

Linux

1.在登录Linux时，一个具有唯一进程ID号的shell将被调用，这个ID是什么(b)

A.NID B.PID C.UID D.CID
process ID 进程id号
UID 用户id号 ,如何查看?查看的是当前登录的用户信息

2.下面那个用户存放用户密码信息(b)

A./boot B./etc C./var D./dev

df -h #查看linux磁盘空间剩余量的
ls -lh #显示文件夹详细信息和大小的
du -h #显示当前目录 所有文件大小信息
du -sh . #显示当前目录,总大小
man du #查看du命令的帮助信息
du --help #显示帮助信息

3.用于自动补全功能时，输入命令或文件的前1个或后几个字母按什么键(b)

A.ctrl B.tab C.alt D.esc

4.vim退出不保存的命令是(a)

A.:q B.q C.:wq D.:q!

5.文件权限读、写、执行三种符号的标志依次是(a)

A.rwx B.xrw C.rdx D.rws

r 4
w 2
x 1
对应的数字权限转化

- rw- r-- r-- 1 root root 0 Apr 20 17:31 s16.py

- 普通文本的类型
rw- users root用户读写不可执行
r-- 属组 root组内的成员 只读

r-- 其他人 只读

6.某文件的组外成员的权限是只读、属主是全部权限、组内权限是可读可写、该文件权限为(d)

A.467 B.674 C.476 D.764

#others r-- 4 users u rwx 4+2+1=7 group r+w 6

7.改变文件的属主的命令是(c)

A.chmod B.touch C.chown D.cat

chmod #修改文件权限的
touch #创建普通根本
chown change owner #更改用户者,更改属主
chgrp #更改属组

8.解压缩文件mydjango.tar.gz, 我们可以用(a)

A.tar -zxvf mydjango.tar.gz
B.tar -xvz mydjango.tar.gz
C.tar -czf mydjango.tar.gz
D.tar - xvf mydjango.tar.gz

.gz
.tar
都可以通多tar命令解压
tar -x 解压缩tar后缀的
tar -z 解压缩 gz后缀的
tar -f 必须制定一个压缩文件名 ,这个-f参数,必须写在最后面
tar -v 显示压缩/解压缩过程

9.检查linux是否安装了,可用哪些命令(b) #注意rpm -qi只能查询用yum安装的软件, 编译的查不到

A.rpm -ivh nginx
B.rpm -q nginx
C.rpm -U nginx
D.rpm -x nginx

linux的软件包格式都是rpm的
rpm也是linux提供的一个命令,用于查询通过yum安装的软件
rpm -i #是安装软件
rpm -q query #查询软件
rpm -qi #详细信息
rpm -ql #详细安装路径

10.Linux配置文件一般放在什么目录(a)

A.etc B.bin C.lib D.dev

#bin(可执行文件)

11.linux中查看内存，交换内存的情况命令是(c) #free -m

A.top B.last c.free D.lastcomm

#top - linux任务管理器,查看系统负载状态,动态进程信息的

#last - 用户最后登录的信息

12.观察系统动态进程的命令是(b)

A.free B.top C.lastcomm D.df

13.如果执行命令，chmod 746 file.txt，那么该文件的权限是(a)

A.rwxr-rw-

B.rw-r-r-

C.-xr-rwx

D.rwxr-r-

14.找出当前目录以及其子目录所有扩展名为“.txt”的文件，那么命令是(d)

A.ls .txt

B.find /opt -name “.txt”

C.ls -d .txt

d.find -name “.txt”

15.什么命令常用于检测网络主机是否可达? c

A.ssh B.netstat C.ping D.exit

#ssh - 远程登录

#netstat - 查看linux端口信息的

#ping D.exit(退出会话)

16.退出交互式shell，应该输入什么? d

A:q! B.quit C.; D.exit

#创建文件夹,如果父目录不存在,怎么办? 通过小写的 -p参数,递归创建父目录

mkdir -p

17.在父目录不存在的时候，添加的参数是? d

A.-P B.-d C.-f D.-p

18.下列文件中，包含了主机名到IP地址映射关系的文件是? b

- A. /etc/hostname #主机名
- B. /etc/hosts #强制写入域名解析记录
- C. /etc/resolv.conf 存放dns服务器的文件
- D. /etc/networks

19.请问你使用的linux发行版是什么？如何查看linux发行版信息？

```
centos7
cat /etc/os-release
```

20.请问你公司的服务器环境是物理机还是虚拟化？

物理服务器 dell power R720 型号

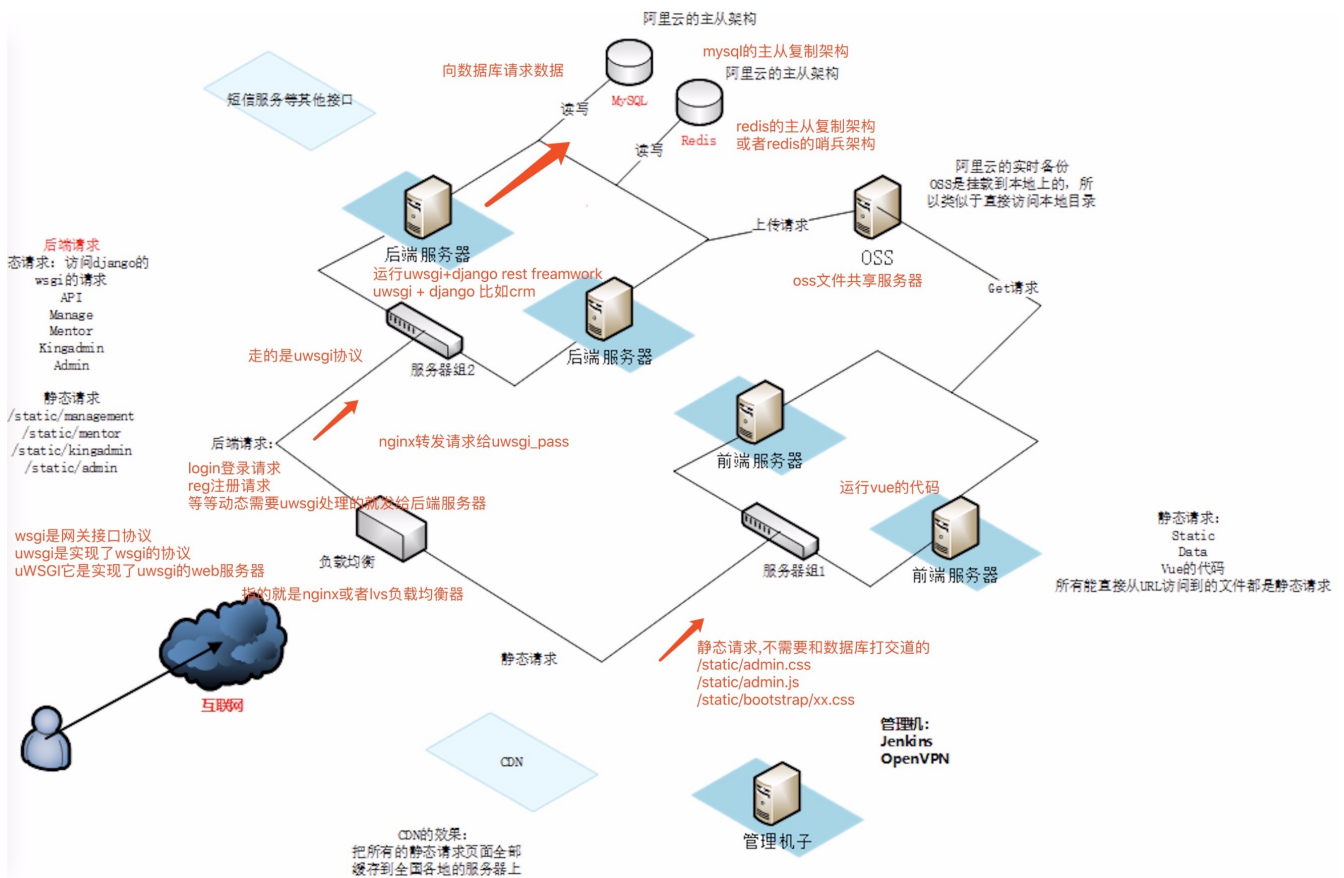
- 单独安装一个操作系统
- 通过虚拟化技术,在一台机器上,安装N个操作系统
 - vmware虚拟化技术
 - 个人学习版:vmware workstation
 - 企业版虚拟化 是 vmware vsphere ,安装处一个esxi虚拟化服务器

云服务器:一个路飞学城的业务,就是8台阿里云,数据库用的也都是阿里的rds数据库,提供主从复制,高可用等功能

路飞学城, 线上服务器环境, 8台阿里云服务器, 配置信息如下

实例ID/名称	标签	监控	可用区	IP地址	状态	配置	付费方式	操作
I-2zefbha9mr8fun1hjnra test-backend.luf...			北京 可用区 A	192.168.0.11(私有)	运行中	1 vCPU 4 GB (I/O优化) ecs.mn4.small 1Mbps (峰值) 服务器配置	包年包月 2021年11月13日 00:00 到期	管理 远程连接 升降配 续费 更多
I-2zea7wyzv7b7k27q3q4x 测试服务器, 测试运行uwsgi+数据库			北京 可用区 A	192.168.0.12(私有)	运行中	1 vCPU 1 GB (I/O优化) ecs.xn4.small 0Mbps	包年包月 2021年11月12日 00:00 到期	管理 远程连接 升降配 续费 更多
I-2ze7s23bovr5bbukp47 线上服务器, 前端机器02 front-web02.luff...			北京 可用区 A	192.168.0.13(私有)	运行中	2 vCPU 8 GB (I/O优化) ecs.sn2.medium 0Mbps	包年包月 2021年11月11日 00:00 到期	管理 远程连接 升降配 续费 更多
I-2zehy7gffoks0s97a8e8 线上服务器, uwsgi后端服务器02 backend-web02.luff...			北京 可用区 A	192.168.0.14(私有)	运行中	4 vCPU 8 GB (I/O优化) ecs.sn1.large 2Mbps	包年包月 2021年11月11日 00:00 到期	管理 远程连接 升降配 续费 更多
I-2zeatuq0koxs0tgdtdwj 管理中心, 内网管理课程平台服务器环境 manage01			北京 可用区 A	192.168.0.15(私有)	运行中	2 vCPU 4 GB (I/O优化) ecs.n4.large 1Mbps	包年包月 2021年11月24日 00:00 到期	管理 远程连接 升降配 续费 更多
I-2ze0wcse9h32ghwyc6z 线上服务器, 前端机器01 front-web01.luff...			北京 可用区 A	192.168.0.16(私有)	运行中	2 vCPU 8 GB (I/O优化) ecs.sn2.medium 0Mbps	包年包月 2021年11月23日 00:00 到期	管理 远程连接 升降配 续费 更多
I-2ze0lwbuz0pku139t23 线上服务器, 后端机器02 backend-web01.luff...			北京 可用区 A	192.168.0.17(私有)	运行中	4 vCPU 8 GB (I/O优化) ecs.sn1.large 5Mbps	包年包月 2021年11月16日 00:00 到期	管理 远程连接 升降配 续费 更多

第8台服务器: 运行mysql主从复制+redis主从复制



21.vim有几种工作模式

命令模式
编辑模式
底线命令模式

22.nginx的主配置文件是?如何实现多虚拟主机?nginx反向代理参数是?

nginx.conf
多个server{
proxy_pass http://ip

23.如何解压缩后缀是.gz文件?

gzip -d *.gz

24.如何解压缩后缀是.tar文件?

tar -xf .tar

25.如何解压缩后缀是.xz文件?

xz -d .xz

26.www服务在internet最为广泛，采用的结构是?

27.如何给linux添加dns服务器记录?

```
/etc/resolv.conf  
添加2条主备dns记录  
nameserver dns服务器ip  
  
28.每月的5,15,25的晚上5点50重启nginx  
crontab -e  
50 17 5,15,25 /usr/bin/systemctl restart nginx  
50 17 5,15,25 /opt/nginx112/sbin/nginx -s reload
```

29.每分钟清空/tmp/内容

```
/usr/bin/rm -rf /tmp/*  
30.每天早上6.30清空/tmp/的内容  
30 6 /usr/bin/rm -rf /tmp/  
  
31.每个星期三的下午6点和8点的第5到15分钟之间备份mysql数据到/opt/  
5-15 18,20 3 /usr/bin/cp -r /var/lib/mysql /opt/
```

32.某文件权限是drw-r—rw-, 请解读该权限?

d:目录文件
rw- 属主: 可读可写
r- 属组: 可读
rw- other: 可读可写

33.centos版本系统服务管理命令是?

```
service  
systemctl
```

34.如何远程登录阿里云123.206.16.61?

```
ssh root@123.206.16.61
```

35.备份mariadb的命令是?

```
mysqldump -uroot -p
```

36.简述特殊符号的含义?

root用户的身份提示符
重启定向覆盖写

重定向追加写
\$PATH 取值符
· 当前目录
.. 上级目录

37.如果你发现在公司无法使用rm, 使用提示'禁止你使用rm', 是什么原因?
别名alias

38.如何修改test.py属组为alex?

```
chgrp alex test.py
```

39.如何在windows和linux传输文件? 有哪些方法?

```
xftp lrzsz scp
```

40.如何杀死mariadb进程?

```
kill mariadb  
  
ps -ef|grep mysql  
kill pid  
killall mariadb
```

41.简述dns解析流程? 访问www.pythonav.cn的解析流程

自上而下的顺序
1. 优先查找本地dns缓存
2. 查找本地/etc/hosts文件, 是否有强制解析
3. 如果没有去/etc/resolv.conf指定的dns服务器中查找记录(需联网)
4. 在dns服务器中找到解析记录后, 在本地dns中添加缓存
5. 完成一次dns解析

42.linux如何安装软件?有几种方式?

```
yum  
rpm  
源码包
```

43.出于安全角度, 简述如何安装启动redis服务端?

```
更改端口  
开启protomode yes安全模式  
设置redis密码  
redis-server redis.conf
```

44.如何保证本地测试环境和线上开发环境一致性? 思路?

- 1.docker打包镜像
- 2.手动解决环境问题 pip3 freeze导出依赖

45.virtualenv是什么?简述如何使用

在开发Python应用程序的时候，系统安装的Python3只有一个版本：3.4。所有第三方的包都会被pip安装到Python3的site-packages目录下。

如果我们要同时开发多个应用程序，那这些应用程序都会共用一个Python，就是安装在系统的Python 3。如果应用A需要jinja 2.7，而应用B需要jinja 2.6怎么办？

这种情况下，每个应用可能需要各自拥有一套“独立”的Python运行环境。virtualenv就是用来为一个应用创建一套“隔离”的Python运行环境。

- 1.安装 `pip3 install virtualenv`
- 2.创建虚拟环境 `virtualenv --no-site-packages --python=python3 env1`
- 3.激活虚拟环境 `source /opt/MyVirtualenv/venvDjango1/bin/activate`
- 4.测试 `python3` 或者 `pip3 list`

46.virtualevnrwrapper是什么? 简述使用

virtualenv 的一个最大的缺点就是，每次开启虚拟环境之前要去虚拟环境所在目录下的 bin 目录下 source 一下 activate，这就需要我们记住每个虚拟环境所在的目录。

Virtualenvwrapper是virtualenv的扩展包，用于更方便管理虚拟环

- 1.安装虚拟环境 `pip3 install virtualenvwrapper`
- 2.创建并进入虚拟环境 `mkvirtualenv env1`
- 3.切换虚拟环境 `workon 虚拟环境名`

redis是什么?

Redis是一个开源的基于内存的，key-value数据结构的缓存数据库，支持数据持久化，m-s复制，常用数据类型有string set hash list，

最佳应用场景：适用于数据变化快且数据库大小可遇见（适合内存容量）的应用程序。

例如：股票价格、数据分析、实时数据搜集、实时通讯。

Redis只能使用单线程，性能受限于CPU性能，故单实例CPU最高才可能达到5-6wQPS每秒（取决于数据结构，数据大小以及服务器硬件性能，日常环境中QPS高峰大约在1-2w左右）

其他nosql数据库？

Memcached可以利用多核优势，单实例吞吐量极高，可以达到几十万QPS（取决于key、value的字节大小以及服务器硬件性能，日常环境中QPS高峰大约在4-6w左右）。适用于最大程度扛量。

只支持简单的key/value数据结构，不像Redis可以支持丰富的数据类型。

无法进行持久化，数据不能备份，只能用于缓存使用，且重启后数据全部丢失。

MongoDB

更高的写负载，MongoDB拥有更高的插入速度，支持高可用性，支持索引高速查询，占用磁盘空间较大，支持持久化

47.redis哨兵是什么? 作用是

Redis-Sentinel是Redis官方推荐的高可用性(HA)解决方案

redis哨兵是监控redis主从服务，不存储数据的，作用是用于自动切换redis服务主从关系，即当主库服务停止后，会将其中一个从库变为主库

48.redis-cluster是什么?

#即使使用哨兵，redis每个实例也是全量数据存储，每个redis存储的内容都是完整的数据。

#为了最大化利用内存，可以采用cluster集群 -->分布式存储，即每台redis存储不同的内容。

采用redis-cluster架构正是满足这种分布式存储要求的集群的一种体现。

redis-cluster架构中，被设计成共有16384个hash slot。

每个master分得一部分slot，其算法为： $\text{hash_slot} = \text{crc16}(\text{key}) \bmod 16384$ ，这就找到对应slot。

采用hash slot的算法，实际上是解决了redis-cluster架构下，有多个master节点的时候，数据如何分布到这些节点上去。

key是可用key，如果有{}则取{}内的作为可用key，否则整个可以是可用key。群集至少需要3主3从，且每个实例使用不同的配置文件。

49.什么是静态资源，什么是动态资源?

静态资源指定的是网站的CSS/JS/HTML文件，放在磁盘上固定不动。

动态资源一般指的是数据，即后端给前端提供的数据

50.配置linux软连接的命令?

```
ln -s 目标文件名 软连接名
```

51.如何永久添加/opt/python36/的环境变量?

```
vim /etc/profile          #全局变量配置文件，每次登录都会加载这个文件
添加PATH = /opt/python36/bin:
source /etc/profile
```

52.给如下代码添加注释

```
server{          # 一个虚拟主机
    listen 80;    # 监听的端口，访问的端口80
    server_name 192.168.11.11; # 访问的域名192.168.11.11
    location / { # 访问的路径 /
        root html; # 指定页面的目录，访问/会找到html目录
        index index.html # 指定网页，访问/就是访问index.html
    }
}

server{          #虚拟主机
    listen 8080;  #nginx监听端口
    server_name 192.168.11.11; #nginx访问域名
    location / {  #location匹配url
        include uwsgi_params; #将uwsgi参数添加进nginx
        uwsgi_pass 0.0.0.0:8000; #反向代理转发请求给uwsgi
    }
}
```

53.supervisor是什么? 如何使用?

#后台管理进程任务的 -->你本来自己手动敲的命令,现在交给supervisor去管理

使用:

1.安装 `easy_install supervisor`

2.生成配置文件 `echo_supervisord_conf > /etc/supervisor.conf`

3.写入自定义的配置

[program:crm] ; 项目名称

`command=/root/Envs/knight/bin/uwsgi -ini /opt/knight/uwsgi.ini` ;启动项目的命令

`stopasgroup=true` ;默认为false,进程被杀死时,是否向这个进程组发送stop信号,包括子进程

`killasgroup=true` ;默认为false,向进程组发送kill信号,包括子进程

4.启动supervisor服务

`supervisord -c /etc/supervisor.conf`

5.启动所有项目

`supervisorctl -c /etc/supervisor.conf start all`

54.简述项目部署流程? 如何部署路飞, uwsgi+nginx+supervisor+nginx

部署路飞:

1.安装python3 环境

2.安装 mysql, redis, nginx

3.部署前端

1.安装node.js的环境

2.安装依赖包 `npm install`

3.修改axios的发送的端口接口 `axios.get('后台接口的地址')`

4.打包

4.部署后端

1.安装virtualenv

2.创建虚拟环境

3.安装django和uwsgi, 以及项目的依赖包

4.修改uwsgi的配置文件

5.通过uwsgi -ini 配置文件启动django项目

5.配置nginx

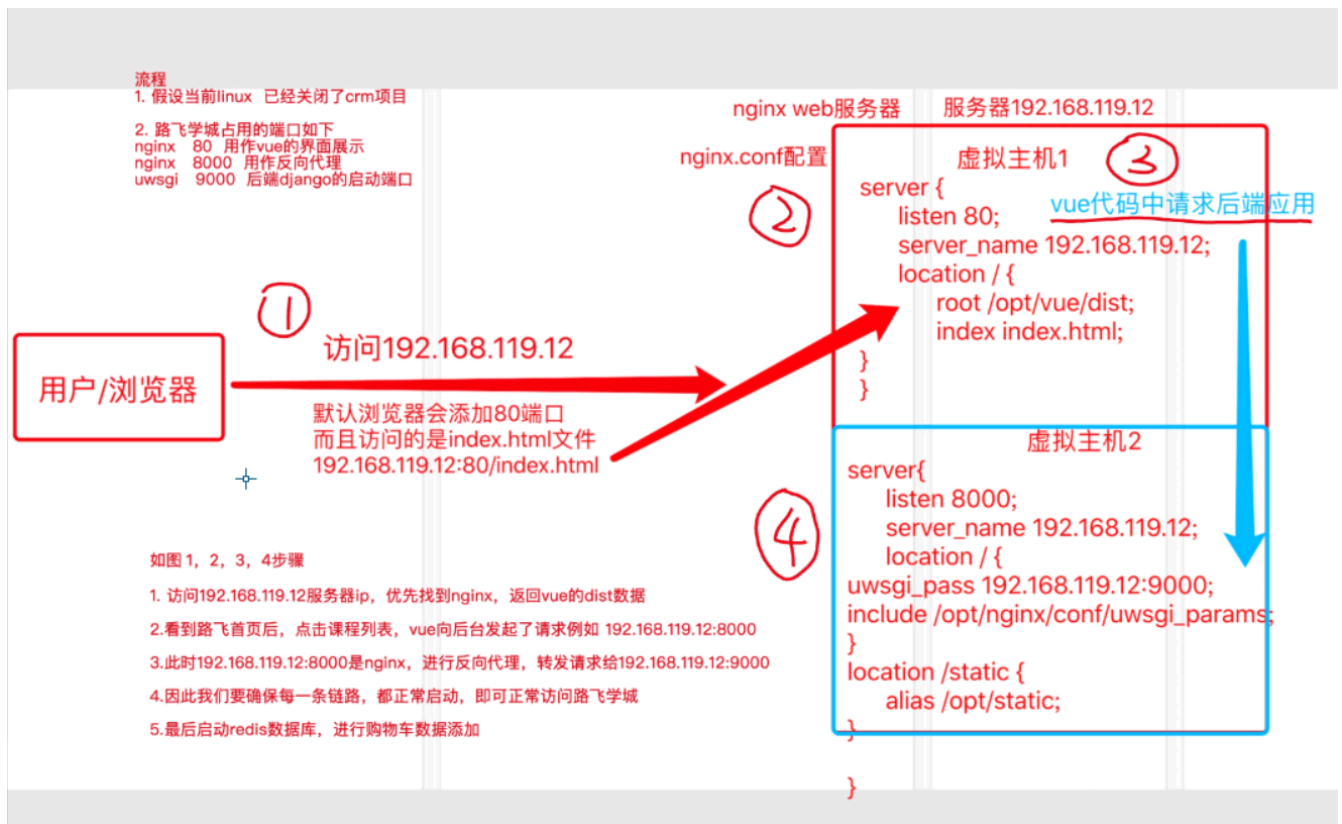
1.创建两个虚拟主机, 分别监听80和8000端口

2.访问80端口是访问vue

3.访问8000端口是vue发起的8000端口请求, 反向代理到8000的uwsgi

6.启动nginx,mysql,redis

7.通过supervisor来管理



55.docker是什么？简述docker优势

linux容器软件
docker应用于快速构建应用

56.你常用的docker常用命令有哪些？操作镜像、容器、仓库的命令

docker images # 查看本地镜像
docker search 镜像 # 通过docker hub搜索镜像
docker rmi 镜像 # 删除镜像
docker save 镜像 > 路径 # 导出镜像
docker load < 路径 # 导入镜像
docker build -t . # 打包生成镜像,打包dockerfile生成docker镜像

----- 操作容器命令 -----

docker run -d 镜像 解释器 # 根据镜像生成容器, 后台允许
docker run -it 镜像 解释器 # 根据镜像生成并进入容器
docker start/stop 容器id # 启动/停止容器
docker ps # 查看当前运行的容器
docker rm 容器id # 删除容器
docker exec 容器id # 进入当前正在运行的容器
docker commit 容器id 镜像名 # 将容器提交为镜像
docker contain ls # 查看当前运行的容器

----- 操作仓库的命令 -----

docker pull 镜像 # 下载镜像
docker push 镜像 # 推送镜像

57.哪个命令无法查看linux文件内容？ d

A.tac B.more C.head D.man

58.使用rm -i 系统会提示什么信息？ b

- A. 命令所有参数
- B. 是否真的删除
- C. 是否有写的权限
- D. 文件的路径

59.为何说rm -rf 慎用？

-r递归删除
-f强制删除

60.python操作linux的模块是？

os

61.如果端口8080被占用，如何查看是什么进程？

```
netstat -tunlp | grep 8080
```

62.redis是如何做持久化的？

rdb

Redis会定期保存数据快照至一个rdb文件中，并在启动时自动加载rdb文件，恢复之前保存的数据，通过save指令触发持久化，redis单独开启一个子进程进行数据持久化。

rdb缺点，定期执行，可能会丢失数据，并且数据量特别大时候，如果服务器cpu性能较低，rdb开启子进程持久化性能影响很大，影响redis对外提供服务的能力。

aof

Redis会把每一个写请求都记录在一个日志文件里。在Redis重启时，会把AOF文件中记录的所有写操作顺序执行一遍，确保数据恢复到最新。

随着AOF不断地记录写操作日志，因为所有的操作都会记录，所以必定会出现一些无用的日志。大量无用的日志会让AOF文件过大，也会让数据恢复的时间过长。

优先：数据安全，不怕数据损坏，如断电灯问题，还可以用redis-check-aof修复数据，AOF文件人为可读

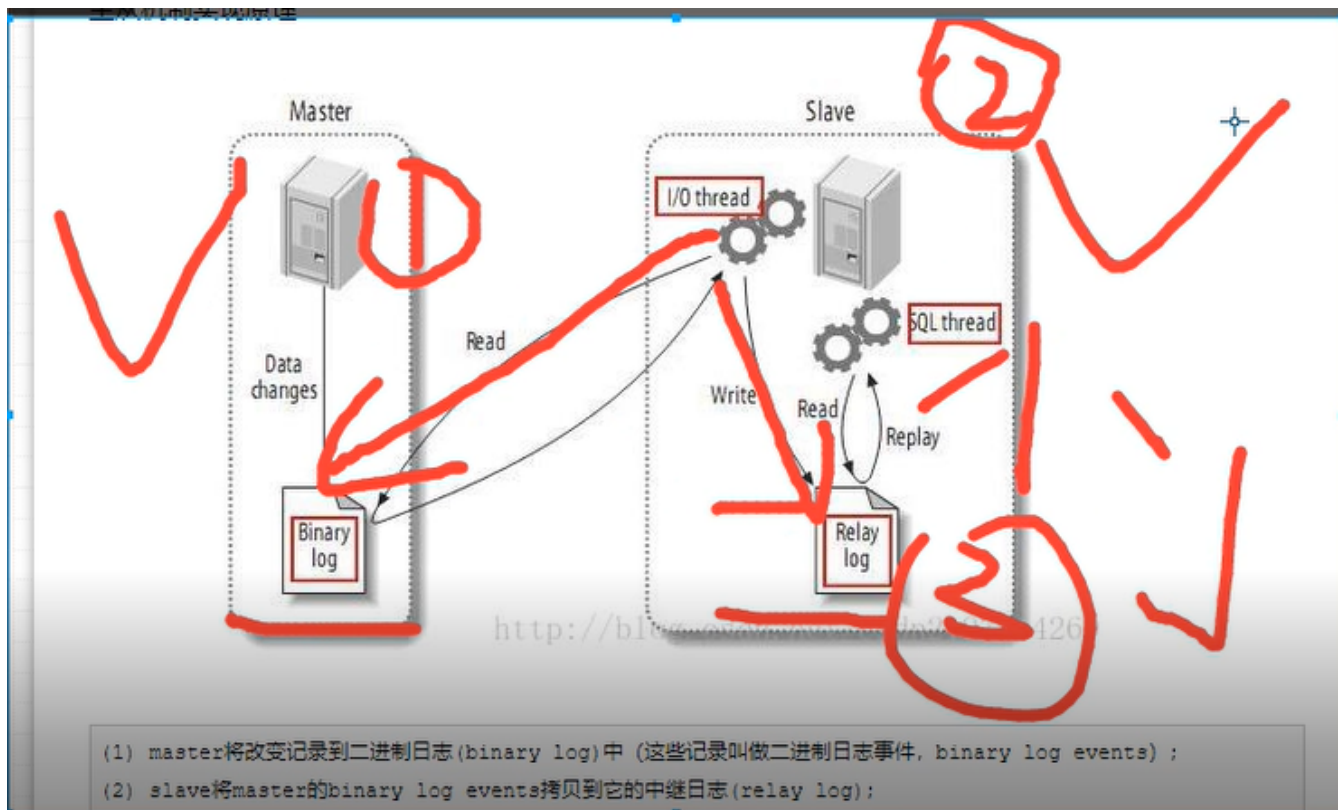
缺点：占磁盘，性能损耗高，数据恢复慢

怎么用rdb和aof

如果既配置了RDB，又配置了AOF，则在进行数据持久化的时候，都会进行，但是在根据文件恢复数据的时候，以AOF文件为准，RDB文件作废

63.简述mysql主从复制原理？

(1) master将改变记录到二进制日志(binary log)中（这些记录叫做二进制日志事件，binary log events）；(2) slave将master的二进制 log events拷贝到它的中继日志(relay log)；(3) slave重做中继日志中的事件，将改变反映它自己的数据。



64.创建mysql用户alex, 并且授予权限select权限, 命令是什么?

```
grant select on . to alex@'%';
```

65.nginx如何实现负载均衡?

```
upstream {}
```

66.nginx的负载均衡调度算法有几种? 是什么?

调度算法	概述
轮询	按时间顺序逐一分配到不同的后端服务器(默认)
weight	加权轮询,weight值越大,分配到的访问几率越高
ip_hash	每个请求按访问IP的hash结果分配,这样来自同一IP的固定访问一个后端服务器
url_hash	按照访问URL的hash结果来分配请求,是每个URL定向到同一个后端服务器
least_conn	最少链接数,那个机器链接数少就分发

67.linux下载软件包的方法有?

```
wget curl
```

68.windows和linux常用远程连接工具有哪些?

```
xshell  
putty  
securecr
```

69.如何给与一个脚本可执行权限

```
chmod u+x file
```

70.过滤出settings.py中所有的空白和注释行

```
grep -v "^#" file |grep -v "^$"
```

71.过滤出file1中以abc结尾的行

```
grep "abc$" file1
```

72.容器退出后，通过docker ps查看不到，数据会丢吗？

不会丢，因为容器停止了，并没有被删除 docker ps -a可以看到

73.如何批量清理后台停止的容器

```
docker rm $(docker ps -aq)
```

74.如何查看容器日志？

```
docker logs -f
```

75.wsgi是什么？

WSGI是Web服务器网关接口。它是一个协议，描述了Web服务器如何与Web应用程序通信。

76.Django中使用的是？

wsgiref 是个单socket模块，性能很低，用于测试django代码正常运行的

uwsgi 替代wsgiref

uwsgi --ini uwsgi.ini

uwsgi支持多进程，让你的django，并发性更好

用nginx结合uwsgi，nginx处理静态文件，uwsgi处理动态请求，实现网站的高并发。

Flask中实现wsgi的是：werkzeug

Tornado中实现wsgi的是：tornado和gevent

77.消息队列的作用？

- 1) 程序解耦
- 2) 数据冗余，例如rabbitmq的ack机制，消息确认机制
- 3) 削峰能力
- 4) 可恢复性，就算系统中部分组件挂掉，消息在队列也不丢失，待组件恢复后继续处理消息。
- 5) 异步通信，如发红包，短信等流程丢入队列，可以优先级很低的去处理。

git常用命令

```

1:git init-----初始化
2:git add .-----从工作区，添加到版本库
3:git commit -m"xxx"----从暂存区，添加到分支
4:git status-----查看状态
5:git log -----查看版本库的日志
6:git reflog-----查看所有日志
7:git reset -head 版本号--切换
8:git stash-----保存
9:git stash-----将第一个记录从“某个地方”重新拿到工作区（可能有冲突）
git stash list-----查看“某个地方”存储的所有记录
git stash clear-----清空“某个地方”
git stash pop-----将第一个记录从“某个地方”重新拿到工作区（可能有冲突）
git stash apply -----编号,将指定编号记录从“某个地方”重新拿到工作区（可能有冲突）
git stash drop -----编号，删除指定编号的记录
10:git branch dev----创建分支
11:git branch -d dev---删除分支
12:git checkout dev---切换分支
13:git merge dev-----合并分支
14:git branch-----查看所有分支
15:git clone https:xxx--克隆
16:git add origin https:xxx-起个别名
17:git push origin dev --添加到dev分支
18:git pull origin master-拉代码
19:git fetch origin master-去仓库获取
20:git merge origin/master-和网上下的master分支合并
# -----协同开发-----
默认是master分支-----master
开发的分支-----dev
做代码review-----review
程序员自己的分支-----.....
1: 每个员工创建自己的分支
2: 将自己的代码提交的到自己的分支-----xxx,sss,www.....
3: 由组长或老大做代码的review,-----代码提交的review分支
4: 再提交到dev.
5: 再合并到master分支

```

熟悉 Linux常用操作。

```

1: man rm-----查看命令帮助
2: mkdir-----创建目录
3: touch-----创建文件
4: cd-----切换。
5: ls-----查看目录
6: ls -lh-----查看目录详细
7: pwd-----查看当前目录
8: vim-----添加内容
9: echo-----追加内容
10: cat-----查看文件内容
11: mv-----移动
12: cp-----拷贝
13: mv-----重命名

```

15: find-----搜索
16: rm-----删除数据
17: ping-----查看能不能上网
19: tar cf -----打压缩
20: tar xf-----解压缩

#安装:

yum install

rpm包安装

编译安装

#快捷键:

1: Tab键-----自动补全命令或路劲。

2: ctrl+l-----清屏

3: ctrl+c-----取消当前操作

4: vi/vim 快捷键:

复制当前行 -----yy

粘贴-----p

剪切-----dd

撤销-----u

恢复