

# APPENDIX

# A

## 附录 A 部分练习参考答案

关于生命、宇宙和一切，这个宏大问题的答案……就是……42，Deep Thought 带着无限的威严和平静说道。

——道格拉斯·亚当斯，1979 年 10 月

（出自 *The Hitchhiker's Guide to the Galaxy*, 1979, Pan Books）

### 第 1 章

#### 正则表达式

##### 1-1 匹配字符串

bat、hat、bit 等

`[bh][aiu]t`

##### 1-2 姓+名

`[A-Za-z-]+ [A-Za-z-]+`

（任何一对由单个空格分隔的单词，即姓和名、连字符都可以）

##### 1-3 名+姓

`[A-Za-z-]+, [A-Za-z-]`

（任何由一个逗号和单空格分隔的单词和单个字母，例如姓氏的首字母）

```
[A-Za-z-]+, [A-Za-z-]+
```

(任何一对由一个逗号和单个空格分隔的单词, 例如姓、名、连字符都可以)

## 1-8 Python 长整型

```
\d+[1L]
```

(仅仅十进制整数)

## 1-9 Python 浮点数

```
[0-9]+(\.[0-9]*)?
```

(描述了一个简单的浮点数, 也就是说, 任何数量的数字后可以选择性地跟随一个小数点及零个或多个数字, 如 “0.004”、“2”、“75.” 等。)

## 第 2 章

### 2-3 套接字

TCP

### 2-6 Daytime 服务

```
>>>import socket
>>> socket.getservbyname('daytime', 'udp')
13
```

## 第 3 章

### 3-20 标识符

**pass** 是一个关键字, 所以它不能用作标识符。所有这种情况下, 常见的习惯是向变量名字中附加一条下划线 ( )。

## 第 4 章

### 4-2 Python 线程

I/O 密集型。为什么?

## 第 5 章

### 5-1 客户端/服务器架构

窗体客户端是 GUI 事件, 这些事件通常由用户生成, 并且必须由充当服务器的窗体系统处理; 它负责及时更新显示给用户的内容。

## 第 6 章

### 6-1 扩展 Python

- 性能改进
- 保护源代码
- 功能的新变化或期望的变化
- 更多

## 第 7 章

### 7-16 提高字幕幻灯片的设计（部分答案）

代码中的主要问题是它为演讲标题和个别幻灯片标题都调用了 `str.title()`。第 43 行确实需要改进。

```
s.Shapes[0].TextFrame.TextRange.Text = line.title()
```

我们可以做出快速改变，以使得代码仅仅将标题大小写应用于标题（目前在所有大写字母中），而不管无标题幻灯片的标题。

```
s.Shapes[0].TextFrame.TextRange.Text = title and  
line.title() or line
```

然而，我们可以做得更好。这个练习涉及如何处理 TCP/IP，以及如何避免改变它为“TCP / IP”。假设我们定义一个新的变量 `eachWord`。我的建议是检查 `eachWord == eachWord.upper()` 是否成立。如果它是一个缩写，那么就别管它；否则，我们就可以应用标题大小写。是的，虽然也有例外，但如果我们覆盖了 80%，那么目前对我们来说就已经足够好了。

## 第 8 章

### 8-1 DB-API

DB-API 是一个针对所有 Python 数据库适配器的常见接口规范。它是很不错的，因为它强制所有的适配器编写者对相同的规范进行编码，以便最终用户程序员能够编写一致的代码，这些代码可以以最少的工作量更加容易地移植到其他数据库中。

## 第 10 章

### 10-6 CGI 错误

Web 服务器要么不会返回数据，要么会返回错误文本，这将在你的浏览器中导致一个 HTTP 500 或内部服务器错误，因为这些返回的数据不是一个有效的 HTTP 或 HTML 数据。`cgitb` 模块捕获 Python 错误消息与追溯，并将其作为有效数据通过 CGI 返回且显示给用户一个强大的调试工具。

## 第 13 章

### 13-8 Web 服务和 csv 模块

用下面的代码替换 stock.py 中的 for 循环。

```
import csv
for tick, price, chg, per in csv.reader(f):
    print tick.ljust(7), ('%.2f' % round(float(price),
2)).rjust(6), chg.rjust(6), per.rjust(6)
```

## 第 14 章

### 14-2 CSV 与 str.split()

显然，当解析使用不同分隔符的数据时，逗号真的并不是可使用的最好分隔符，这一点不言而喻。除此之外，如果字段（单个“列”值）可以包含引号，那么使用逗号也充满危险。此外，如果字段可以包含引号，也会引发其他问题，这不仅仅因为字符串中可以包含逗号，还因为解析包含逗号的内容时，逗号可以出现在一个引号的左边或右边。

引号本身会导致问题——当你希望引号字符串中的所有单词被当作单个实体时，你将如何解析一个包含引号字符串的字符串？（提示：查看 <http://docs.python.org/library/shlex>）

### 14-11 稳健性

在 xmlrpcsrvr.py 文件中，修改第 13 行为如下内容。

```
FUNCS = ('add', 'sub', 'mul', 'div', 'mod')
```

添加所有需要的函数：

```
FUNCS = ('add', 'sub', 'mul', 'div', 'mod',
        'gt', 'ge', 'lt', 'le', 'eq', 'ne',
        truediv, 'floordiv',
    )
```

在 operator 模块中，这些都是可用的，所以除了这些变化之外，不需要额外的工作。现在你应该有足够的知识，并能够增加一元操作符~、\*\*，以及接位操作符&、|、^和~。

## 第 15 章

### 15-1 Jython

Jython 是（大部分）标准 Python 解释器的 Java 实现，标准 Python 解释器是用 C 语言编写的，因此它的另一个名字为 CPython。Jython 是字节编译的，以运行于 Java 虚拟机（JVM）上。并非一个直接移植的产物，Jython 的创建者认识到 Java 有它自己的内存管理和异常处理框架，所以这些语言特性并不需要移植。Jython 版本以特定的兼容性进行编号，也就是说，Jython 2.5 兼容 CPython 2.5。最初版本的 Jython 命名为 JPython，但它后来被 Jython 取代。可以在 <http://wiki.python.org/jython/JythonFaq/GeneralInfo> 上的 Jython 在线 FAQ 找到更多信息。