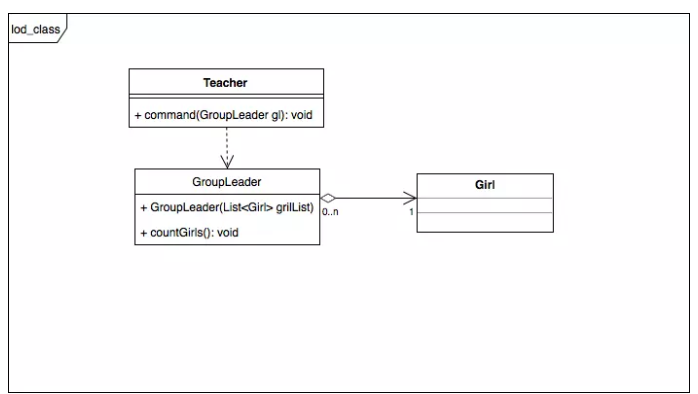
# 五、迪米特法则

2018.05.07 21:19:23字数 238阅读 136

### 1、定义

##### 迪米特法则(Law of Demeter, LoD)，也称为最少知识原则，一个对象应该对其他对象有最少的了解。



### 2、实现

### （1）创建女孩类

class Girl {

}

### （2）创建体育委员

class GroupLeader {

private List<Girl> listGirls;

public GroupLeader(List<Girl> listGirls) {

this.listGirls = listGirls;

}

public void countGirls() {

Log.d("JimmyZhang", "女生的数量:" + listGirls.size());

}

}

### （3）创建教师类

class Teacher {

public void commond(GroupLeader groupLeader) {

groupLeader.countGirls();

}

}

### （4）测试实现

class Client {

public static void main(String args) throws IOException {

List<Girl> listGirls = new ArrayList<>();

for (int i = 0; i < 20; i++) {

listGirls.add(new Girl());

}

Teacher teacher = new Teacher();

GroupLeader groupLeader = new GroupLeader(listGirls);

teacher.commond(groupLeader);

}

}

### 3、实用总结

##### （1）在类的划分上，应当尽量创建松耦合的类，类之间的耦合度越低，就越有利于复用，一个处在松耦合中的类一旦被修改，不会对关联的类造成太大波及；

##### （2）在类的结构设计上，每一个类都应当尽量降低其成员变量和成员函数的访问权限；

##### （3）在类的设计上，只要有可能，一个类型应当设计成不变类；

##### （4）在对其他类的引用上，一个对象对其他对象的引用应当降到最低。