

2020 年下半年 数据库系统工程师 （上午+下午）答案详解

如碰到错别字或者答案错误，请联系售后微信:ruankaopass 或者扫码

扫一扫，叫我微信号:ruankaopass



提供软考历年真题，视频

【1】B

【解析】

算术逻辑单元是运算器的重要组成部分，负责处理数据，实现对数据的算术运算和逻辑运算。

所以本题正确的选项是 B

CPU 中其它部件作用：

程序计数器是用于存放下一条指令所在 单元的地址的地方。在程序执行前，必须将程序的起始地址，即程序的一条指令所在的内存单元地址送入程序计数器，当执行指令时，CPU 将自动修改程序计数器的内容，即每执行一条指令程序计数器增加一个量， 使其指向下一个待指向的指令。程序的转移等操作也是通过该寄存器来实现的。

地址寄存器一般用来保存当前 CPU 所访问的内存单 元的地址，以方便对内存的读写操作。

累加器是专门存放算术或逻辑运算的一个操作数和运算结果的寄存器。

ALU 是 CPU 的执行单元，主要负责运算工作。

指令寄存器一般用来保存当前正在执行的一条指令。

数据寄存器主要是用来保存操作数和运算结果等信息的，其目的是为了节省读取操作数所需占用总线和访问存储器的时间。

地址寄存器一般用来保存 当前 CPU 所访问的内存单元的地址，以方便对内存的读写操作。

作为程序员，应该要能控制其所编写程序的执行过程，这就需要利用程序计数器来实现，因

此程序员能访问的是程序计数器。

【2】C

【解析】

本题考查计算机系统硬件基础知识。

基于成本和性能方面的考虑，Cache（即高速缓存）是为了解决相对较慢的主存与快速的 CPU 之间工作速度不匹配问题而引入的存储器。Cache 中存储的是主存内容的副本。

【3】D

【解析】

本题考查计算机系统硬件方面关于 DMA 方式的基础知识。

程序控制(查询)方式：CPU 需要不断查询 I/O 是否完成，因此一直占用 CPU。

程序中断方式：与程序控制方式相比，中断方式因为 CPU 无需等待而提高了传输请求的响应速度。

DMA 方式：DMA 方式是为了在主存与外设之间实现高速、批量数据交换而设置的。DMA 方式比程序控制方式与中断方式都高效。CPU 只负责初始化，不参与具体传输过程。

【4】D

【解析】

在 Windows 系统中，磁盘碎片整理程序可以分析本地卷，以及合并卷上的可用空间使其成为连续的空闲区域，从而使系统可以更高效地访问文件或文件夹。

【5】C

【解析】

本题考查数据结构方面的基础知识。

栈和队列都是操作受限的线性表，栈仅在表尾插入和删除元素，队列仅在表头删除元素、在表尾插入元素。

队列是先入先出的线性表，栈是后进先出的线性表。一个线性序列经过队列结构后只能得到与原始序列相同的元素序列，而经过一个栈结构后则可以得到多种元素序列

串是由零个或多个任意字符组成的有限序列。

【6】 B

【解析】

同【5】

【7】 D

【解析】

同【5】

【8】 A

【解析】

本题考查数据结构基础知识。

遍历运算是二叉树的基本运算，主要有先序、中序、后序和层序遍历。

先序遍历的基本方法:对于非空二叉树，先访问根结点，然后先序遍历根的左子树，最后先序遍历根的右子树。因此，若已知某二叉树的先序遍历序列，则可直接得到其树根结点。

中序遍历的基本方法:对于非空二叉树，先中序遍历根的左子树，然后访问根结点，最后中序遍历根的右子树。因此，若已知某二叉树的根结点，则可根据中序遍历序列将该二叉树左右子树上的结点划分开。

后序遍历的基本方法:对于非空二叉树，首先后序遍历根的左子树，接着后序遍历根的右子树，最后访问根结点。因此，若已知某二叉树的后序遍历序列，则可直接得到其树根结点。因此，按照遍历左子树要在遍历右子树之前进行的原则，根据访问()位置的不同，可得到二叉树的前序、中序和后序三种遍历方法。

【9】 C

【解析】

给定 N 个权值作为 N 个叶子结点，构造一棵二叉树，若该树的带权路径长度达到最小，称这样的二叉树为最优二叉树，也称为哈夫曼树。哈夫曼树是带权路径长度最短的树，权值较大的结点离根较近。

霍夫曼树可以用来进行通信电文的编码和解码。利用哈夫曼树求得的用于通信的二进制编码称为哈夫曼编码。树中从根到每个叶子节点都有一条路径，对路径上的各分支约定指向左

售后微信:ruankaopass
淘宝店: 软考真题教育

子树的分支表示“0”码，指向右子树的分支表示“1”码，取每条路径上的“0”或“1”的序列作为各个叶子节点对应的字符编码，即是哈夫曼编码。具有 n 个叶子节点的权值为 W_1, W_2, \dots, W_n 的最优二叉树的形态不是唯一的。

【10】B

【解析】

本题考查数据结构方面的基础知识。

线性表的查找有顺序查找、折半查找、分块查找方法。

其中，顺序查找方法的特点是算法非常简单，但效率较低，因为它是用所给关键字与线性表中各元素的关键字逐个比较，直到成功或失败。

折半查找方法的优点是比较次数少，查找速度快，平均性能好；其缺点是要求待查表为有序表，且插入和删除困难。因此，折半查找方法适用于不经常变动而查找频繁的有序列表。

分块查找方法又称索引查找，它主要用于“分块有序表的查找。所谓“分块有序是指将线性表 L （一维数组）分成 m 个子表（要求每个子表的长度相等），且第 $i+1$ 个子表中的每一个项目均大于第 i 个子表中的所有项目。“分块有序表应该包括线性表 L 本身和分块的索引表。因此，分块查找的关键在于建立索引表 I ，其查找的平均长度介于顺序查找和折半查找之间。

【11】D

【解析】

字典攻击：在破解密码或密钥时，逐一尝试用户自定义词典中的可能密码（单词或短语）的攻击方式。与暴力破解的区别是，暴力破解会逐一尝试所有可能的组合密码，而字典式攻击会使用一个预先定义好的单词列表（可能的密码）。

密码盐：在密码学中，是指通过在密码任意固定位置插入特定的字符串，让散列后的结果和使用原始密码的散列结果不相符，这种过程称之为“加盐”。如果密码盐泄露，黑客可以利用他们数据字典中的密码，加上泄露的密码盐，然后进行散列，然后再匹配，由于密码盐可以加在任意位置，也加大了破解的难度。所以即使密码盐泄露，字典攻击和不加盐时的效果是不一样的。

【12】D

【解析】

Hash,一般翻译做散列、杂凑,或音译为哈希,是把任意长度的输入通过散列算法变换成固定长度的输出,该输出就是散列值。这种转换是一种压缩映射,也就是,散列值的空间通常远小于输入的空间,不同的输入可能会散列成相同的输出,所以不可能从散列值来确定唯一的输入值。简单的说就是一种将任意长度的消息压缩到某一固定长度的消息摘要的函数。哈希表是根据键(Key)而直接访问在内存存储位置的数据结构。

在密码学里面,随机预言机(英语:Random oracle)是一部预言机,对任何输入都回传一个真正均匀随机的输出,不过对相同的输入,该预言机每次都会用同一方法输出。换句话说,随机预言机是一个将有可能输入与输出作随机映射的函数。

【13】C

【解析】

拒绝服务攻击即是攻击者想办法让目标机器停止提供服务,是黑客常用的攻击手段之一。

最常见的 DoS 攻击有计算机网络带宽攻击和连通性攻击。带宽攻击指以极大的通信量冲击网络,使得所有可用网络资源都被消耗殆尽,最后导致合法的用户请求无法通过。连通性攻击指用大量的连接请求冲击计算机,使得所有可用的操作系统资源都被消耗殆尽,最终计算机无法再处理合法用户的请求。

分布式拒绝服务攻击 DDOS 是一种基于 DoS 的特殊形式的拒绝服务攻击,是一种分布的、协同的大规模攻击方式。

拒绝服务攻击并不会造成用户密码的泄露。

【14】C

【解析】

社会工程攻击,是一种利用“社会工程学”来实施的网络攻击行为。在计算机科学中,社会工程学指的是通过与他人的合法地交流,来使其心理受到影响,做出某些动作或者是透露一些机密信息的方式。这通常被认为是一种欺诈他人以收集信息、行骗和入侵计算机系统的行为。所以,ABD 都符合社会攻击学的特点。

【15】C

【解析】

Linux 中，权限的格式：-rw-rw-rw-

(1) 第 0 位确定文件类型(说明：-:普通文件，d:目录，l: 连接文件，c:字符设备文件[键盘鼠标] b:块设备文件[硬盘])。

(2) 第 1-3 位确定所有者(该文件的所有者)拥有该文件的权限。R:读 ， w: 写权限 x:执行权限(-表示没有)。

(3) 第 4-6 位确定所属组(同用户组的)拥有该文件的权限。

(4) 第 7-9 位确定其他用户拥有该文件的权限。

【16】A

【解析】

计算机软件著作权的客体是指著作权法保护的计算机软件著作权的范围(受保护的對象)。

根据《著作权法》第三条和《计算机软件保护条例》第二条的规定，著作权法保护的计算机软件是指计算机程序及其有关文档。

根据《计算机软件保护条例》第三条第一款的规定，计算机程序是指为了得到某种结果而可以由计算机等具有信息处理能力的装置执行的代码化指令序列，或者可被自动转换成代码化指令序列的符号化语句序列。计算机程序包括源程序和目标程序，同一程序的源程序文本和目标程序文本视为同一软件作品。

根据《计算机软件保护条例》第三条第二款的规定， 计算机程序的文档是指用自然语言或者形式化语言所编写的文字资料和图表，以用来描述程序的内容、组成、设计、功能规格、开发情况、测试结果及使用方法等。文档一般以程序设计说明书、流程图、用户手册等表现。因此，答案为 A。

【17】C

【解析】

根据我国《著作权法》规定，人身权是指计算机软件著作权人基于自己的软件作品而享有的并与该作品有关的以人格利益为内容的专有权利，这种权利是基于计算机软件著作权人人身权的，并与人身不可分离、不能放弃或转让，并没有直接财产内容的权利。人身权包括发表权、署名权、修改权等。

①发表权。发表权是著作权人享有的决定是否将其作品发表以及如何发表的权利。所谓发表是指将作品的原件或复制件，公之于众。根据我国《计算机软件保护条例》的规定，中国公

民和单位开发的计算机软件不论是否发表，都享有著作权，并且不限制在何地发表。

②署名权。署名权即表明开发者身份的权利以及在软件上署名的权利，它不受时间的限制，也不因权利人的死亡或者消失而消灭。署名权主要包括 3 方面的内容：有权在自己开发的软件上署名；决定署名的方式；禁止他人在自己的软件作品上署名。

③修改权。修改权就是软件开发者自行或授权他人对软件进行增补、删节，或者改变指令、语句顺序等修改的权利。修改是创作的延续，其本身也是一种创作，无论作品是否发表，著作权人都享有修改权。

对于本试题，选项 C 的“署名权”是基于计算机软件著作权人人身的，并与人身不可分离、不能放弃或转让的。软考真题售后微信:ruankaopass

【18】D

【解析】

操作系统主要包括以下几个方面的功能：

①进程管理，其工作主要是进程调度，在单用户单任务的情况下，处理器仅为一个用户的一个任务所独占，进程管理的工作十分简单。但在多道程序或多用户的情况下，组织多个作业或任务时，就要解决处理器的调度、分配和回收等问题

②存储管理分为几种功能：存储分配、存储共享、存储保护、存储扩张。

③设备管理分有以下功能：设备分配、设备传输控制、设备独立性。

④文件管理：文件存储空间的管理、目录管理、文件操作管理、文件保护。

⑤作业管理是负责处理用户提交的任何要求。

【19】B

【解析】

长期调度，又称为作业调度或高级调度，这种调度将已进入系统并处于后备状态的作业按某种算法选择一个或一批，为其建立进程，并进入主机，当该作业执行完毕时，还负责回收系统资源，在批处理系统中，需要有作业调度的过程，以便将它们分批地装入内存，在分时系统和实时系统中，通常不需要长期调度。它的频率比较低，主要用来控制内存中进程的数量。

中期调度，又称为交换调度。它的核心思想是能将进程从内存或从 CPU 竞争中移出，从而降低多道程序设计的程度，之后进程能被重新调入内存，并从中断处继续执行，这种交换的操

作可以调整进程在内存中的存在数量和时机。其主要任务是按照给定的原则和策略，将处于外存交换区中的就绪状态或等待状态的进程调入内存，或把处于内存就绪状态或内存等待状态的进程交换到外存交换区。

短期调度，又称为进程调度、低级调度或微观调度。这也是通常所说的调度，一般情况下使用最多的就是短期调度。它的主要任务是按照某种策略和算法将处理机分配给一个处于就绪状态的进程，分为抢占式和非抢占式。

【20】C

【解析】

根据操作系统内核是否对线程可感知，可以把线程分为内核线程和用户线程。根据操作系统内核是否对线程可感知，可以把线程分为内核线程和用户线程。所以，线程可以实现在内核空间和用户空间。

【21】A

【解析】

进程观点把操作系统看作由若干个可以同时独立运行的程序和一个对这些程序进行协调的核心所组成，这些同时运行的程序称为进程。系统进行资源分配和调度的独立单位。

【22】C

【解析】

程序设计语言的基本成分包括数据、运算、控制和传输，数据是程序操作的对象，按照数据组织形式可以分为多种类型，其中枚举属于用户定义类型，数组属于构造类型。

【23】B

【解析】

同【22】

【24】C

【解析】

同【22】

【25】B

【解析】

本题考查计算机软件开发模型相关的基础知识。

软件开发模型是软件开发的全部过程、活动和任务的结构框架，用以指导软件的开发。螺旋模型综合了瀑布模型和演化模型的优点，并增加了风险分析，沿着螺线由内向外，每旋转一圈，就得到原型的一个新版本。

【26】A

【解析】

软件测试的目的是发现软件的错误，验证软件是否满足用户需求，并通过分析软件错误产生的原因，以帮助发现当前开发工作所采用的软件过程的缺陷，以便进行软件过程改进。软件测试不能发现软件中的所有错误，也不可能保证软件完全正确。

【27】B

【解析】

耦合程度从低到高的顺序为：非直接耦合、数据耦合、标记耦合、控制耦合、外部耦合、公共耦合和内容耦合。最好的是非直接耦合，没有直接联系，模块之间不相互依赖于对方。最差的是内容耦合，一个模块访问了另一个模块的内部数据

内聚程度从高到低的顺序为：功能内聚、顺序内聚、通信内聚、过程内聚、瞬时内聚、逻辑内聚和偶然内聚。

所以，在软件设计中通常用耦合度和内聚度作为衡量模块独立程度的标准，耦合程度最高的是内容耦合；内聚程度最高的是功能内聚。

【28】B

【解析】

同【27】

【29】A

【解析】

1989 年在贝尔实验室诞生的 ADSL 是 xDSL 家族成员中的一员，被营为现代信息高速公路上

的快车”。它因其下行速率高、频带宽、性能优等特点而深受广大客户的喜爱，成为继 MODEM、ISDN 之后的又一种全新更快捷，更高效的接入方式。它是运行在原有普通电话线上的一种新的高速宽带技术。事实上，ADSL 的传输技术中，ADSL 用其特有的调制解调硬件来连接现有双绞线连接的各端。软考真题售后微信:ruankaopass

【30】A

【解析】

同【29】

【31】A

【解析】

在 TCP/IP 协议栈中，应用层协议数据单元为消息或者报文。其中段是传输层，用户数据报是网络层，帧是数据链路层。物理层是比特流。

应用层—消息、报文(message)

传输层—数据段(segment)

网络层—分组、数据包(packet)

链路层—帧 (frame)

物理层—比特流

【32】D

【解析】

在 UNIX 系统中，telnet 服务的默认端口是 23，ftp 的端口号是 21 和 20。

【33】A

【解析】

关系型数据库是表的集合，表是记录的集合。

【34】B

【解析】

同【33】

【35】 B

【解析】

关系数据库中的关系模型事实上可以看作是一个二维表,这个二维表中的列称为属性(或字段),行称为元组(或记录)。

【36】 C

【解析】

关系的基本运算有两类:一类是传统的集合运算(并、差、交、笛卡尔积),另一类是专门的关系运算(选择、投影、连接、除法、外连接等)。

传统的集合运算:

(1) 并(UNION) 设有两个关系 R 和 S,它们具有相同的结构。R 和 S 的并是由属于 R 或属于 S 的元组组成的集合,运算符为 \cup 。记为 $T=R \cup S$ 。

(2) 差(DIFFERENCE) R 和 S 的差是由属于 R 但不属于 S 的元组组成的集合,运算符为 $-$ 。记为 $T=R - S$ 。

(3) 交(INTERSECTION) R 和 S 的交是由既属于 R 又属于 S 的元组组成的集合,运算符为 \cap 。记为 $T=R \cap S$ 。 $R \cap S = R - (R - S)$ 。

因此,对于两个关系 E 和 F, $E \cap F$ 的运算结果的任一元组,同时属于 E 和 F。 答案为 C。

【37】 C

【解析】

关系数据库中的关系模型事实上可以看作是一个二维表,这个二维表中的列称为属性(或字段),行称为元组(或记录)。

【38】 C

【解析】

完整性约束:防止的是对数据的意外破坏。

实体完整性:规定基本关系 R 的主属性 A 不能取空。

用户自定义完整性:就是针对某一具体关系数据库的约束条件,反映某一具体应用所涉及的数据必须满足的语义要求,由应用的环境决定。如:年龄必须为大于小于 150 的整数。

参照完整性引用完整性:规定,若 F 是基本关系 R 的外码,它与基本关系 S 的主码 K,相对应(基本关系 R 和 S 不一定是不同的关系),则 R 中每个元组在 F 上的值必须为:或者取空值;或者等于 S 中某个元组的主码值。

本题中属性完整性约束不属于关系的完整性约束。

【39】C

【解析】

参照完整性引用完整性:规定,若 F 是基本关系 R 的外码,它与基本关系 S 的主码 K,相对应(基本关系 R 和 S 不一定是不同的关系),则 R 中每个元组在 F 上的值必须为:或者取空值;或者等于 S 中某个元组的主码值。所以,在一个关系型数据库中,参照完整性约束可以通过外码来实现。

【40】C

【解析】

关系型数据库中常用的查询语言包括:域关系演算、元组关系演算、关系代数。

在计算机科学中,域关系演算(DRC)是 Michel Lacroix 和 Alain Pirotte 为关系数据模型发明的作为声明性数据库查询语言。

元组演算是埃德加科德导入的演算,是关系模型的一部分,发展目的是提供宣告式的数据库查询语言。数据库查询语言 QUEL 和后来的 SQL 中的一些灵感是由元组演算而来。SQL 和原来的关系模型和演算已有许多不同,后来成为实际上的数据库查询语言标准,几乎所有的关系数据库管理系统中都会用到 SQL 或是其变体。

域关系演算与元组关系演算最大的区别是域关系演算中的变量表示数据库的表属性,而元组关系演算的变量表示元组,即数据库的一行。

关系代数是一种抽象的查询语言,用对关系的运算来表达查询,作为研究关系数据语言的数学工具。关系代数的运算对象是关系,运算结果亦为关系。

【41】B

【解析】

一个取值域是原子的,是指该域的元素是不可分的单元。

【42】D

【解析】

考查关系数据库的基础知识。Delete 和收回权限的语法格式分别为：

(1) DELETE FROM 表名 WHERE 条件表达式

(2) 收回权限语句 REVOKE<权限>[, ... n] ON <对象类型> <对象名> FROM <用户>[, ... n]
[RESTRICT|CASCADE]

因此，此题的删除语句为 delete from... 收回权限用 REVOKE SELECT ON product FROM role A CASCADE。

【43】C

【解析】

同【42】

【44】B

【解析】

同【42】

【45】D

【解析】

同【42】

【46】B

【解析】

根据该关系模式的依赖关系，可以求得候选键为 A，D。根据函数依赖关系 $\{A \rightarrow B, A \rightarrow D, D \rightarrow A, B \rightarrow C\}$ ，可以得知并不存在部分依赖关系，由于 $A \rightarrow B, B \rightarrow C$ ，所以存在传递依赖的关系，所以，最高属于 2NF，原因是存在非主属性对码的传递函数依赖。

【47】D

【解析】

同【46】

【48】C

【解析】(答案有争议)

将 R 分解为两个关系模式 $R1\langle U1, D1\rangle$ 和 $R2\langle U2, D2\rangle$ 如果 $U1 \cap U2 \rightarrow U1 - U2$ 或者 $U1 \cap U2 \rightarrow U2 - U1$ 成立, 那么此分解具有无损连接性。所以, 如果 $U1 \cap U2 \rightarrow U1 - U2$ 不成立, 那么此分解不具有无损连接性, 这个说法是片面的, 是错误的。

【49】D

【解析】

考察 BCNF 的定义。满足 BCNF 的定义为: BCNF 中每个函数依赖左部都包含码。答案为 D。

【50】A

【解析】

多值函数依赖的定义为: 在 $R(U, F)$ 中, 其属性集为 U 。 X, Y, Z 是 U 的子集, 并且 $Z = U - X - Y$ 。当且仅当对 $R(U)$ 的任何一个关系 r , 给定一组属性 (x, z) 的值, 有一组 Y 值, 这组 Y 值仅仅决定于 x 值而与其他属性 $z(U - X - Y)$ 的值无关, 那么称 Y 多值依赖于 X 或 X 多值决定 Y , 记为: $X \twoheadrightarrow Y$

具有 6 种性质:

对称性: $X \twoheadrightarrow Y$, 则 $X \twoheadrightarrow Z$, 其中 $Z = U - X - Y$

传递性: $X \twoheadrightarrow Y, Y \twoheadrightarrow Z$, 则 $X \twoheadrightarrow Z - Y$

函数依赖可以看作多值依赖的特殊情况

若 $X \twoheadrightarrow Y, X \twoheadrightarrow Z$, 则 $X \twoheadrightarrow YZ$

若 $X \twoheadrightarrow Y, X \twoheadrightarrow Z$, 则 $X \twoheadrightarrow Y \cap Z$

若 $X \twoheadrightarrow Y, X \twoheadrightarrow Z$, 则 $X \twoheadrightarrow Z - Y$

平凡的多值依赖: 如果 Z 为空, 就是平凡的多值依赖; 如果 Z 不为空, 就是非平凡的多值依赖。

所以, 此题的正确答案为 A。

【51】C

【解析】

丢失修改: 是指两个事务 $T1$ 和 $T2$ 读入同一数据并修改, $T2$ 提交的结果破坏了 $T1$ 提交的结

果，导致 T1 的修改被丢失。

脏读:指当一个事务正在访问数据，并且对数据进行了修改，而这种修改还没有提交到数据库中，这时，另外一个事务也访问这个数据，然后使用了这个数据。因为这个数据是还没有提交的数据，那么另外一个事务读到的这个数据是脏数据，依据脏数据所做的操作可能是不正确的。

不可重复读:指在一个事务内，多次读同一数据。在这个事务还没有结束时，另外一个事务也访问该同一数据。那么，在第一个事务中的两次读数据之间，由于第二个事务的修改，那么第一个事务两次读到的数据可能是不一样的。这样就发生了在一个事务内两次读到的数据是不一样的，因此称为是不可重复读。

幻影现象是指同一事务对数据对象的两次访问得到的数据记录数不同。原因在于两次访问期间满足条件的记录被别的事务修改了或因为别的事务对表进行了增删，造成两次访问满足条件的记录不同了。属于不可重复读类不一致性。

一级封锁协议解决了丢失修改，二级封锁协议解决了脏读，三级封锁协议解决了不可重复读。所以，正确答案分别为 CAA。

【52】A

【解析】

同【51】

【53】A

【解析】（答案有争议）

同【51】

【54】D

【解析】

A 原子性 (Atomicity)

就是把事物分割成像原子一样，表示我们事物需要细微的去控制。比如我给你转钱，里面有我扣钱，你到账。总不能我扣了钱，你没到账这种情况吧。所以就是指转账这个事物，里面的所有环节哪怕一个出错，都需要事物回滚，就是一切回到之前那样。

C 一致性 (Consistency)

一致性是指事务必须使数据库从一个一致性状态变换到另一个一致性状态，也就是说一个事务执行之前和执行之后都必须处于一致性状态。还是转账来说，假设用户 A 和用户 B 两者的钱加起来一共是 1000, 那么不管 A 和 B 之间如何转账，转几次账，事务结束后两个用户的钱相加起来应该还得是 1000，这就是事务的一致性。

I 隔离性(Isolation)

隔离性是当多个用户并发访问数据库时，比如操作同一张表时，数据库为每一个用户开启的事务，不能被其他事务的操作所干扰，多个并发事务之间要相互隔离，即要达到这么一种效果：对于任意两个并发的事务 T1 和 T2，在事务 T1 看来，T2 要么在 T1 开始之前就已经结束，要么在 T1 结束之后才开始，这样每个事务都感觉不到有其他事务在并发地执行。

D 持久性(Durability)

持久性是指一个事务一旦被提交了，那么对数据库中的数据的改变就是永久性的，即便是在数据库系统遇到故障的情况下也不会丢失提交事务的操作。

【55】D

【解析】

数据库恢复操作的基本原理是冗余机制。

【56】B

【解析】

数据库系统在运行过程中可能会发生 CPU 故障，这属于系统故障。在此类故障的恢复过程中，需要根据日志进行的操作为 redo+undo。

【57】C

【解析】(答案有争议)

同【56】

【58】A

【解析】

触发器是一种特殊类型的存储过程，不由用户直接调用。创建触发器时会对其进行定义，以

便在对特定表或列作特定类型的数据修改时执行。触发器可以查询其他表，而且可以包含复杂的 SQL 语句。它们主要用于强制服从复杂的业务规则或要求。例如，您可以根据客户当前的帐户状态，控制是否允许插入新订单。触发器也可用于强制引用完整性，以便在多个表中添加、更新或删除行时，保留在这些表之间所定义的关系。

用户执行 SELECT 语句时是不能激活触发器的，只能通过 update, insert, delete 语句时才能激活触发器。所以答案 C 不正确。

【59】D

【解析】

存储过程(Stored Procedure)是在大型数据库系统中，一组为了完成特定功能的 SQL 语句集，它存储在数据库中，一次编译后永久有效，用户通过指定存储过程的名字并给出参数(如果该存储过程带有参数)来执行它。存储过程是数据库中的一个重要对象。主要作用包括了：提供了在服务器端快速执行 SQL 语句的有效途径存储过程降低了客户端和服务端之间的通信量；方便实施企业规则，当企业规则发生变化时只要修改存储过程，而无需修改其他应用程序。所以，答案为 D。

【60】B

【解析】

- (1) 排它锁(简记 X 锁)，又称写锁。若事物 T 对数据对象 A 加上 X 锁，则只允许 T 读取和修改 A，其他任何事物再不能对 A 加任何类型的锁，直到 T 释放 A 上的锁。
- (2) 共享锁(简记 S 锁)，又称读锁。若事物 T 对数据对象 A 加上 S 锁。则其他事物只能对 A 加 S 锁，不能再加 X 锁，直到 T 释放 A 上的 S 锁。所以，此题答案为 B。

【61】B

【解析】

SELECT 基本结构为：

SELECT [ALL | DISTINCT] <列名>[...n]

FROM <表名|视图名>[...n]

[WHERE <条件表达式>]

[GROUP BY <列名> [HAVING <条件表达式>]]

[ORDER BY <列名>[ASCDESC][...n]]

【62】A

【解析】

E-R 图向关系模式转换时，实体标识符转换为关系的码。

【63】A

【解析】

OLTP 指的是联机事务处理，OLAP 指的是联机分析处理。

【64】B

【解析】

同【63】

【65】C

【解析】

NULL 值代表空值。

【66】A

【解析】

在数据库系统中，使数据恢复到故障发生前的一致状态的机制称为恢复机制。

【67】A

【解析】

通过将一个关系拆分成两个更小的关系来使其满足范式时，必须用相同的属性使两个子关系互相关联来保持数据的完整性约束。

【68】A

【解析】

分布式数据库的透明性包括了分片透明、分配透明(复制透明、位置透明)、映像透明(模型透明)。而代码透明的说法是错误的。

【69】C

【解析】

分布式数据库的 CAP 理论指的是:对于一个分布式数据库系统,一致性、可用性和分区容错性这三个特点,最多只能满足 2 个。

一致性(Consistency)-系统在执行过某项操作后仍然处于一致的状态。在分布式系统中,更新操作执行成功后所有的用户都应该读取到最新的值,这样的系统被认为具有强一致性。也就是说所有节点在同一时间具有相同的数据;

可用性(Availability)-每一个操作总是能够在一定的时间内返回结果,这里需要注意的是“一定时间内”和“返回结果”。也就是说保证每个请求不管成功或者失败都有响应;

分区容忍(Partition tolerance)-分区容忍性可以理解为系统在存在网络分区的情况下仍然可以接受请求(满足一致性和可用性)。这里网络分区是指由于某种原因网络被分成若干个孤立的区域,而区域之间互不相通。还有一些人将分区容忍性理解为系统对节点动态加入和离开的处理能力,因为节点的加入和离开可以认为是集群内部的网络分区。也就是说系统中任意信息的丢失或失败不会影响系统的继续运作。

【70】C

【解析】

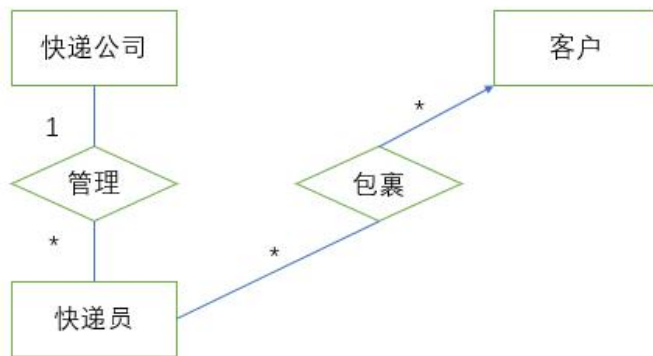
NOSQL 数据库的四大分类是指键值存储数据库,列存储数据库,文档型数据库,图数据库。

【71-75】A D B A B

数据库系统是相互关联的数据和一组允许用户访问和修改这些数据的程序的集合。数据库系统的主要目的是向用户提供数据的抽象视图。也就是说,系统隐藏了数据存储和维护的某些细节。为了使系统可用,必须高效地检索数据。对效率的需求导致设计人员使用复杂的数据结构来表示数据库中的数据。由于许多数据库系统用户不是经过计算机训练的,因此开发人员通过几个抽象层次向用户隐藏复杂性,以简化用户与系统的交互。物理层是描述数据实际存储方式的最低抽象层。逻辑层是描述数据库中存储的数据以及这些数据之间存在什么关系的下一个更高抽象级别,视图级别是只描述整个数据库的一部分的最高抽象级别。

试题一

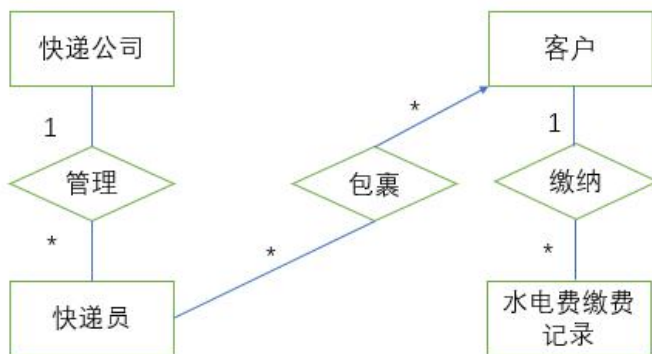
【问题 1】



【问题 2】

- (a) 快递公司名称
- (b) 客户手机号码

【问题 3】



水电费缴费记录（编号，客户手机号码，缴费类型，金额，时间）

试题二

【问题 1】

人员编号→人员姓名小区编号→物业经理姓名

该关系模式的主键为(人员编号, 小区编号), 非主属性为: 人员姓名、物业经理姓名、职责

该关系模式没有达到 2NF, 存在如上所示的两个非主属性对码的部分函数依赖。

【问题 2】

对人员分配表进行模式分解，消除存在的非主属性对码的部分函数依赖。

人员(人员编号, 人员姓名)

小区(小区编号, 物业经理姓名)

人员分配(人员编号, 小区编号, 人员职责)

【问题 3】

如上

其中，人员分配关系中的“人员编号”，“小区编号”同时又是外键

试题三

【问题 1】

- (a) foreign key references customers(CNO)或者 references customers(CNO)
- (b) foreign key references goods(GNO)或者 references goods(GNO)
- (c) check(0number between 1 and 99)

【问题 2】

- (d) 0price*0number
- (e) Orders.Cno= Customers.Cno
- (f) Orders.Gno=Goods.Gno
- (g) Order
- (h) 0amount Desc 或者 0price*0number Desc

售后微信:ruankaopass
淘宝店: 软考真题教育

【问题 3】

- (i) View
- (j) Count(0no)
- (k) Avg(0number)
- (l) Group
- (m) Orders.Gno

【问题 4】

(n) goods

(0) Except (差运算)

试题四

【问题 1】

(a) cursor

(b) open

(c) Pno, Pprice, Mno

(d) commit; return 0;

【问题 2】

T4: P_name=100; T7: P_name= 99

T8:不正确, 该调度存在丢失修改的不一致, 用户 1 的修改被用户 2 的修改覆盖了。

【问题 3】

(1) 两段封锁协议(2PL)理论上证明使用两段封锁协议产生的是可串行化调度, 事务必须分两个阶段:数据加锁(扩展阶段)和数据解锁(收缩阶段)。

(2) 两段封锁协议可以避免并发带来的不一致性, 但不能避免死锁;避免死锁可以采用:死锁检测(等待图是否有回路)及死锁解除机制, 或者采用一次性封锁方法。

试题五

【问题 1】

X=1 (T1 提交了, 执行 redo 操作); Y=50、Z=10(T2、 T3 没有提交, 执行 undo 回滚操作)

【问题 2】

redo:T1;

undo:T2、 T3

【问题 3】

于检查点的恢复过程如下:

(1) 从日志文件末尾向前扫描到最近的检查点记录 LSN9。

扫一扫, 叫我微信号:ruankaopass



提供软考历年真题, 视频

- (2) 读取检查点记录中的活动事务列表，这里是 T1、T2、T3。
- (3) 根据 T1、T2、T3 在出错前是否提交, 进行相应操作:提交的事务 redo 重做，未提交的事务 undo 回滚，如上问。
- (4) 完成恢复, 不必在向前扫描日志文件。