

```
package com.principle.demeter.improve;

import java.util.ArrayList;
import java.util.List;

//客户端
public class Demeter1 {

    public static void main(String[] args) {
        System.out.println("~~~使用迪米特法则的改进~~~");
        //创建了一个 SchoolManager 对象
        SchoolManager schoolManager = new SchoolManager();
        //输出学院的员工id 和 学校总部的员工信息
        schoolManager.printAllEmployee(new CollegeManager());

    }

}
```

```
//学校总部员工类
class Employee {
    private String id;

    public void setId(String id) {
        this.id = id;
    }

    public String getId() {
        return id;
    }
}
```

```
//学院的员工类
class CollegeEmployee {
    private String id;

    public void setId(String id) {
        this.id = id;
    }

    public String getId() {
        return id;
    }
}
```

```

//管理学院员工的管理类
class CollegeManager {
    //返回学院的所有员工
    public List<CollegeEmployee> getAllEmployee() {
        List<CollegeEmployee> list = new ArrayList<CollegeEmployee>();
        for (int i = 0; i < 10; i++) { //这里我们增加了10个员工到 list
            CollegeEmployee emp = new CollegeEmployee();
            emp.setId("学院员工id= " + i);
            list.add(emp);
        }
        return list;
    }
}

```

```

//输出学院员工的信息
public void printEmployee() {
    //获取到学院员工
    List<CollegeEmployee> list1 = getAllEmployee();
    System.out.println("-----学院员工-----");
    for (CollegeEmployee e : list1) {
        System.out.println(e.getId());
    }
}
}

```

//学校管理类

//分析 SchoolManager 类的直接朋友类有哪些 Employee、CollegeManager
 //CollegeEmployee 不是 直接朋友 而是一个陌生类，这样违背了 迪米特法则

```

class SchoolManager {
    //返回学校总部的员工
    public List<Employee> getAllEmployee() {
        List<Employee> list = new ArrayList<Employee>();

        for (int i = 0; i < 5; i++) { //这里我们增加了5个员工到 list
            Employee emp = new Employee();
            emp.setId("学校总部员工id= " + i);
            list.add(emp);
        }
        return list;
    }
}

```

```

//该方法完成输出学校总部和学院员工信息(id)
void printAllEmployee(CollegeManager sub) {

```

//分析问题

```
//1. 将输出学院的员工方法，封装到CollegeManager  
sub.printEmployee();
```

```
//获取到学校总部员工
```

```
List<Employee> list2 = this.getAllEmployee();
```

```
System.out.println("-----学校总部员工-----");
```

```
for (Employee e : list2) {
```

```
    System.out.println(e.getId());
```

```
}
```

```
}
```

```
}
```