**测试开发工程师笔试题**

（总分100分，时间1周）

**姓名： 性别： 联系电话：**

1. 某学校为管理岗位业务培训信息，建立3个表:

学员信息表S

|  |  |  |
| --- | --- | --- |
| 序号 | 字段 | 字段描述 |
| 1 | SID | 学号 |
| 2 | SN | 学员名称 |
| 3 | SDATE | 学员生日 |
| 4 | SD | 所属院系 |

课程信息表C

|  |  |  |
| --- | --- | --- |
| 序号 | 字段 | 字段描述 |
| 1 | CID | 课程编号 |
| 2 | CN | 课程名称 |
| 3 | CT | 任课教师 |
| 4 | CD | 开课院系 |

学员选课表SC

|  |  |  |
| --- | --- | --- |
| 序号 | 字段 | 字段描述 |
| 1 | CID | 课程编号 |
| 2 | SID | 学员编号 |
| 3 | G | 成绩 |

要求使用标准SQL语句查询，实现如下4个处理：

1.查询选修课程编号为’C1012’的学员姓名和所属院系（4分）

select sn,sd from s,sc

where s.sid=sc.sid and sc.cid='c1012'

2.查询选修课程名称包含’C语言’字样的学员学号和姓名（4分）

select sc.sn,sc.sid from s,sc,c

where s.sid=sc.sid and c.cid=sc.cid and cn='c语言'

3.查询选修课程超过5门的学员学号和所属院系（6分）

select sn,sc.sid from s,sc

where s.sid=sc.sid

GROUP BY sid

having(COUNT(cid)>5)

4.查询所有需要补考的学员姓名和所属院系（考试成绩低于60或者无成绩的学员）（6分）

select sn,sd from s,sc

where s.sid=sc.sid and g<60 or g=NULL

二、写出一条Sql语句： 取出表A中第31到第40记录（SQLServer, 以自动增长的ID作为主键, 注意：ID可能不是连续的。）（8分）

select top 10 \* from A

where ID not in (select top 30 ID from A)

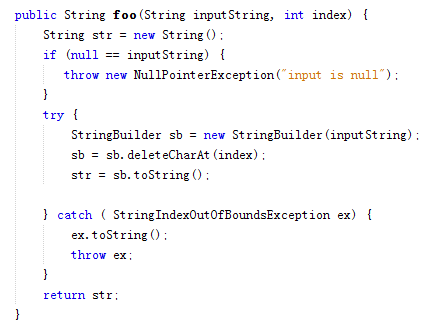
三、以VSS节点 192.168.29.146：8877 $/A自动化/cruiser/脚本/ＢＶＴ脚本，设计程序实现统计汇总，编程语言不限：（32分）

　　1.按产品模块分类统计ｘｍｌ文件中的ｔｅｓｔｃａｓｅ数与代码行数（去白行）;

　　2.按产品模块分类统计ｘｌｓ\ｘｌｓｘ文件中页签数与数据行数。

四、设计开发一款小工具，用于模拟指定IP机器的CPU及内存占用。（20分）

五、有一测试方法如下：



请对该方法进行单元测试，写明测试要点。（10分）

六、星际争霸是一款著名即时战略游戏，其中分为人族(T)、虫族(Z)和神族(P),每个种族的初级兵种如下表所述：（10分）

|  |  |  |
| --- | --- | --- |
| 种族 | 兵种 | 属性 |
| 人族(T) | 机枪兵 | 1. 远战，攻击距离10 2. 生命值：40、防卫力：0、攻击力：6 |
| 虫族(Z) | 狗仔 | 1. 近战 2. 生命值35、防卫力0、攻击力5 |
| 神族(P) | 狂战士 | 1. 近战 2. 生命值100、防卫力1、攻击力16、护盾60 |

请根据上述描述进行建模。

**public** **class** test3 {

**public static void** main**(**String**[]** args**)**{  
 GunSolider **gunSolider** = **new** GunSolider**()**;  
 **gunSolider**.setAttack**("6")**;  
 **gunSolider**.setDefence**("0")**;  
 **gunSolider**.setFarAttack**("10")**;  
 **gunSolider**.setLife**("40")**;  
 DogSolider **dogSolider** = **new** DogSolider**()**;  
 **dogSolider**.setDefence**("0")**;  
 **dogSolider**.setLife**("35")**;  
 **dogSolider**.setAttack**("5")**;  
 WildSolider **wildSolider** = **new** WildSolider**()**;  
 **wildSolider**.setDefence**("1")**;  
 **wildSolider**.setLife**("100")**;  
 **wildSolider**.setAttack**("16")**;  
 **wildSolider**.setShield**("60")**;  
}

}

**class** T

{

String farAttack;

String life;

String defence;

String attack;

**public** String getFarAttack() {

**return** farAttack;

}

**public** **void** setFarAttack(String farAttack) {

**this**.farAttack = farAttack;

}

**public** String getLife() {

**return** life;

}

**public** **void** setLife(String life) {

**this**.life = life;

}

**public** String getDefence() {

**return** defence;

}

**public** **void** setDefence(String defence) {

**this**.defence = defence;

}

**public** String getAttack() {

**return** attack;

}

**public** **void** setAttack(String attack) {

**this**.attack = attack;

}

}

**class** Z{

String nearAttack;

String life;

String defence;

String attack;

**public** String getNearAttack() {

**return** nearAttack;

}

**public** **void** setNearAttack(String nearAttack) {

**this**.nearAttack = nearAttack;

}

**public** String getLife() {

**return** life;

}

**public** **void** setLife(String life) {

**this**.life = life;

}

**public** String getDefence() {

**return** defence;

}

**public** **void** setDefence(String defence) {

**this**.defence = defence;

}

**public** String getAttack() {

**return** attack;

}

**public** **void** setAttack(String attack) {

**this**.attack = attack;

}

}

**class** P{

String nearAttack;String life;

String defence;String attack;

String shield;

**public** String getNearAttack() {

**return** nearAttack;

}

**public** **void** setNearAttack(String nearAttack) {

**this**.nearAttack = nearAttack;

}

**public** String getLife() {

**return** life;

}

**public** **void** setLife(String life) {

**this**.life = life;

}

**public** String getDefence() {

**return** defence;

}

**public** **void** setDefence(String defence) {

**this**.defence = defence;

}

**public** String getAttack() {

**return** attack;

}

**public** **void** setAttack(String attack) {

**this**.attack = attack;

}

**public** String getShield() {

**return** shield;

}

**public** **void** setShield(String shield) {

**this**.shield = shield;

}

}

**class** GunSolider **extends** T{

}

**class** DogSolider **extends** Z{

}

**class** WildSolider **extends** P{

}