

导入模块

本章中介绍的模块提供了导入其他Python模块和挂钩以自定义导入过程的新方法。

本章描述的完整模块列表如下：

- [zipimport --- 从 Zip 归档导入模块](#)
 - [zipimporter 对象](#)
 - [例子](#)
- [pkgutil --- 包扩展工具](#)
- [modulefinder --- 查找脚本使用的模块](#)
 - [ModuleFinder 的示例用法](#)
- [runpy --- 查找并执行 Python 模块](#)
- [importlib --- import 的实现](#)
 - [概述](#)
 - [函数](#)
 - [importlib.abc —— 关于导入的抽象基类](#)
 - [importlib.machinery —— 导入器和路径钩子函数。](#)
 - [importlib.util —— 导入器的工具程序代码](#)
 - [例子](#)
 - [用编程方式导入](#)
 - [检查某模块可否导入。](#)
 - [直接导入源码文件。](#)
 - [实现延迟导入](#)
 - [导入器的配置](#)
 - [importlib.import_module\(\) 的近似实现](#)
- [importlib.resources -- 包资源的读取、打开和访问](#)
 - [函数式 API](#)
- [importlib.resources.abc -- 资源的抽象基类](#)
- [importlib.metadata -- 访问软件包元数据](#)
 - [概述](#)
 - [函数式 API](#)
 - [入口点](#)
 - [分发的元数据](#)
 - [分发包的版本](#)
 - [分发包的文件](#)
 - [分发包的依赖](#)
 - [将导入映射到分发包](#)
 - [分发包对象](#)
 - [分发包的发现](#)
 - [实现自定义 Provider](#)

■ 示例

- `sys.path` 模块搜索路径的初始化
 - 虚拟环境
 - `_pth` 文件
 - 嵌入式 Python