

with — оператор, который открывает "контекст".

Контекст — это область, которую открывает оператор, в ней создается экземпляр переданного в with класса и для него настроена особая обработка входа, выхода и опционально ошибок.

open() — это функция, которая возвращает нам класс IO (intensive operations), который является интерфейсом для работы с файлом.

```
1  print("Hello World!")
2
3  # for i in range(3):
4  fp = open("/path/to/file.txt", "r")
5  try:
6      some_func(fp)
7  except Exception as ex:
8      pprint(ex)
9  finally:
10     fp.close()
11
12  with open("/path/to/file.txt", "r") as fp:
13     content: list[str] = [x.strip() for x in fp.readlines()]
14
15  print(fp)  # Error — переменная fp существовала только в контексте with
16  print(content)
17  # > ["Hello!", "World!"]
18
19
20  from pathlib import Path
21
22  stra = ""
23  for i in range(333):
24     stra += i
25
26  with open(Path.home() / "large_number.txt", "w") as fp:
27     fp.write(stra)
28
29
```

csv — comma separated values /
данные разделенные запятыми

Основной формат для хранения данных
в табличном виде

```
1 name,age,city
2 Max,25,Moscow
3 Dias,47,Almaty
```

name	age	city
Max	25	Moscow
Dias	47	Almaty

json - JavaScript Object Notation

Формат текстовых данных для хранения данных в формате ключ: значение.

Может применяться как для хранения табличных данных, так и объектных

```
1  json_a = {"table": [  
2      {  
3          "name": "Max",  
4          "age": 25,  
5          "city": "Moscow"  
6      },  
7      {  
8          "name": "Dias",  
9          "age": 47,  
10         "city": "Almaty"  
11     }  
12 ]}  
13  
14 json_b = {"array": ["Max",  
    "Dias"]}
```

Number. - int, float

String. - str()

Boolean. - bool()

Array. - list()

Object. - dict()

Whitespace. - hz

Null. - None

name	age	city
Max	25	Moscow
Dias	47	Almaty

pickle — это модуль python, который позволяет хранить объекты python в файле в бинарном виде

Далее, читая этот файл, можно получить объекты python даже тех классов, которые вы написали сами.

Обратите внимание, нужно использовать бинарные версии модов функции open!

Для этого просто добавляйте букву b

```
1  import pickle
2
3  with open(file_path, "rb") as
   fp:
4      data = pickle.load(fp)
5
6  from dataclasses import
   dataclass
7
8  @dataclass
9  class Dog:
10     name: str
11     age: int
12
13     dogs = [Dog("bobik", 3),
14             Dog("sharik"), 5]
15
16     with open(file_path, "wb") as
        fp:
17         data = pickle.dump(dogs, fp)
```

В плане синтаксиса библиотека используется 1-в-1 как json