

数据库索引可以明显提高哪一操作的效率？

正确答案: A 你的答案: 空 (错误)

SELECT  
INSERT INTO ... VALUES ...  
UPDATE  
DELETE

数据库：以下哪种锁定方式能提供最佳的并行访问性能？

正确答案: D 你的答案: 空 (错误)

列锁定  
表锁定  
块锁定  
行锁定

从 DELETE 语句中省略 WHERE 子句，将产生什么结果？

正确答案: B 你的答案: 空 (错误)

DELETE 语句将失败因为没有记录可删除  
DELETE 语句将从表中删除所有的记录  
DELETE 语句将提示用户进入删除的标准  
DELETE 语句将失败，因为语法错误

Traceroute 一般使用的是哪种网络层协议？

正确答案: D 你的答案: 空 (错误)

vrrp  
udp  
arp  
icmp

ospf 协议中哪种 lsa 只能在本区域内传播？

正确答案: A 你的答案: 空 (错误)

2  
3  
5  
7

在 linux 系统中，下列哪些信号无法捕获？

正确答案: B 你的答案: 空 (错误)

SIGHUP  
SIGKILL

SIGQUIT  
SIGUSR1

Linux 下, 如何查看一个端口被什么进程占用?

正确答案: B 你的答案: 空 (错误)

```
netstat -an|grep 端口号  
netstat -tnlp | grep 端口号  
iostat -an | grep 端口号  
iostat -dxt | grep 端口号
```

列表如何去掉重复元素?

正确答案: B 你的答案: 空 (错误)

列表无法去重

先把 list 转换为一个去重的集合, 然后在 list 化

先把 list 转换为一个去重的元组, 然后在 list 化

列表不会有重复

Python 的列表(List)和元组(Tuple)区别是什么?

正确答案: A 你的答案: 空 (错误)

列表可变, 元组不可变

没有区别

限制不一样

列表可以被迭代, 元组无法迭代

关于 Python 类的继承正确的说法是?

正确答案: C 你的答案: 空 (错误)

python 类无法继承

可以继承但是, 无法执行父类的构造函数

可以有多个父类

只能有一个父类

以下关于端口的描述哪些是正确的?

正确答案: A B C D 你的答案: 空 (错误)

FTP 使用 TCP 20 端口

FTP 使用 TCP 21 端口

DNS 使用 TCP 53 端口

DNS 使用 UDP 53 端口

下面关于 http 协议中的 GET 和 POST 方式的区别, 哪些是错误的?

正确答案: A C 你的答案: 空 (错误)

他们都可以被收藏，以及缓存

GET 请求参数放在 URL 中

GET 只用于查询请求，不能用于请求数据

GET 不应该处理敏感数据的请求

一个简单的网站可能由这三部分组成：一个 web 服务器（提供网页和 http 服务），一个后台服务器（比如 java 后台程序，负责计算），一个 mysql 数据库服务器（存储用户数据）。然而我们知道，这样一个网站也是很脆弱的，上述三个服务器的任何一个出现问题（服务器问题，或者服务器的网络问题，或者上面运行的程序问题），都会导致网站访问不了。显然这样是无法符合美团大众点评给用户的服务承诺的。现在，假定你是公司的技术经理，你认为你会针对上述 3 个部分，做些什么样的加强或改善工作，以帮助提高网站的可用性呢？（尽可能多的列举）

#### 1、异地容灾：

在异地的备份可以接管业务。由于平时两地的数据同步，也不会造成用户数据丢失。

#### 2、双机备份

双机热备特指基于高可用系统中的两台服务器的热备（或高可用），因两机高可用在国内使用较多，故得名双机热备，双机高可用按工作中的切换方式分为：主-备方式（Active-Standby 方式）和双主机方式（Active-Active 方式），主-备方式即指的是一台服务器处于某种业务的激活状态（即 Active 状态），另一台服务器处于该业务的备用状态（即 Standby 状态）。而双主机方式即指两种不同业务分别在两台服务器上互为主备状态（即 Active-Standby 和 Standby-Active 状态）。

#### 3、基于共享存储（磁盘阵列）的方式

共享存储方式主要通过磁盘阵列提供切换后，对数据完整性和连续性的保障。用户数据一般会放在磁盘阵列上，当主机宕机后，备机继续从磁盘阵列上取得原有数据。

#### 4、磁盘阵列

从哪几个方面评价一个 hash 函数的好坏？列举几种常见的 hash 函数？

hash 函数好坏的评判标准 1.高效，节省 cpu，才能提高并发，作为中间层，需要高效的根据 key 来计算 hash 2.冲突尽可能的小，小到可以建立唯一索引 3.尽可能的节省空间。例如，要把这个结果存储到数据库中，在给这个 hash 后的结果建立索引，那么我们希望这个列越小越好，以便节省数据存储空间。特别是数据库中建立索引的时候，被索引的字段自然是越小越好 4.要均匀，特别是有多个节点的时候，保证每个 key 分布的均匀，比较重要，否则负载没法均衡 5.rehash 的时候，保证 key 的重新分布尽可能的小，避免给后端带来较大的冲击 常见 hash 函数 比如，md5, sha-1, crc16, crc32 等