# 1、Filed

## 1、取得class

### 1、Class.forName()

#### 举例

Class<?> demo = Class.forName("com.reflect.Person2");

### 2、Person.class

Field[]fields=Person2.class.getDeclaredFields();

### 3、new 对象出来.class

Person pserson=new Person ();

Field[] fields= pserson.getClass().getDeclaredFields();

## 2、取得反射的字段内容，注意私有和公有public

<http://www.cnblogs.com/manuosex/p/3852716.html>

### 1.测试实体类，y公有 x私有

|  |
| --- |
| public class Point {    private int x;  public int y;  public Point(int x, int y) {  super();  this.x = x;  this.y = y;  }    } |

### 2.1、使用getField 取得公有的y（不能取得私有的）

|  |
| --- |
| Point p = new Point(3,5);    //得到一个字段  Field fieldY = p.getClass().getField("y"); //y 是变量名  System.out.println(fieldY.get(pt1)); // 5 |

### 2.2 如果是私有的getDeclaredField ，setAccessible

#### 1、使用上面的getField会报错，提示找不到NoSuchFieldException

#### 2、getDeclaredField，提示有，但是取不出来java.lang.IllegalAccessException: //Class com.ncs.ReflectTest can not access a member of class com.ncs.Point with modifiers "private"

#### 3、暴力反射2+setAccessible

|  |
| --- |
| Field fieldX = pt1.getClass().getDeclaredField("x");  //这个管你公的私的，都拿来  fieldX.setAccessible(true);//上面的代码已经看见钱了，开始抢了  System.out.println(fieldX.get(pt1));   //out 3 OK!! |

## 3、getType（） 取得字段类型

Field fieldX = pt1.getClass().getDeclaredField("x");

field.getType()== String.class

## 4.、getName，则表示该字段名字（String 类型）

field.getName()

## 4、set把一个类里所有String类型的字段里的b的值 变 成a的值

|  |
| --- |
| Field[] fields = pt1.getClass().getFields();       for(Field field : fields ) {       //我说的是String 类型的字段，当然要选择一下       //if(field.getType().equals(String.class)) {       //字节码都是一份的，干什么用equals啊       if(field.getType()== String.class) {          String oldValue = (String)field.get(pt1);          String newValue = oldValue.replace('b', 'a');           //改 对象的字段            field.set(pt1, newValue);                }          } |

## 5、从xml文件中读出内容利用field反射，放进实体对象中，实战，其中有省略

|  |
| --- |
| Field[] fields=eaheadvo.getClass().getDeclaredFields();  OAXMLHeadVO eaheadvo=new OAXMLHeadVO();  **while**(iteEle.hasNext()){  **for**(**int** i=0;i<fields.length;i++){  Field field=fields[i];  setFieldValue(field, eaheadvo, valueTxt);  } |

### 5.2实现方法如下，这样就会有实体对象出现，其内部也有相应的值

|  |
| --- |
| /\*\*  \* 为属性赋值  \* **@param** field 要赋值的属性  \* **@param** ins 属性所属对象  \* **@param** value 值  \*/  **private** **void** setFieldValue(Field field,Object ins,String value){  **try**{  **if**(field==**null** || ins==**null** || value==**null** || "".equals(value)){  **return**;  }  field.setAccessible(**true**);    String fieldType=field.getGenericType().toString();  **if**("class java.lang.String".equals(fieldType)){  field.set(ins, value);  }  **if**("class java.lang.Integer".equals(fieldType)){  Integer val=Integer.*valueOf*(value);  field.set(ins, val);  }  **if**("class java.math.BigDecimal".equals(fieldType)){  BigDecimal bde=**new** BigDecimal(value);  field.set(ins, bde);  }  **if**("class java.util.Date".equals(fieldType)){  SimpleDateFormat df=**new** SimpleDateFormat("yyyyMMdd");  Date date=df.parse(value);  field.set(ins, date);  }  }**catch**(Exception ex){  logger.error(ex);  **throw** **new** BusinessException("处理失败");  }    } |

## 6、[getDeclaredFields() 与getFields()的区别](http://blog.csdn.net/zhaofeiyue1234/article/details/50781005)

<http://blog.csdn.net/zhaofeiyue1234/article/details/50781005>

<https://yq.aliyun.com/articles/41923>

## （1）getDeclaredFields()返回Class中所有的字段，包括私有字段；能访问从其它类继承来的公有方法

## （2）getFields  只返回公共字段，即有public修饰的字段 不能访问从其它类继承来的方法

# 2、Method 用到再说吧

## getMethods()与getDeclaredMethods()区别:getMethods()只能访问类中声明为公有的方法,私有的方法它无法访问,能访问从其它类继承来的公有方法. getDeclaredMethods ()能访问类中所有的字段,与public,private,protect无关,不能访问从其它类继承来的方法

Method[] methods = this.getClass().getMethods();

31 for (Method m : methods) {

32 System.out.println(m.getName());

33 }

# 3、getConstructors

getConstructors()与getDeclaredConstructors()区别:getConstructors()只能访问类中声明为public的构造函数.getDeclaredConstructors()能访问类中所有的构造函数,与public,private,protect无关