

че — заготовленный. Список `sys.argv` в Python содержит *аргументы командной строки* — он является списком строк, которые отражают слова, набранные в командной строке, причем первым элементом всегда будет имя выполняемого сценария. Мы применяли аргументы командной строки в качестве переключателей в инструменте оценочных испытаний из главы 21, но здесь они используется как общий механизм ввода:

```
#!/python
"""
Файл: formats.py (Python 2.X и 3.X)
Разнообразные специализированные функции для форматирования строк
с целью отображения.
Тестирование можно производить с помощью заготовленного теста
или аргументов командной строки.
Что сделать: добавить круглые скобки для отрицательных денежных сумм,
реализовать больше возможностей.
"""
def commas(N):
    """
    Форматирует положительное целое число N для отображения
    с запятыми, разделяющими группы цифр: "xxx,yyy,zzz".
    """
    digits = str(N)
    assert digits.isdigit()
    result = ''
    while digits:
        digits, last3 = digits[:-3], digits[-3:]
        result = (last3 + ',' + result) if result else last3
    return result

def money(N, numwidth=0, currency='$'):
    """
    Форматирует число N для отображения с запятыми, 2 десятичными цифрами,
    ведущим символом $ и знаком, а также необязательным дополнением:
    "$ -xxx,yyy.zz".
    numwidth=0 - отсутствие дополнения пробелами,
    currency='' - опустить символ $
    или символ не ASCII для других валют (например, фунт - u'\xA3' или u'\u00A3').
    """
    sign = '-' if N < 0 else ''
    N = abs(N)
    whole = commas(int(N))
    fract = ('%.2f' % N)[-2:]
    number = '%s%s.%s' % (sign, whole, fract)
    return '%s*s' % (currency, numwidth, number)

if __name__ == '__main__':
    def selftest():
        tests = 0, 1 # не проходит: -1, 1.23
        tests += 12, 123, 1234, 12345, 123456, 1234567
        tests += 2 ** 32, 2 ** 100
        for test in tests:
            print(commas(test))
        print('')
        tests = 0, 1, -1, 1.23, 1., 1.2, 3.14159
        tests += 12.34, 12.344, 12.345, 12.346
        tests += 2 ** 32, (2 ** 32 + .2345)
        tests += 1.2345, 1.2, 0.2345
```