public class EqualsDemo{  
 private String name;  
 private String password;  
 private int id;  
  
 @Override  
 public boolean equals(Object o) {  
 if (this == o) return true;  
 if (o == null || getClass() != o.getClass()) return false;  
  
 EqualsDemo that = (EqualsDemo) o;  
  
 if (id != that.id) return false;  
 if (name != null ? !name.equals(that.name) : that.name != null) return false;  
 return password != null ? password.equals(that.password) : that.password == null;  
 }  
  
 @Override  
 public int hashCode() {  
 int result = name != null ? name.hashCode() : 0;  
 result = 31 \* result + (password != null ? password.hashCode() : 0);  
 result = 31 \* result + id;  
 return result;  
 }  
  
}  
class EqualsChild extends EqualsDemo{  
  
 private String password;  
  
 @Override  
 public boolean equals(Object o) {  
 //判断引用是否相等  
 if (this == o) return true;  
  
 //所有子类都有不同  
 //判断类是否为null和类是否一样  
 if (o == null || getClass() != o.getClass()) return false;  
  
 //如果所有子类都有同一语义时使用  
 //if (!(o instanceof EqualsDemo)) return false;  
  
 //引用超类的equals方法  
 if (!super.equals(o)) return false;  
  
 //设置为本类对象  
 EqualsChild that = (EqualsChild) o;  
  
 //逐一校验此类每个参数  
 return password != null ? password.equals(that.password) : that.password == null;  
  
 }  
  
 @Override  
 public int hashCode() {  
 return password != null ? password.hashCode() : 0;  
 }  
  
}