

# 调试和分析

这些库可以帮助你进行 Python 开发：调试器使你能够逐步执行代码，分析堆栈帧并设置中断点等等，性能分析器可以运行代码并为你提供执行时间的详细数据，使你能够找出你的程序中的瓶颈。审计事件提供运行时行为的可见性，如果没有此工具则需要进行侵入式调试或修补。

- [审计事件表](#)
- [bdb --- 调试器框架](#)
- [faulthandler --- 转储 Python 回溯信息](#)
  - 转储跟踪信息
  - 对 C 栈进行转储
    - C 栈兼容性Stack Compatibility
  - 故障处理程序的状态
  - 一定时间后转储跟踪数据。
  - 转储用户信号的跟踪信息。
  - 文件描述符相关话题
  - 示例
- [pdb --- Python 的调试器](#)
  - 调试器命令
- [Python 性能分析器](#)
  - 性能分析器简介
  - 实时用户手册
  - `profile` 和 `cProfile` 模块参考
  - `Stats` 类
  - 什么是确定性性能分析？
  - 局限性
  - 准确估量
  - 使用自定义计时器
- [timeit --- 测量小代码片段的执行时间](#)
  - 基本示例
  - Python 接口
  - 命令行接口
  - 例子
- [trace --- 跟踪或记录 Python 语句的执行](#)
  - 命令行用法
    - 主要的可选参数
    - 修饰器
    - 过滤器
  - 编程接口
- [tracemalloc --- 跟踪内存分配](#)
  - 例子

- 显示前10项
- 计算差异
- 获取一个内存块的溯源
- 美化的 top
  - 记录所有被追踪内存块的当前和峰值大小
- API
  - 函数
  - 域过滤器
  - 过滤器
  - 帧
  - 快照
  - 统计
  - StatisticDiff
  - 跟踪
  - 回溯