

## Лабораторная работа 12. Создание и использование абстрактного метода

Пример класса с абстрактным методом.

```
import abc
class Class1(abc.ABC):
    def __init__(self, val):
        self.x = val
    @abc.abstractmethod # Абстрактный метод
    def func(self):
        raise NotImplementedError("Нельзя вызывать абстрактный метод")

class Class2(Class1): # Наследуем абстрактный метод
    def another_func(self): # Определяем другой метод
        print(-self.x)

class Class3(Class2): # Наследуем два метода
    def func(self): # Переопределяем абстрактный метод
        print(self.x)

try: # Перехватываем исключения
    c = Class1(10) # Ошибка. Метод func() не переопределен
except TypeError as msg:
    print(msg) # вывод: Can't instantiate abstract class Class1 with abstract ...

try: # Перехватываем исключения
    c = Class2(10) # Ошибка. Метод func() не переопределен
except TypeError as msg:
    print(msg) # вывод: Can't instantiate abstract class Class1 with abstract ...

c = Class3(30)
c.func() # вывод: 30
c.another_func() # вывод: -30
```