# Типы данных

***int:*** целые числа

***bool:*** логические значения

***float:*** числа с плавающей точкой

***str:*** строки

***list:*** список

***tuple:*** кортеж

***dict:*** словарь

***set:*** множество

***NoneType:*** значение None

| Тип данных | Примеры |
| --- | --- |
| *int* | 1, 42, -10 |
| *bool* | False, True |
| *float* | 3.14, 6.67e-14 |
| *str* | ‘hello’, “world” |
| *list* | [1, False, 1, “hello”] |
| *tuple* | (-10, 3.14, True) |
| *dict* | {‘key\_1’: ‘value\_1, ‘key\_2’: ‘value\_2’} |
| *set* | {‘hello’, 42, False, ‘False’} |

# Базовые операции

| Тип | Операция | Синтаксис |
| --- | --- | --- |
| *int* | сложение | *a+b* |
| вычитание | *a-b* |
| умножение | *a\*b* |
| целочисленное деление | *a // b* |
| остаток от деления | *a % b* |
| возведение в степень | *a \*\* b* |
| *float* | сложение | *a + b* |
| вычитание | *a - b* |
| умножение | *a \* b* |
| деление | *a / b* |
| возведение в степень | *a \*\* b* |
| *bool* | логическое НЕ | *not a* |
| логическое ИЛИ | *a or b* |
| логическое И | *a and b* |
| *str* | конкатенация | *str\_1 + str\_2* |
| дублирование | *str \* n* |

# Встроенные функции

Объекты данных

| *id* | идентификатор объекта |
| --- | --- |
| *type* | тип (класс) объекта |

Числа

| *round* | округление чисел |
| --- | --- |
| *abs* | абсолютное значение |
| *divmod* | целочисленное деление и остаток от деления |
| *bin* | двоичное представление числа |
| *oct* | восьмеричное представление |
| *hex* | шестнадцатеричное представление |
| *pow* | возведение в степень |

Символы и строки

| *chr* | возвращает символ по коду |
| --- | --- |
| *ord* | возвращает код символа |
| *len* | длина строки |

# Встроенные функции

Булевы значения

| *all* | логическое И всех аргументов |
| --- | --- |
| *any* | логическое ИЛИ всех аргументов |

Последовательности

| *len* | длина последовательности |
| --- | --- |
| *sum* | сумма элементов списка (кортежа) |
| *min* | минимальный элемент |
| *max* | максимальный элемент |
| *sorted* | сортировка списка |

Ввод и вывод текстовой информации

| *print* | вывод данных |
| --- | --- |
| *input* | ввод данных |
| *open* | открытие файла |

# Методы базовых типов

Строки

| *upper* | изменение на верхний регистр |
| --- | --- |
| *lower* | изменение на нижний регистр |
| *capitalize* | первый символ в верхнем регистре |
| *split* | разделение строки |
| *strip* | удаление пробелов в начале и в конце |
| *join* | объединение строк |
| *replace* | замена подстроки на другую |
| *count* | количество вхождений подстроки |
| *index* | индекс первого вхождения строки |
| *isalpha/isdigit* | строка состоит только из букв (цифр) |

Словари

| *get* | возвращение значения по ключу |
| --- | --- |
| *keys* | список ключей |
| *values* | список значений |
| *items* | список пар «ключ — значение» |

# Методы базовых типов

Списки

| *append* | добавить элемент в конец списка |
| --- | --- |
| *extend* | объединить списки |
| *insert* | вставить элемент с заданным индексом |
| *remove* | удалить элемент списка |
| *pop* | удалить элемент списка и вернуть его |
| *reverse* | изменить порядок элементов |
| *sort* | сортировка списка |
| *copy* | копировать список |

Множества

| *add* | добавить элемент во множество |
| --- | --- |
| *remove* | удалить элемент из множества |
| *union* | объединение множеств |
| *intersection* | пересечение множеств |
| *difference* | разность множеств |

# Работа с файлами

| Режим | Комментарий |
| --- | --- |
| *‘r’* | открыть файл только для чтения |
| *‘w’* | открыть файл только для записи, предварительно очистить его |
| *‘a’* | открыть файл только для записи в конец файла |
| *‘+’* | обновление файла (чтение и запись) |
| *‘x’* | создать файл, если его не существует |
| *‘t’* | открыть файл в текстовом виде |
| *‘b’* | открыть файл в бинарном виде |

Запись в файл

***write:*** принимает только строковый аргумент

***print:*** работает как стандартная функция вывода на экран

Чтение файла

| read(N) | прочитать весь файл или N символов |
| --- | --- |
| *readline* | прочитать одну строку |
| *readlines* | возвращает список строк |

# Основные синтаксические конструкции

def function(arg):

# тело функции

if condition\_1:

# блок кода, если condition\_1 == True

elif condition\_2:

# блок кода, если condition\_2 == True

else:

# блок кода, если оба условия False

for element in array:

# блок кода для каждого element:

while condition:

# выполнить блок кода, пока condition == True

with open(...) as file:

# менеджер контекста для работы с файлами