

# Solution

In the following, [n] means the line number n.

## 1. Java Night Countdown(30%)

1) javanight.util.Countdown

2) if(nMinute > 0)

```
{
```

```
    nMinute--;
```

```
    return true;
```

```
}else if(decrease_hour())
```

```
{
```

```
    nMinute = 59;
```

```
    return true;
```

```
}else
```

```
    return false;
```

3) return nMinute + nHour\*60 + nDay\*24\*60;

4) import javanight.util.\*; or import javanight.util.Countdown;

5) add public in front of the following method

```
Countdown(int year, int month, int day, int hour, int min)
```

```
countdown()
```

```
get_total_minute()
```

```
get_minute()
```

```
get_hours()
```

```
get_days()
```

```
in class Countdown.
```

6) In Countdown.java

```
[8], [18], [31], [34], [35], [36] - public
```

```
[6] including nDay, nHour, nMinute - private
```

```
[19], [23], [28] - private
```

Another solution of (5) (6):

(5) In Countdown.java

```
[2] changes into package javanight.publicity;
```

(6) In Countdown.java

```
[8], [18], [31], [34], [35], [36] - package default
```

```
[6] including nDay, nHour, nMinute - private
```

```
[19], [23], [28] - private
```

**Note:**

If you answer (5) with the second version, (6) must also be the second version.

## 2. Java Night Choir(20%)

(1)nStudent = IDs.length;

```
position = new [1][];
```

```
position[0] = new int[nStudent];
```

```
for(int i=0; i<IDs.length; i++)
```

```
IDs;
```

or

```
position[0] =
```

```
    position[0][i] = IDs[i];
```

```
reshape(nRow);
```

(2)this(IDs, 3)

(3)int[][] t;

```
int r, c;
```

```

t = new int[nRow][];
for(r=0; r<nStudent%nRow; r++)
    t[nRow-r-1] = new int[nStudent/nRow +1];
for(r= nStudent%nRow; r< nRow; r++)
    t[nRow-r-1] = new int[nStudent/nRow ];
int rr=0, cc=0;
for(r=0; r<position.length; r++)
    for(c=0; c<position[r].length; c++)
    {
        t[rr][cc] = position[r][c];
        cc++;
        if(cc == t[rr].length;)
        {
            rr++;
            cc = 0;
        }
    }
position = t;
(4) if( row >= 0 &&
    row < position.length &&
    num >= 0 &&
    num < position[row].length)
    return position[row][num];
else
    return -1;
(5) for(int r=0; r<position.length; r++)
    for(int c=0; c<position[r].length; c++)
        System.out.print(position[r][c]+" ");
    System.out.println();

```

### 3. Java Night PK(20%)

```

(1) private static void
(2) int tmp = a.ID;
    a.ID = b.ID;
    b.ID = tmp;
    tmp = a.Score;
    a.Score = b.Score;
    b.Score = tmp;
(3) 1: 2(90) 1(80)
(4) 1: 3(90) 5(80) 6(70) 1(95)
    2: 3(90) 5(80) 1(95) 6(70)
    3: 1(95) 5(80) 3(90) 6(70)
(5) 1: 3(90) 5(80) 1(85) 2(60) 4(70)
    2: 3(90) 5(80) 1(85) 4(70) 2(60)
    3: 3(90) 5(80) 1(85) 4(70) 2(60)
    4: 3(90) 5(80) 1(85) 4(70) 2(60)

```

### 4. Java Night Drinks(30%)

```

(1) 4
(2) private in [2] change into public, protected or
    just make it default by remove

```

(3) Cannot find empty constructor in Java.class

Sol: add to class BlendedJava  
public BlendedJava()  
{  
    super(0.0, 0.0);  
}

(4) super.drink(ratio\*sip);  
    milk.drink(sip-ratio\*sip);

(5) false

(6) hahaha (compile error)

(7) false

(8) ohohoh (run-time error)

(9) true

(10) true

(11) null

(12) hahaha (compile error)

(13) true

## Grading Criteria

In the following, "exact" means your answer must match the above exactly; otherwise, you get no point.

### 1. Java Night Countdown(30%)

(1) exact

(2) 少了 "else if" -2

    小錯誤 -1

(3) \* 寫成 x -1

(4) exact

(5) 寫錯一個 -1

    "多寫" 2 個以上 -1

(6) 同(5)

### 2. Java Night Choir(20%)

(1) 沒有 new 空間給任一變數 或 變數沒有宣告 -1

    一個邏輯上錯誤(影響程式正確性者) -1

(2) this(IDs, 3) -0

    Choir(IDs, 3) or this.Choir(IDs, 3) -2

    如果寫把第 1 小題 nRows 改成 3, 除非第 1 小題全對, 不然不給分

(3) 有正確先重新產生出答案中 IDs array 跟 答案後半部的 reshape code 是各一半的分數

    後者企圖呼叫 constructor 扣 2pts

    小錯誤太多 -2 (1~2 個不扣)

(4) 判斷式寫錯 -2

    只寫 return position -3

    小錯誤 -1

(5) minute syntax error -1

    wrong format -2

    run time error/compile error -4

### 3. Java Night PK(20%)

(1) exact

- (2) exact
- (3) exact
  - except wrong with newline("\n") -1
- (4) exact
  - except wrong with newline("\n") -1
- (5) exact
  - except wrong with newline("\n") -1
- 4. Java Night Drinks(30%)
  - (1) exact
  - (2) only "what is it" (explanation) -2
    - only "how to fix it" (solution) -1
  - (3) only "what is it" (explanation) -2
    - only "how to fix it" (solution) -1
    - Do not specifying where "Java()" nor "super(number, number)"
      - should be placed -2 (= no solution)
  - (4) fail the Java part -2
    - fail the milk part -2
    - Directly modified variable "super.amount" without changing
      - its permission from private to package or looser ( in 4. (2) )
      - 2 ( = fail the Java part)
  - (5)~(13) exact