

原型模式

原型模式是一种创建型模式，提供了一种创建对象的机制。原型模式用在当对象创建代价较大，需要大量的时间和资源，且你已经存在了一个类似的对象。所以，这种模式提供了一种机制将原始对象拷贝到一个新的对象，然后根据我们的需要修改它。这个模式使用Java中的克隆来拷贝对象。

举一个例子，这个模式理解起来会很简单，假如，我们有一个对象从数据库中加载数据。现在，我们需要多次在我们的程序中修改这份数据，因此通过使用new创建一个新的对象并再次从数据库中加载所有的数据并不是一个好主意。因此，一个更好的方式就是克隆一个已经存在的对象到一个新的对象，然后再进行数据操作。

原型模式要求你拷贝的对象应该提供拷贝特性。不可以通过其他类来做这件事。然而对对象属性使用浅拷贝还是深拷贝取决于你的需求，这是一个设计决定。

实现

```
package com.journaldev.design.prototype;

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

public class Employees implements Cloneable{

    private List<String> empList;

    public Employees(){
        empList = new ArrayList<String>();
    }

    public Employees(List<String> list){
        this.empList=list;
    }

    public void loadData(){
        //read all employees from database and put into the list
        empList.add("Pankaj");
        empList.add("Raj");
        empList.add("David");
        empList.add("Lisa");
    }

    public List<String> getEmpList() {
        return empList;
    }

    @Override
    public Object clone() throws CloneNotSupportedException{
        List<String> temp = new ArrayList<String>();
    }
}
```

```
        for(String s : this.getEmpList()){
            temp.add(s);
        }
        return new Employees(temp);
    }

}

package com.journaldev.design.test;

import java.util.List;

import com.journaldev.design.prototype.Employees;

public class PrototypePatternTest {

    public static void main(String[] args) throws
CloneNotSupportedException {
        Employees emps = new Employees();
        emps.loadData();

        //Use the clone method to get the Employee object
        Employees empsNew = (Employees) emps.clone();
        Employees empsNew1 = (Employees) emps.clone();
        List<String> list = empsNew.getEmpList();
        list.add("John");
        List<String> list1 = empsNew1.getEmpList();
        list1.remove("Pankaj");

        System.out.println("emps List: "+emps.getEmpList());
        System.out.println("empsNew List: "+list);
        System.out.println("empsNew1 List: "+list1);
    }

}
```

上面程序的输出：

```
emps HashMap: [Pankaj, Raj, David, Lisa]
empsNew HashMap: [Pankaj, Raj, David, Lisa, John]
empsNew1 HashMap: [Raj, David, Lisa]
```

如果没有提供对象克隆，每次我们都需要进行数据库调用来获取雇员列表，然后再做相关的操纵，这样会操作成资源和时间的浪费。