# 常用服务器配置

视频文件地址：

<http://10.41.95.141:8080/files/>

部门电子图书馆：ftp://1\_\_\_0.41.95.141/BookShare/

大神视频：\_\_https:/\_/www.bilibili.com/video/BV1Za411t7c6/?spm\_id\_from=pageDriver

英：[h\_ttps://w\_ww.bilibili.com/video/BV1KS4y1G7ku/?spm\_id\_from=333.999.0.0&vd\_source=257c7b6ef7b40a65034a320ae8772ad6](https://www.bilibili.com/video/BV1KS4y1G7ku/?spm_id_from=333.999.0.0&vd_source=257c7b6ef7b40a65034a320ae8772ad6%20)

## FTP服务器：

ftp://10.41.95.141

账号：sw

密码sw\_rw

## gitlab服务器：

10.41.95.141

docker-compose.yml地址：/home/mailserver/gitlab/

## gitlab-runner：

10.41.95.85,

10.41.95.78,

10.41.95.128,

10.41.53.93

## harbor服务器：

10.41.95.93

账号：admin

密码：1234qwer!@#$QWER

# 工作会用到的工具：

系统环境：Windows，Linux

编译环境：Python

文档语法：MD

编译工具：VSCode

部署环境：docker，未来可能会有k8s

镜像仓库：harbor

代码仓库：gitlab，svn

数据库：postgresql

缓存：redis

消息队列：kafka

代理服务器：Nginx

server端：uwsgi+django/flask、fastApi，celery

<http://10.34.64.152:8090/user/sign_up>

client：Vue2/3，各种图表插件，3D渲染，UI组件

## Python 面向对象的解释型计算机程序设计语言

规范：https://www.python.org/dev/peps/pep-0008/

任务：

https://www.runoob.com/python/python-100-examples.html

https://hub.yzuu.cf/Yixiaohan/show-me-the-code

## git：开源的分布式版本控制系统

部门规范：http://10.41.95.141/esrz30/git-commint-specification.git

教程：

http://10.37.37.139/apps/armstrong3/present/as06/video.html

http://10.37.37.139/apps/armstrong3/present/as06/videoB.html

## gitlab：用于仓库管理系统的开源项目

安装：http://10.37.37.139/apps/armstrong3/present/as05/video.html（还有harbor）

教程：

http://10.37.37.139/apps/armstrong3/present/as07A/video.html

http://10.37.37.139/apps/armstrong3/present/as07B/video.html

ftp://10.41.95.141/%E9%A1%B9%E7%9B%AE%E7%B3%BB%E7%BB%9F%E7%BD%91%E7%AB%99%E7%8E%AF%E5%A2%83%E6%90%AD%E5%BB%BA/ gitlab CI CD?�署 - 2022�?????10.01.40.mp4

## Django官网**开放 源代码 的 Web应用框架**

2.X：https://docs.djangoproject.com/zh-hans/2.2/

3.X：<https://docs.djangoproject.com/zh-hans/3.2/>

## Flask中文网：****是一个使用 Python 编写的轻量级 Web 应用框架****

https://flask.net.cn/

## FastApi官网：****是****一个现代的，快速（高性能）python web框架

<https://fastapi.tiangolo.com/>

## Vue官网：Vue.js****是****一款流行的JavaScript前端框架

2.X：https://cn.vuejs.org/index.html

3.X：<https://v3.cn.vuejs.org/>

**VueX：** **vuex是**基于vue框架的一个状态管理库

官网：https://vuex.vuejs.org/zh/guide/

教程：<ftp://10.41.95.141/Training/maple/vuex%A4%C0%A8%C9/>

## Postgresql：对象-关系型数据库管理系统

10：http://www.postgres.cn/docs/10/index.html

13：<http://www.postgres.cn/docs/13/index.html>

Ansible ：<https://github.com/vitabaks/postgresql_cluster>

Patroni：[简介 — Patroni 2.1.4 文档](https://patroni.readthedocs.io/en/latest/)

Pgpool：<https://www.pgpool.net/docs/pgpool-II-3.5.4/doc/pgpool-zh_cn.html>

Keepalived：[https://github.com/francs/PostgreSQL-Keepalived-HA](https://github.com/francs/PostgreSQL-Keepalived-HA" \t "_blank)

## docker：开源的应用容器引擎

官网：https://docs.docker.com/desktop/

教程：

http://10.37.37.139/apps/armstrong3/present/as01A/video.html

http://10.37.37.139/apps/armstrong3/present/as01B/video.html

http://10.37.37.139/apps/armstrong3/present/as01C/video.html

## Nginx：****Nginx是****一款自由的、开源的、高性能的HTTP服务器和反向代理服务器

http://nginx.org/en/docs/

## redis：基于内存实现的键值型非关系（NoSQL）数据库

http://www.redis.cn/documentation.html

# 基础知识点：

数据结构

1、栈与队列的区别，使用场景

2、链表（单链表，双链表，循环双链表，散列表（hash表+链表））与数组与hash表的区别，使用场景

3、常见算法：KMP算法（子串匹配）滑动窗口，合并区间，动态规划，贪心算法

4、常用排序算法（冒泡排序，简单选择排序，插入排序，快速排序（希尔排序），归并排序，堆排序，计数排序）

5、树的遍历算法，前序遍历，中序遍历（深度优先遍历），后序遍历，广度优先遍历（层序遍历）

6、图寻道算法：最小生成树算法（Prim普里姆算法，Kruskal克鲁斯卡尔算法），洪水算法

操作系统+计算机组成与原理

1、多级存储系统（寄存器->cache（缓存）->主存（磁盘）->辅存（外设））

2、内存替换算法：先来先服务，短作业优先，lru（最近少用）

3、进程与线程与协程的区别于应用场景

4、程式是如何在计算机里面运行的

计算机网络

1、http协议的理解

2、tcp三次握手

3、路由表跳转

4、各种协议常用端口，80,21,22,433

5、网关，网卡，路由理解：ftp://10.41.95.141/%E9%A1%B9%E7%9B%AE%E7%B3%BB%E7%BB%9F%E7%BD%91%E7%AB%99%E7%8E%AF%E5%A2%83%E6%90%AD%E5%BB%BA/%E8%AE%A1%E7%AE%97%E6%9C%BA%E7%BD%91%E7%BB%9C-ipv4%E5%92%8CDNS/

数据库

1、SQL增删改查

2、连表查询

3、视图应用

测试

1、黑盒测试（功能测试，ab测试，回归测试，边缘测试，等价测试）

2、白盒测试(基本路径测试）

3、单元测试

4、性能测试（压力测试，疲劳测试，冒烟测试）

原型制作+UI设计：

1、

语言（Python）：

1、面向对象三个特征

2、基本库和第三方库使用

3、数据类型有哪些

# 进阶知识点：

数据结构：

跳表，b树，b+树，b-树，红黑树（剪枝）

操作系统：

OOM（内存泄露，内存溢出），内存池，缓冲池

计算机网络：

网络拥塞追踪，ICMP协议理解，ARP协议理解，SOAP协议理解，DNS服务器，

SSL：<ftp://10.41.95.141/%E9%A1%B9%E7%9B%AE%E7%B3%BB%E7%BB%9F%E7%BD%91%E7%AB%99%E7%8E%AF%E5%A2%83%E6%90%AD%E5%BB%BA/SSL/>

数据库：

负载均衡

查询优化（SQL优化，读写分离，分库分表，负载均衡）

事务处理机制

数据回滚

数据日志捕捉

触发器

缓存：

数据一致性

底层原理

代理服务器：

IP拦截，限流，扩容（负载均衡）

Nginx：

负载均衡

反向代理

正向代理

IP拦截，限流，扩容（负载均衡）

语言（Python）：

1、迭代器

2、垃圾回收机制

3、实例方法，静态方法，类方法区别，应用场景

4、预留内存池

5、分配内存机制

6、列表扩容

7、server，epoll、poll三者区别

软件工程：

1、各种UML绘制（用例图，时序图，类图，流程图，ER图……）

2、各种开发方法（瀑布，螺旋，V型，敏捷）

3、23种设计模式