

Одно из применений множественного наследование – расширение функциональности класса каким-то заранее определенным способом. Например, если нам понадобится логировать какую-то информацию при обращении к методам класса.

Рассмотрим класс **Loggable**:

```
import time

class Loggable:
    def log(self, msg):
        print(str(time.ctime()) + ": " + str(msg))
```

У него есть ровно один метод **log**, который позволяет выводить в лог (в данном случае в stdout) какое-то сообщение, добавляя при этом текущее время.

Реализуйте класс **LoggableList**, отнаследовав его от классов **list** и **Loggable** таким образом, чтобы при добавлении элемента в список посредством метода **append** в лог отправлялось сообщение, состоящее из только что добавленного элемента.

Примечание

Ваша программа не должна содержать класс **Loggable**. При проверке вашей программе будет доступен этот класс, и он будет содержать метод **log**, описанный выше.