

Стать IT-шником может каждый



# Revision



# ВАЖНО!

- Камера должна быть включена на протяжении всего занятия.
- Если у Вас возник вопрос в процессе занятия, пожалуйста, поднимите руку и дождитесь, пока преподаватель закончит мысль и спросит Вас, также можно задать вопрос в чате или когда преподаватель скажет, что начался блок вопросов.
- Организационные вопросы по обучению решаются с кураторами, а не на тематических занятиях.
- Вести себя уважительно и этично по отношению к остальным участникам занятия.
- Во время занятия будут интерактивные задания, будьте готовы включить микрофон или демонстрацию экрана по просьбе преподавателя.

# ПЛАН ЗАНЯТИЯ

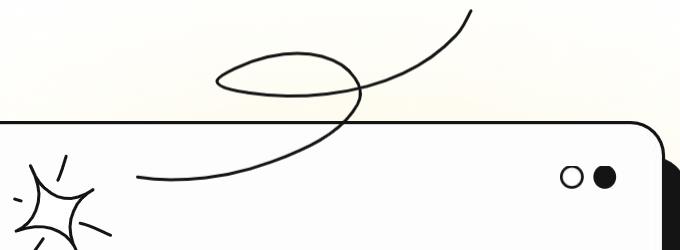
Введение

Основной блок

Задание для закрепления

Вопросы по основному блоку

Заключение



Стать IT-шником может каждый



# ПОВТОРЕНИЕ ИЗУЧЕННОГО



Стать IT-шником может каждый



# Повторение

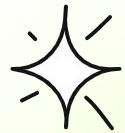
- Подключение к базе данных
- Работа с базой данных



Стать IT-шником может каждый



# ВОПРОСЫ





# Введение

Основные типы данных и математические операции

○ ●

if и while

Методы для работы со строками

for и функции

Стать IT-шником может каждый



# ОСНОВНОЙ БЛОК



Стать IT-шником может каждый



# Основные типы данных и математические операции



# Основные типы данных в Python



Тип данных	Значение	Определение в Python	Вариант использования
Целые числа	-3, -2, -1, 0, 1, 2, 3	int	a = int(input())
Вещественные числа	-1.5, -1.1, 0.6, 1.7	float	a = float(input())
Булевые значения	True, False	True, False	flag = True
Строка	'abracadabra'	str	a = str(5)

# Основные типы данных в Python



Тип данных	Значение	Определение в Python	Вариант использования
Список	[1, 2, 3], ['a', 'b', 'c']	list	a = list(['a', 'b', 'c'])
Кортеж	('red', 'blue', 'green')	tuple	a = tuple('red', 'blue', 'green'))
Множество	{'black', 'blue', 'white'}, {1, 3, 9, 7}	set	a = set('black', 'blue', 'white'))
Словарь	{'color': 'red', 'model': 'VC6', 'dimensions': '30×50'}	dict	a = dict(color='red', model='VC6', dimensions='30×50')

Стать IT-шником может каждый



# Основные математические операции в Python

Оператор	Операция	Пример	Результат
+	Сложение	$478 + 32$ "a"+ "b"	510 "ab"
-	Вычитание	$999 - 135$	864
*	Умножение	$123 * 321$ "hi"*3	39483 "hihihi"
/	Деление	$39 / 2$	19.5

Стать IT-шником может каждый



# if и while



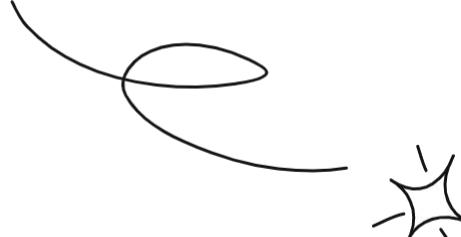
## Полный синтаксис условного оператора if

```
if условие1:  
    # выполняемые действия, если условие1 истинно  
elif условие2:  
    # выполняемые действия, если условие2 истинно  
else:  
    # выполняемые действия, если все условия ложны
```



## Цикл while —

это цикл с предусловием, то есть он выполняет блок кода, пока условие истинно.



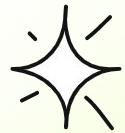
# Цикл while в Python

```
while условие:  
    # выполняемые действия  
    # возможно использование break и continue  
else:  
    # выполняемые действия, если цикл завершился без break
```

Стать IT-шником может каждый



# ВОПРОСЫ



Стать IT-шником может каждый



# ЗАДАНИЕ ДЛЯ ЗАКРЕПЛЕНИЯ





## Попробуйте:

Объясните, что происходит в этом фрагменте программы. Перечислите ситуации, когда цикл продолжает выполняться, а когда завершается:

```
list_of_tables = ["users", "product", "sales"]
str_of_tables = ', '.join(list_of_tables)
tables=["empty"]

while not set(tables).issubset(list_of_tables):
    tables = input(f"Введите таблицу или список таблиц из набора {str_of_tables}: ").split()
    if set(tables) == {"users", "product"}:
        print("Эти таблицы не связаны. Сделайте другой выбор")
        tables=["empty"]
```

Стать IT-шником может каждый



# Методы для работы со строками



# Методы для работы со строками

**S.split(символ)**

**S.isdigit()**

**S.isalpha()**

Разбиение строки по  
разделителю

Состоит ли строка из цифр

Состоит ли строка из букв

# Методы для работы со строками

**S.islower()**

**S.isupper()**

**S.upper()**

Состоит ли строка из символов в  
нижнем регистре

Состоит ли строка из символов в  
верхнем регистре

Преобразование строки к  
верхнему регистру

# Методы для работы со строками

**S.lower()**

**S.join(список)**

**S.strip([chars])**

Преобразование строки к  
нижнему регистру

Сборка строки из списка с  
разделителем S

Удаление пробельных символов в  
начале и в конце строки

Стать IT-шником может каждый

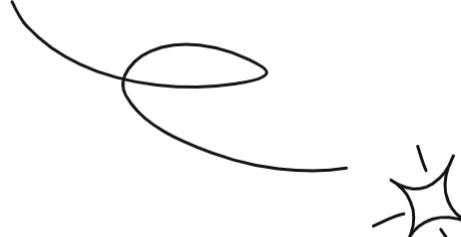


# for и функции



## Цикл for —

это цикл, который используется для выполнения повторяющихся операций над элементами последовательности. Он позволяет пройти по всем элементам списка, строки, кортежа или другой итерируемой структуры данных. Каждая итерация цикла представляет собой обработку одного элемента.



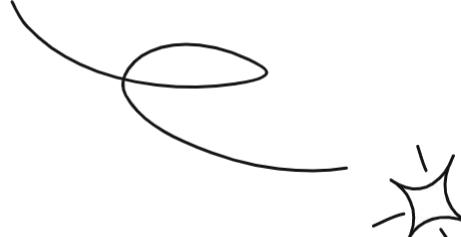
# Цикл for в Python

```
for i in итератор:  
    # выполняемые действия  
    # возможно использование break и continue  
else:  
    # выполняемые действия, если цикл завершился без break
```



## ФУНКЦИЯ —

это **блок кода**, который выполняет определенную задачу и может быть многократно использован в программе.



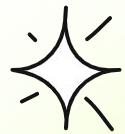
# ФУНКЦИЯ

```
def имя_функции(аргументы):  
    # выполняемые действия  
    return возвращаемые значения
```

Стать IT-шником может каждый



# ВОПРОСЫ



Стать IT-шником может каждый



# ЗАДАНИЕ ДЛЯ ЗАКРЕПЛЕНИЯ





## Попробуйте:

Объясните, что происходит в этом фрагменте программы.

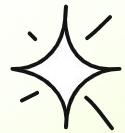


```
def input_columns(table):
    columns = {"users" : ["id", "name", "age"],
               "product": ["pid", "prod", "quantity"],
               "sales": ["sid", "pid", "id"]}
    field = input(f"Выберите одно из полей {columns[table]} этой таблицы или введите * для
выбора всех: ").strip()
    if field not in columns[table] and field != "*":
        print("Введено некорректное значение. Будет использовано значение *")
        field = "*"
    return "SELECT " + field + " "
```

Стать IT-шником может каждый



# ВОПРОСЫ



Стать IT-шником может каждый



# ПРАКТИЧЕСКАЯ РАБОТА



## Практическое задание

В базе данных ich\_edit три таблицы. Users с полями (id, name, age), Products с полями (pid, prod, quantity) и Sales с полями (sid, id, pid). Написать мини-интерфейс к базе данных, который умеет выполнять разные команды.

1. Выбрать таблицу для запроса. Предусмотреть возможность выбрать несколько таблиц. Вывести результат их соединения, если это возможно, или сообщение об ошибке.
2. Выбрать одно поле из выбранной таблицы и искомое значение этого поля. Вывести все подходящие строки

Стать IT-шником может каждый



# ОСТАВШИЕСЯ ВОПРОСЫ





## Домашнее задание

Доработать мини-интерфейс к базе данных, который был сделан на занятии. Новые  
возможности интерфейса:



1. Ввести список полей выбранной таблицы.
2. При вводе искомого значения добавить возможность выбора знака - найти записи, в которых выбранное поле больше, меньше или равно введенному значению.

Стать IT-шником может каждый



# ЗАКЛЮЧЕНИЕ

