# 一、代码阅读题

阅读下面的代码，系统打印的结果是什么？

**package** test;

**public** **class** Test {

**public** **static** **void** main(String[] args) {

SubsubClass ssc = **new** SubsubClass(3, 6, 9);

ssc.show();

}

}

**class** SuperClass {

**int** a;

**int** b;

**public** SuperClass(**int** x, **int** y) {

**this**.a = x;

**this**.b = y;

}

}

**class** SubClass **extends** SuperClass {

**int** b;

**int** c;

**public** SubClass(**int** xx, **int** yy, **int** zz) {

**super**(xx, yy);

**this**.b = b + zz;

**this**.c = a + b;

}

}

**class** SubsubClass **extends** SubClass {

**int** a;

**public** SubsubClass(**int** xx, **int** yy, **int** zz) {

**super**(xx, yy, zz);

**this**.a = a + b + c;

}

**public** **void** show() {

System.***out***.println("a=" + a + ";b=" + b + ";c=" + c);

}

}

打印：a=21;b=9;c=12

# 二、 算法实现

**Q2,算法题，请给出下面二维数组的排序算法**

int[][] Sort(int[][] input)

测试用例：

输入：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| [1] | [3] | [2] | [4] | [5] |
| [6] | [7] | [9] | [8] | [10] |
| [11] | [13] | [12] | [14] | [15] |
| [17] | [16] | [18] | [19] | [20] |
| [22] | [24] | [23] | [21] | [25] |

输入乱序

输出：

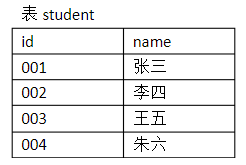
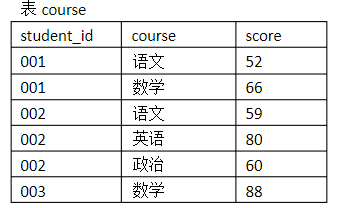
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| [11] | [7] | [4] | [2] | [1] |
| [16] | [12] | [8] | [5] | [3] |
| [20] | [17] | [13] | [9] | [6] |
| [23] | [21] | [18] | [14] | [10] |
| [25] | [24] | [22] | [19] | [15] |

注意输出的数字是沿对角排序, 注意处理2维数组的下标.

输入可以是hard code的常量,如果是这样,请将输入也打印到控制台.

答题请提供源码和可执行的class/jar文件

# 三 、数据库sql编写

以上两张表满足外键约束： course.student\_id = student.id

1.用一条SQL语句查出分数在60以下（不及格）的学生姓名及课程。（5分）

2.用一条SQL语句查询考试人数排名前三的课程名称和考试人数（5分）

3.用一条SQL语句查出各门课程的及格率（及格率：该门课程成绩60分及以上的学生数/考试的总学生数）（10分）

4.用一条SQL语句查出选修了4门课程以上且所有课程都及格（成绩60分及以上）的学生姓名。 （10分）

# 四、技术方案设计, 相关工作经验5年以下选做

库存子系统的Mysql（或Oracle）数据库里有张库存表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | 商品ID | 仓库ID | 仓库库存量 | 累计卖出量 |
| 1 | Sku001 | Store001 | 10000 | 129 |
| 2 | Sku002 | Store001 | 13000 | 750 |

功能性需求：当前store001的多个的商品在做抢购业务，需要您在库存子系统里设计一个库存扣减的服务API给订单子系统使用，当订单创建时，实时调用库存扣减的API，更新库存表的累计卖出量，如果累计卖出量超出仓库库存量，则库存扣减失败。

非功能性要求：

1）库存子系统采用10个节点的集群方式部署，有多个Java进程对外提供库存扣减API，需要解决并发修改数据的脏读和脏写的问题

2）库存扣减API需要实现幂等性，当订单模块多次相同的报文请求库存扣减时，只扣减一次，重复扣减的报文请求返回扣减成功。

3）库存子系统的每1个APP部署节点，需要达到1000TPS的库存扣减的性能要求（1秒内完成1000笔库存扣减）

部署硬件清单：

1. Mysql部署的虚拟机 8C 64G 1台
2. java应用部署的虚拟机2C 4G 10台
3. eureka服务发现服务器共用，订单子系统从eureka拿到库存子系统的10个节点的IP地址清单后，采用轮询IP的方式调用库存子系统的API
4. 不能再使用memcached或redis等其他第三方软件

针对以上的要求，请给出库存扣减的服务API的核心设计思想(10分)和API接口定义（10分）及代码实现概要