

在开发 Python 应用程序的过程中，经常需要将一些数据处理并保存成不同的文件格式，例如 Office、PDF 和 CSV 等文件格式。在本章的内容中，将详细讲解在 Python 第三方库将数据处理成特殊文件格式的知识，为读者步入本书后面知识的学习打下基础。

7.1 使用 Tablib 模块

在 Python 程序中，可以使用第三方模块 Tablib 将数据导出为各种不同的格式，包括 Excel、JSON、HTML、Yaml、CSV 和 TSV 等格式。在使用之前需要先安装 Tablib，安装命令如下所示。

```
pip install tablib
```

在接下来的内容中，将详细讲解使用 Tablib 模块的知识。

7.1.1 基本用法

1. 创建 Dataset (数据集)

在 Tablib 模块中，使用 `tablib.Dataset` 创建一个简单的数据集对象实例：

```
data = tablib.Dataset()
```

接下来就可以填充数据集对和数据。

2. 添加 Rows (行)

假如我们想收集一个简单的人名列表，首先看下面的实现代码：

```
#名称的集合
names = ['Kenneth Reitz', 'Bessie Monke']

for name in names:
    #分割名称
    fname, lname = name.split()

    # 将名称添加到数据集
    data.append([fname, lname])
```

在 Python 中我们可以通过下面的代码获取人名：

```
>>> data.dict
[('Kenneth', 'Reitz'), ('Bessie', 'Monke')]
```

3. 添加 Headers (标题)

7.1.2 操作数据集中的指定行和列

在下面的实例文件 `Tablib01.py` 中，演示了使用 Tablib 模块操作操作数据集中的指定行和列的过程。