

定义

一个对象应该对其他对象保持最少的了解

问题由来

类与类之间的关系越密切，耦合度越大，当一个类发生改变时，对另一个类的影响也越大

解决方案

尽量降低类与类之间的耦合，让对象只与直接相关的进行耦合。

示例讲解

三个类 公司、部门、员工，要统计公司中员工数量

```
1 public class Employee {}
2
3 public class Dept {
4     public void employeeList(List<Employee> employeeList){
5         System.out.println("数量为: " + employeeList.size());
6     }
7 }
8
9 public class Company {
10     public void getEmployeeNumber(Dept dept){
11         List<Employee> list = new LinkedList<Employee>();
12         list.add(new Employee());
13         list.add(new Employee());
14         list.add(new Employee());
15         list.add(new Employee());
16         dept.employeeList(list);
17     }
18 }
```

如果按照上面的就是违反了 迪米特原则，公司与员工是并不直接关联的，应该通过中间角色 部门进行关联，正确的方式应该如下

```
1 public class Employee {}
2
3 public class Dept {
4     public void employeeList(){
5         List<Employee> list = new LinkedList<Employee>();
6         list.add(new Employee());
7         list.add(new Employee());
8         list.add(new Employee());
9         list.add(new Employee());
10        System.out.println("数量为: " + list.size());
11    }
12 }
13
14 public class Company {
15     public void getEmployeeNumber(Dept dept){
```

```
16     dept.employeeList();
17 }
18 }
```