

Введение

Python представляет популярный высокоуровневый язык программирования, который предназначен для создания приложений различных типов. Это и веб-приложения, и игры, и настольные программы, и работа с базами данных. Довольно большое распространение питон получил в области машинного обучения и исследований искусственного интеллекта.

Установка python

<https://python.org/downloads/>

Скачиваем 3.11

Устанавливаем Не забываем поставить такие-то галочки...

Установка Visual Studio Code

<https://code.visualstudio.com>

Скачиваем

Устанавливаем

Перезапускаем ПК

Первые шаги

```
# Добавляем переменную с группой, фамилией и именем
$Env:id = "p50-5-18-Gatskan-Maxim"

# Переходим в папку документов
cd ~\Documents\

# Создаем папку для проектов
mkdir .\mptpy\$Env:id

# переходим в созданную папку
cd .\mptpy\$Env:id

# Открываем Visual Studio Code в этой папке
code .
```

Hello MPT

Создаём папку `study-1`

В ней создаем файл `hello.py`

Пишем код

```
# создание переменной без аннотации типа
hello_msg = "Hello"

# print - вывод в консоль переданных строк
print(hello_msg, "MPT!")
```

```
# Запускаем файл
python hello.py
```

Hello U

```
# создание переменной без аннотации типа
hello_msg = "Hello"

# input - выводит сообщение и читает ввод из консоли
name = input("Enter your name:")

# print - вывод в консоль переданных строк
print(hello_msg, name)
```

Типы

`bool` - булев тип данных

`int` - целочисленный

`float` - число с плавающей запятой

`str` - строковый тип данных

`list` - список объектов

`set` - перечисление объектов.

`tuple` - Кортеж – неизменяемый тип данных

`dict` - контейнерный объект (Словарь)

`bytes` - бинарный тип данных

Операции с числовыми типами int, float

```
# суммы x и y
x + y

# вычитание
x - y

# умножение x на y
x * y

# деление
x / y

# целочисленное деление x на y
x // y

# остаток от деления x / y
x % y

# x отрицается
-x

# x без изменений
+x

# возведение x в y
x ** y

# абсолютное значение или величина x
abs(x)

# конвертирование в int
int(x)

# конвертирование в floating
float(x)

# x в степень y
pow(x, y)
```

Условные операторы

```
# создание целочисленной переменной
number = 12
# проверка значения переменной
if (number > 10 ):
```

```
print(f"Число {number} больше 10")
else if(number==10):
    print(f"Число {number} равно 10")
else:
    print(f"Число {number} меньше 10")
```

the ternary operator is a conditional statement that resolves to one of two expressions. This means that a ternary operation statement can be assigned to a variable. In comparison, an if statement provides conditional branching of program flow but cannot be assigned to a variable.

Циклы while, for

Цикл while

```
# создание целочисленной переменной "number"
number = 1
# цикл выполняется пока переменная number не будет < 10
while number < 10:
    print(f"number = {number}")
    number += 1
# в цикле можно использовать else
else:
    print(f"number = {number}. Цикл завершен")
print("Работа программы завершена")
```

Цикл for

```
# создание переменной без аннотации типа"
message = "Hello"
# переборка строки и вывод каждого значения
for i in message:
    print(i)
```

Задание

5

```
# Сделать калькулятор с функциями +,-,/,*"
# Добавить выбор количества введенных чисел используя то что изучили в лекции
```

4

```
# Сделать калькулятор с функциями +, -, /, *
```

3

```
# Повторить все как в файле "
```