

# 二进制数据服务

本章介绍的模块提供了一些操作二进制数据的基本服务操作。有关二进制数据的其他操作，特别是与文件格式和网络协议有关的操作，将在相关章节中介绍。

下面描述的一些库 [文本处理服务](#) 也可以使用 ASCII 兼容的二进制格式（例如 [re](#)）或所有二进制数据（例如 [difflib](#)）。

另外，请参阅 Python 的内置二进制数据类型的文档 [二进制序列类型 --- bytes, bytearray, memoryview](#)。

- [struct](#) --- 将字节串解读为打包的二进制数据
  - [函数和异常](#)
  - [格式字符串](#)
    - [字节顺序，大小和对齐方式](#)
    - [格式字符](#)
    - [例子](#)
  - [应用](#)
    - [原生格式](#)
    - [标准格式](#)
  - [类](#)
- [codecs](#) --- 编解码器注册和相关基类
  - [编解码器基类](#)
    - [错误处理方案](#)
    - [无状态的编码和解码](#)
    - [增量式的编码和解码](#)
      - [IncrementalEncoder 对象](#)
      - [IncrementalDecoder 对象](#)
    - [流式的编码和解码](#)
      - [StreamWriter 对象](#)
      - [StreamReader 对象](#)
      - [StreamReaderWriter 对象](#)
      - [StreamRecoder 对象](#)
  - [编码格式与 Unicode](#)
  - [标准编码](#)
  - [Python 专属的编码格式](#)
    - [文字编码](#)
    - [二进制转换](#)
    - [独立编解码器函数](#)
    - [文字转换](#)
  - [encodings](#) --- 编码格式包

- `encodings.idna` --- 应用程序中的国际化域名
- `encodings.mbc`s --- Windows ANSI代码页
- `encodings.utf_8_sig` --- 带BOM签名的UTF-8编解码器