

Skillbox

# **Класс как контекст-менеджер. Методы enter и exit**

**Роман Булгаков**

Спикер

# Работа с файлом

```
try:
    file = open('hello.txt', 'w')
    file.write('Privet')
except Exception:
    raise
finally:
    file.close()
```

```
with open('hello.txt', 'w') as file:
    file.write('Privet')
```

Контекст-менеджер

**Контекст-менеджер** — это объект Python, который следит за инициализацией и финализацией данного контекста, в частности **определяет действия, которые должны происходить до и после выполнения блока кода.**

# Применение контекст-менеджеров

1. Синхронизация доступа к общим ресурсам
2. Настройка среды выполнения
3. Управление подключениями к базе данных
4. Работа с временными файлами
5. Обёртка соединений по протоколу и т. д.

```
class SomeProtocol:
    def __init__(self, host, port):
        self.host, self.port = host, port

    def __enter__(self):
        self._client = socket()
        self._client.connect((self.host, self.port))
        return self

    def __exit__(self, ex_type, ex_value, ex_traceback):
        self._client.close()

    def send(self, payload):...

    def receive(self):...

with SomeProtocol(host, port) as protocol:
    protocol.send(['get', signal])
    result = protocol.receive()
```

Пример использования  
(обёртка по протоколу)

Skillbox

**Спасибо  
за внимание!**