый День'
x.swapcase()	Переводить символи нижнього регістру в верхній, а верхнього - в нижній	x.swapcase()	'сТЕПАН'