

命令行入门与详解

什么是命令行

- Kernel与Shell
- 广义的命令行包括一切通过字符终端控制计算机的方式
 - Windows – cmd/PowerShell/Git bash
 - UNIX/Linux系列: sh/zsh/Terminal, etc.

为什么需要命令行

- 你不得不用
 - 因为所有公司的服务器都运行在Linux上
- 将工作自动化
 - 自动化是一切生产力的根源
 - 举个栗子
- 相比GUI，命令行更容易开发维护
 - 说白了就是bug更少
- 远程连接时命令行占用资源远低于GUI
- 命令行上的开发者工具更丰富

命令行的历史与流派

- UNIX家族
 - POSIX标准
 - macOS
 - Linux
 - Windows Subsystem for Linux
- Windows
 - 奇葩

命令行的启动

- Windows: cmd/PowerShell/WSL
- UNIX/Linux: 一开机就是, 不用找
 - macOS: 终端

命令的全部要素

- 可执行程序 (Executable)
 - 参数
 - 环境变量 (Environment variable)
 - 工作目录 (Working directory)
-
- 以上四个要素相同，就可以完全地“重现”一个命令
 - 你碰到的各种各样古怪的问题，原因一定是上述四个要素之一

命令的组成部分

- 工作目录
 - 启动命令的路径
 - 相对路径都是相对于这个路径
 - 在Java中的应用

环境变量

- 进程 (Process)
 - 进程是计算机程序运行的最小单位
 - 独占自己的内存空间和文件资源
- 每个进程都和一组变量相绑定
 - 传递不同的环境变量可以让程序表现出不同的行为
 - CLASSPATH/GOPATH
- 在进程的fork过程中，环境变量可以被完全继承
- 所有的操作系统/编程语言都支持环境变量
- 局部和全局的环境变量

环境变量实战

- 通过export/set设置环境变量，通过echo读取环境变量
- 从Java/Go/Python/Node.js中读取环境变量
- 向Docker容器传递环境变量
- 快速传递一个环境变量
- 使用环境变量控制程序的行为

可执行程序

- Windows: exe/bat/com
- UNIX/Linux: x权限 – 可执行权限
- 去哪里找程序?
 - Windows: Path环境变量 + 当前目录
 - UNIX/Linux: PATH环境变量
- 在脚本的第一行指定解释器 (shebang)
- 别名 (alias)

参数

- UNIX参数的约定——然鹅Java并不鸟这个约定
- 参数中包含空格或者字符串
 - 单引号
 - 双引号
- 参数的展开

输入与输出

- 标准输入 stdin
- 标准输出 stdout
- 标准错误 stderr
- 输出的重定向
 - 覆盖文件
 - 追加文件
 - 改变流向
 - /dev/null

掌握原理之后
需要命令的时候再去搜就好了

逗你玩呢！

Linux常用命令

- 进入目录 `cd` (change directory)
- 展示文件 `ls/l/ls -alth` (list)
- 移动/重命名文件 `mv` (move)
- 删除文件 `rm/rm -r/rm -rf` (remove)
- 拷贝文件 `cp/cp -r` (copy)
- 显示变量 `echo`
- 导出变量 `export`
- Git系列操作 `git pull/push/add/commit/...`

Linux常用命令

- 新建目录 `mkdir/mkdir -p (make directory)`
- 当前目录 `pwd (print working directory)`
- 显示文件 `cat (concatenate and print files)`
- 编辑文件 `vi`

我要怎样才能掌握？

- 自己对着书敲命令
- 尽可能地在开发中使用命令行
 - vi/git/mkdir, etc.
- 使用命令行的各种小工具
- 开发命令行的程序
- 自行在云服务器上开发部署博客
- 自行在云服务器上进行持续集成实战
- 日常使用Linux进行开发

Java与命令行

- Java世界里的一切工具都只做一件事：拼接命令行
- 使用命令行编译、运行Java程序
- Java程序中的环境变量与系统属性详解
- 使用Java程序启动新的进程

Q&A