



9-11 классы

Программирование на Python

Презентация занятия

СТРОКИ И ИНДЕКСЫ

5 занятие



Минцифры
России



20.35
УНИВЕРСИТЕТ

Программирование
на Python

Теоретическая часть

СТРОКИ И ИНДЕКСЫ

5 занятие



2022

Тема: Строки и индексы

```
1  Типы данных в Python:
2  1. str - string - строка
3     (последовательность символов - 'hello')
4  2. int - integer - целый
5     (целые числа - 123, -1, 0, 1000)
6  3. float - floating point - плавающая точка
7     (дробь) 123.0 - 12.3 - 1.23 - 0.123
8  4. bool - boolean type - булевы значения
9     (true-false, истина-ложь) 5 != 3, 3 > 0
10
11  Логическое выражение - это такое выражение,
12  относительно которого мы можем сказать, что
13  оно либо истинно, либо ложно
```

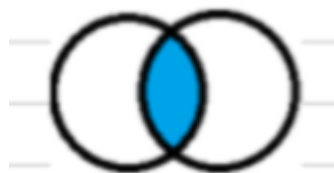


Тема: Строки и индексы



```
if <логическое выр.>:  
    <действия, которые будут>  
    <выполнены, если лог. выраж ист.>  
else:  
    <действия, которые будут>  
    <выполнены, если лог. выраж ложно>
```

```
1 state = input('How are you?')  
2 if state == 'sleepy':  
3     print('Sleep in')  
4 else:  
5     print('Start your day!')
```



Тема: Строки и индексы



ROCK



SCISSORS



PAPER

Какие ключевые слова, команды и конструкции вы использовали в при решении данной задачи?

Реализуйте аналогичную задачу, только теперь у пользователей после игры спрашивается, хотят ли они сыграть ещё раз.



Тема: Строки и индексы

Строка - любая последовательность символов, заключенная в кавычки

```
1  name      = 'Inginirium'
2  second_name = 'МГТУ им Н Э Баумана '
3  address    = "105005, Москва, 2-я Бауманская ул., д. 5, стр. 1"
4  input_str  = input()
5  number_str = str(1830)
6  bmstu_name = second_name + number_str
```





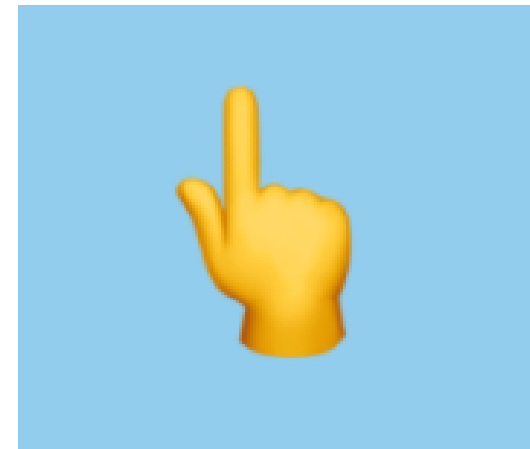
Тема: Строки и индексы

Индекс - порядковый номер символа в строке, начинающийся с 0.

0 1 2 3 4 5 6 7 8 9

INGINIRIUM

```
1 name = 'Inginirium'
2 print(name[0])
3 print(name[1])
4 print(name[2])
5 print(name[3])
6 print(name[4])
7 print(name[5])
8 print(name[6])
9 ...
```





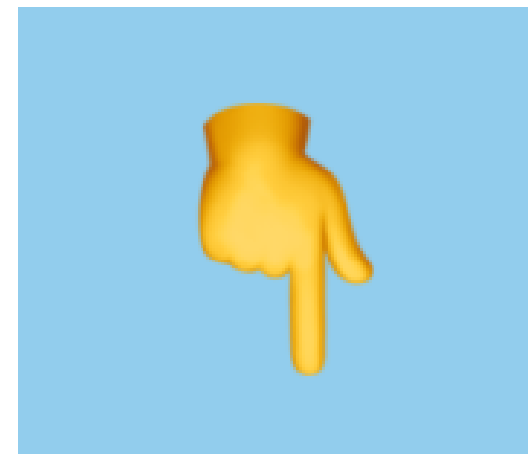
Тема: Строки и индексы

0 1 2 3 4 5 6 7 8 9

INGINIRIUM

-10 -9 -8 -7 -6 -5 -4 -3 -2 -1

```
1 name = 'Inginirium'
2 print(name[-1])
3 print(name[-2])
4 print(name[-3])
5 print(name[-4])
6 print(name[-5])
7 print(name[-6])
8 print(name[-7])
9 ...
```





Тема: Строки и индексы

```
1 name = 'Inginirium'  
2 print(name[10])  
3 name[3] = 'A'
```

`print(name[10])`
IndexError: string index out of range

`name[3] = 'A'`
TypeError: 'str' object does not support item assignment



Тема: Строки и индексы

```
1 print(len('Inginirium'))
2 print(len("Век живи - век учись"))
3 print(len('Функция len возвращает количество символов в строке'))
```

```
1 from_user = input('Привет, что нового? ')
2 if len(from_user) > 30:
3     print('Мне сейчас некогда читать столько много букв..')
4     print('Мама зовёт кушать, чуть позже обязательно прочту 😊')
5 else:
6     print('Оу, вот оно что.. Ну здорово 👍')
```

Тема: Строки и индексы

```
1 print('ура' in 'Муравей')
2 print('стар' not in 'Всем на старт!')
```

```
1 mood = input('Привет, как ты? ')
2 if 'хорошо' in mood:
3     print('👍')
4 elif 'отлично' in mood:
5     print('🎉')
6 elif 'плохо' in mood:
7     print('Всё будет хорошо 😊')
```

Привет, как ты? Просто отлично



Тема: Строки и индексы

ASCII table

American Standard Code for Information Interchange

Char	Dec	Oct	Hex	Char	Dec	Oct	Hex	Char	Dec	Oct	Hex
(sp)	32	0040	0x20	@	64	0100	0x40	`	96	0140	0x60
!	33	0041	0x21	A	65	0101	0x41	a	97	0141	0x61
"	34	0042	0x22	B	66	0102	0x42	b	98	0142	0x62
#	35	0043	0x23	C	67	0103	0x43	c	99	0143	0x63
\$	36	0044	0x24	D	68	0104	0x44	d	100	0144	0x64
%	37	0045	0x25	E	69	0105	0x45	e	101	0145	0x65
&	38	0046	0x26	F	70	0106	0x46	f	102	0146	0x66
'	39	0047	0x27	G	71	0107	0x47	g	103	0147	0x67
(40	0050	0x28	H	72	0110	0x48	h	104	0150	0x68
)	41	0051	0x29	I	73	0111	0x49	i	105	0151	0x69
*	42	0052	0x2a	J	74	0112	0x4a	j	106	0152	0x6a
+	43	0053	0x2b	K	75	0113	0x4b	k	107	0153	0x6b
,	44	0054	0x2c	L	76	0114	0x4c	l	108	0154	0x6c
-	45	0055	0x2d	M	77	0115	0x4d	m	109	0155	0x6d
.	46	0056	0x2e	N	78	0116	0x4e	n	110	0156	0x6e
/	47	0057	0x2f	O	79	0117	0x4f	o	111	0157	0x6f
0	48	0060	0x30	P	80	0120	0x50	p	112	0160	0x70
1	49	0061	0x31	Q	81	0121	0x51	q	113	0161	0x71
2	50	0062	0x32	R	82	0122	0x52	r	114	0162	0x72
3	51	0063	0x33	S	83	0123	0x53	s	115	0163	0x73
4	52	0064	0x34	T	84	0124	0x54	t	116	0164	0x74
5	53	0065	0x35	U	85	0125	0x55	u	117	0165	0x75
6	54	0066	0x36	V	86	0126	0x56	v	118	0166	0x76
7	55	0067	0x37	W	87	0127	0x57	w	119	0167	0x77
8	56	0070	0x38	X	88	0130	0x58	x	120	0170	0x78
9	57	0071	0x39	Y	89	0131	0x59	y	121	0171	0x79
:	58	0072	0x3a	Z	90	0132	0x5a	z	122	0172	0x7a
;	59	0073	0x3b	[91	0133	0x5b	{	123	0173	0x7b
<	60	0074	0x3c	\	92	0134	0x5c		124	0174	0x7c
=	61	0075	0x3d]	93	0135	0x5d	}	125	0175	0x7d
>	62	0076	0x3e	^	94	0136	0x5e	~	126	0176	0x7e
?	63	0077	0x3f	_	95	0137	0x5f				



Тема: Строки и индексы



```
print(chr(125)) # char - character  
print(ord('%')) # ord - order  
print(chr(ord('a') + 3))
```



Тема: Строки и индексы

Вы пишете код, запускаете его, и
ничего не работает как надо



Тема: Строки и индексы

Процесс изучения и исправления ошибок в работе программы (их часто называют багами, от англ. bug — жук), называется отладкой, по-английски — debugging, одно из значений — «удаление насекомых с растений». По одной из версий, именно так приходилось чинить компьютеры на заре компьютерной эпохи.

Чтобы понять, где возникла ошибка, приходится: узнавать текущие значения переменных; выяснять, по какому пути выполнялась программа.



Тема: Строки и индексы

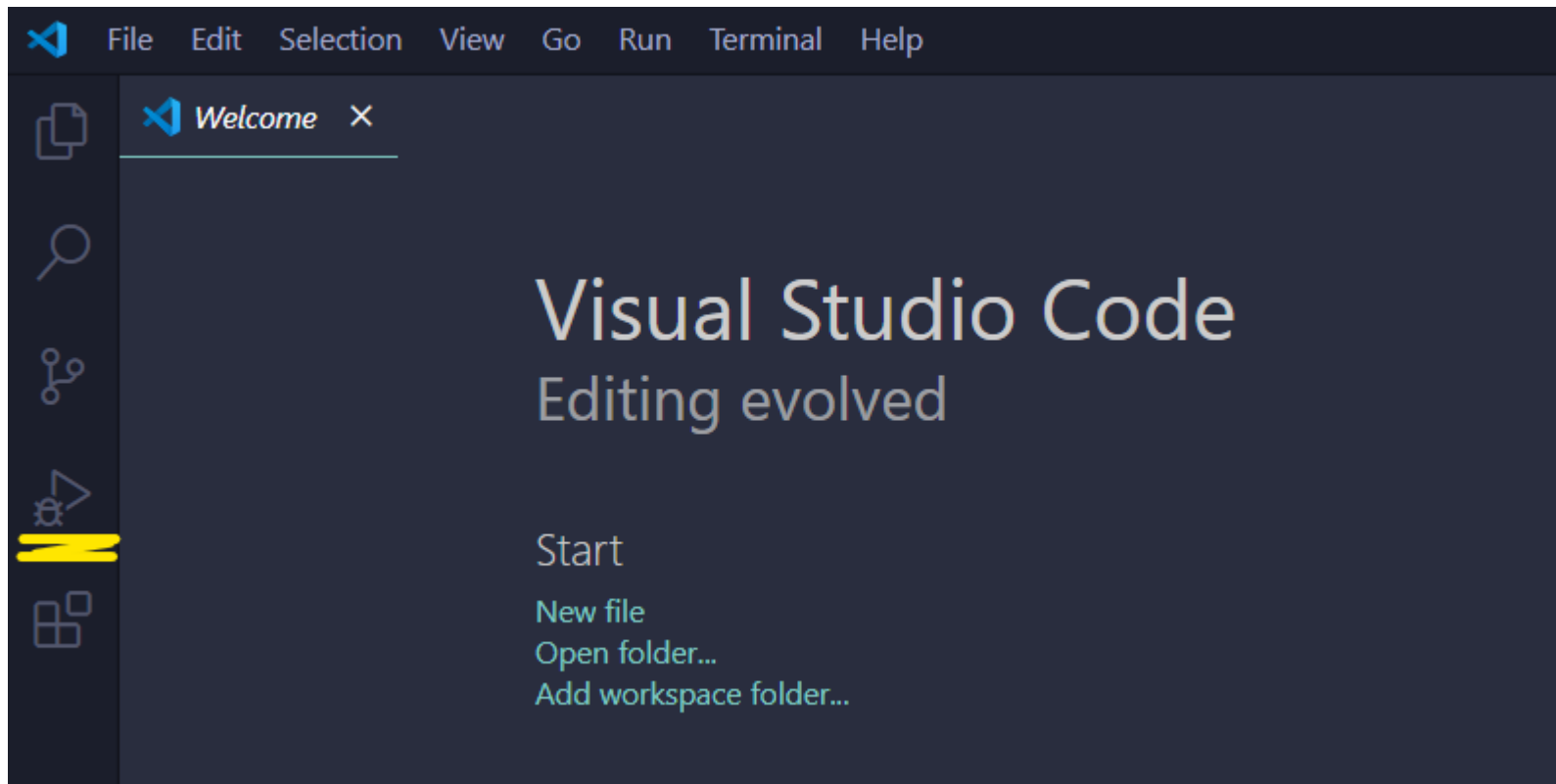
Синтаксическая ошибка: происходит, когда встречается код, который не соответствует правилам языка Python.

Ошибка исполнения: как следует из названия, случается во время исполнения программы.

Логическая ошибка (смысловая): происходит, когда программа ведет себя не так, как было задумано. Интерпретатор запускает программу и не сообщает об ошибке. Но все работает не так, как хотелось бы.



Тема: Строки и индексы





Visual Studio Code interface showing a Python script and its execution.

Left Panel (Variables):

- VARIABLES**
 - Locals**
 - special variables
 - function variables
 - first_num: 123
 - oper: '+'
 - second_num: 321
 - Globals**
- WATCH**
- CALL STACK** (PAUSED ON BREAKPOINT)
 - <module> test.py 19:1
- BREAKPOINTS**
 - ☐ Raised Exceptions
 - ☒ Uncaught Exceptions
 - ☒ test.py 19

Editor (test.py):

```
def calculator(num_1, num_2, oper):
    result = 0
    if oper == '+':
        result = num_1 + num_2
    elif oper == '-':
        result = num_1 - num_2
    elif oper == '*':
        result = num_1 * num_2
    elif oper == '/' and num_2:
        result = num_1 / num_2
    else:
        return None
    return result

first_num = int(input('Please, enter first number: '))
oper = input('Enter arithm operator (+, -, *, /): ')
second_num = int(input('Enter second number: '))
answer = calculator(first_num, second_num, oper)
print(first_num, oper, second_num, '=', answer)
```

Terminal:

```
sions\ms-python.python-2020.9.114305\pythonFiles\lib\python\debugpy
Please, enter first number: 123
Enter arithm operator (+, -, *, /): +
Enter second number: 321
PS C:\Users\kprie\Desktop\python> cd 'c:\Users\kprie\Desktop\python'
n.exe' 'c:\Users\kprie\.vscode\extensions\ms-python.python-2020.9.1
\kprie\Desktop\python\test.py'
Please, enter first number: 123
Enter arithm operator (+, -, *, /): +
Enter second number: 321
```



Программирование
на Python

Практическая часть

СТРОКИ И ИНДЕКСЫ

5 занятие



2022

Тема: Строки и индексы

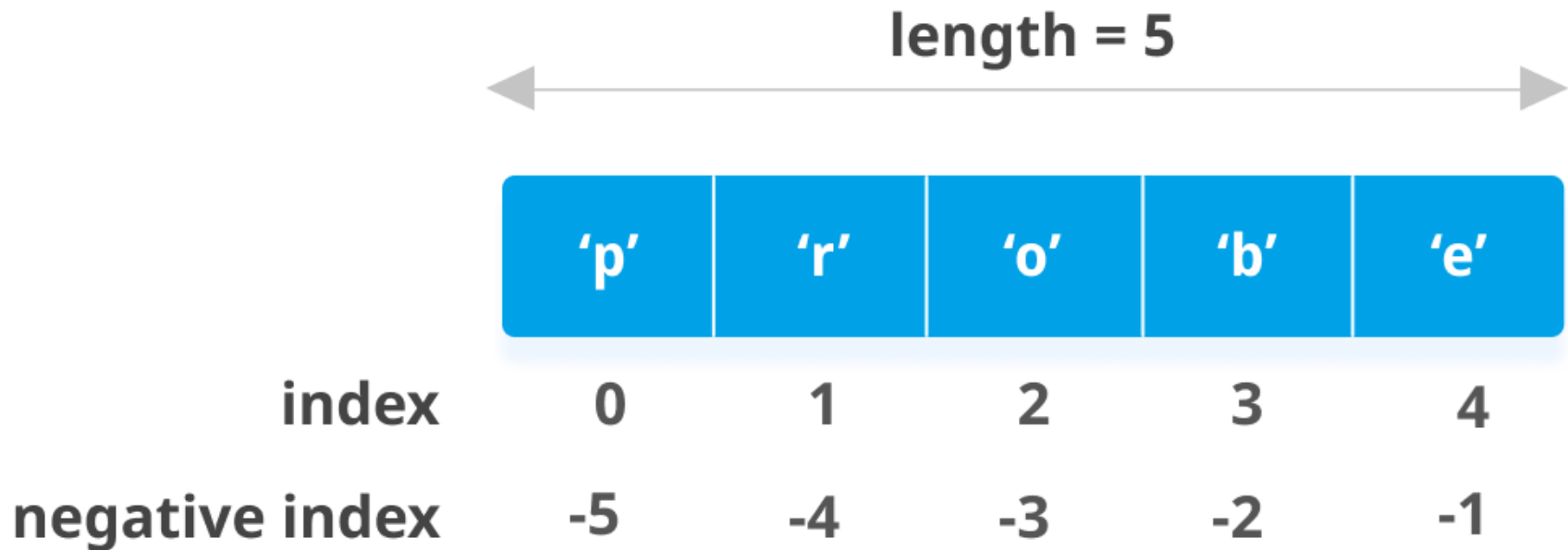
Задание 1

Чему равна длина?
Назовите индексы 'р'
Назовите индексы 'о'
Назовите индексы 'е'

'р'	'r'	'о'	'b'	'е'
-----	-----	-----	-----	-----



Тема: Строки и индексы





Тема: Строки и индексы

Задание 2

Напишите программу, которая выводит четвёртую букву введённого пользователем слова. Например, в слове "Привет" четвёртая буква — "в". Если в слове нет четвёртой буквы, потому что оно слишком короткое, то программа выводит "НЕТ".

*

Напишите программу, которая считывает слово и выводит его последнюю букву.



Тема: Строки и индексы

Задание 3

Пользователь вводит 2 слова. Если последняя буква первого слова совпала с первой буквой второго, то в этом случае необходимо вывести слово “Верно”, в противном случае “Ошибка”.

Поиграем в города?

— **Москва**
Атланта —
Амстердам —

Тема: Строки и индексы

Задание 4

Пользователь (или несколько пользователей за одним компьютером) вводит слова. Начиная со второго введенного слова, программа проверяет, совпадает ли первая буква свежевведенного слова с последней буквой предыдущего. Если да, то программа работает дальше (считывает очередное слово). Если нет — выводит последнее на этот момент введенное слово и завершает работу.

Поиграем в города?

— **Москва**
Атланта —
Амстердам —



Тема: Строки и индексы

Задание 5

Напишите программу, которая будет работать до тех пор, пока пользователь не введёт корректный пароль:

- если пароль короче 8 символов, программа выводит на экран слово "Короткий" и заново считывает версию пароля;
- если же пароль достаточно длинный, но в нём содержится сочетание символов "123", программа выводит на экран слово "Простой" и снова считывает версию пароля;
- если же и вторая проверка пройдена успешно, программа выводит "ОК" и заканчивает свою работу.

