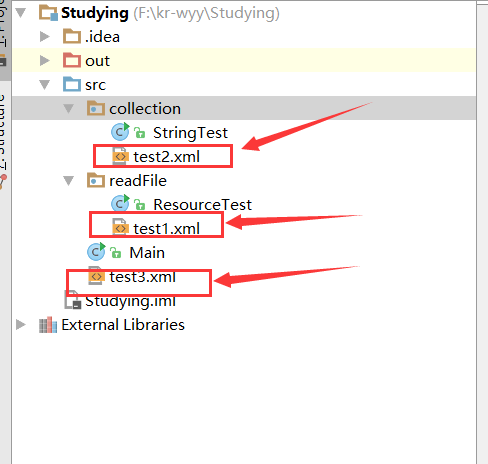
[**java基础知识3--如何获取资源文件（Java中获取资源文件的url）**](http://www.cnblogs.com/alwayswyy/p/6421267.html)

java开发中，常见的resource文件有：.xml,.properties,.txt文件等，后台开发中经常用到读取资源文件，处理业务逻辑，然后返回结果。

获取资源文件的方法说明  
getResource()返回:URL   
getResourceAsStream () 返回的是inputstream，需要定义一个InputStream接收   
//Class.getResource和Class.getResourceAsStream在使用时，路径选择上是一样的。  
相当于你用getResource()取得File文件后，再new InputStream(file)一样的结果.

今天先了解一下如何读取资源文件的URL。  
以下图的逻辑结构为例：  
  
有4种读取文件URL的方法：

**1.通过本类的class类的getResource方法**  
path不以’/'开头时，默认是从此类所在的包下取资源；  
path 以’/'开头时，则是从ClassPath根下获取；

[复制代码](javascript:void(0);)

// 同一个包下

URL url1 = ResourceTest.class.getResource("test1.xml");

System.out.println(url1.toString());

//输出 file:/F:/kr-wyy/Studying/out/production/Studying/readFile/test1.xml

//不同包下,用相对路径

URL url2 = ResourceTest.class.getResource("/collection/test2.xml");

System.out.println(url2.toString());

//输出：file:/F:/kr-wyy/Studying/out/production/Studying/collection/test2.xml

//根目录下

URL url3 = ResourceTest.class.getResource("/test3.xml");

System.out.println(url3.toString());

//输出：file:/F:/kr-wyy/Studying/out/production/Studying/test3.xml

/\* 即： file:/F:/kr-wyy/Studying/out/production/Studying/readFile/test1.xml

file:/F:/kr-wyy/Studying/out/production/Studying/collection/test2.xml

file:/F:/kr-wyy/Studying/out/production/Studying/test3.xml\*/

[复制代码](javascript:void(0);)

**2、通过本类的ClassLoader的getResource方法**  
 path不能以’/'开头，path是从ClassPath根下获取；  
 所以可以认为：类名.class.getResource("/") == 类名.class.getClassLoader().getResource("")

[复制代码](javascript:void(0);)

URL u1 = ResourceTest.class.getClassLoader().getResource("readFile/test1.xml");//同一个包下,还是要包名/文件名

System.out.println(u1);

URL u2 = ResourceTest.class.getClassLoader().getResource("collection/test2.xml");//不同包下,要包名/文件名

System.out.println(u2);

URL u3 = ResourceTest.class.getClassLoader().getResource("test3.xml");//根目录下直接写文件名

System.out.println(u3);

[复制代码](javascript:void(0);)

**3、通过ClassLoader的getSystemResource(),路径和2一致**

[复制代码](javascript:void(0);)

URL r1 = ClassLoader.getSystemResource("readFile/test1.xml");//同一个包下,还是要包名/

System.out.println(r1);

URL r2 = ClassLoader.getSystemResource("collection/test2.xml");//不同包下,要包名/文件名

System.out.println(r2);

URL r3 = ClassLoader.getSystemResource("test3.xml");//根目录下直接写文件名

System.out.println(r3);

[复制代码](javascript:void(0);)

**4、通过Thread方式,路径和2一致(推荐此种)，Thread.currentThread().getContextClassLoader()**为什么推荐此类？  
Test.class.getClassLoader().getResource，因为你想要获取某个资源文件的时候，这个资源文件的位置是相对固定的。  
WEB程序，里面的jar、resources都是由Tomcat内部来加载的，所以你在代码中动态加载jar、资源文件的时候，首先应该是使用Thread.currentThread().getContextClassLoader()。  
如果你使用Test.class.getClassLoader()，可能会导致和当前线程所运行的类加载器不一致（因为Java天生的多线程）

[复制代码](javascript:void(0);)

URL l1=Thread.currentThread().getContextClassLoader().getResource("readFile/test1.xml");

System.out.println(l1);

URL l2=Thread.currentThread().getContextClassLoader().getResource("collection/test2.xml");

System.out.println(l2);

URL l3=Thread.currentThread().getContextClassLoader().getResource("test3.xml");

String l4=l3.getPath();//加上getPath()则去掉前面的file:

System.out.println(l3); //file:/F:/kr-wyy/Studying/out/production/Studying/test3.xml

System.out.println(l4);///F:/kr-wyy/Studying/out/production/Studying/test3.xml

[复制代码](javascript:void(0);)

另外，其他获取资源文件路径的有下面几个：

[复制代码](javascript:void(0);)

// 获取项目路径

System.out.println(ResourceTest.class.getResource("").getPath());

// /F:/kr-wyy/Studying/out/production/Studying/readFile/

// 获取根路径

System.out.println(ResourceTest.class.getResource("/").getPath());

// /F:/kr-wyy/Studying/out/production/Studying/

//获取当前工程路径

System.out.println(System.getProperty("user.dir"));

// F:\kr-wyy\Studying

// 获取所有的类路径 包括jar包的路径

System.out.println(System.getProperty("java.class.path"));

[复制代码](javascript:void(0);)

上面的可以通过创建File文件，通过路径来读取文件内容。

但是对于打包到jar文件中的资源文件，则不能通过上面的getResource方法获取了，但可以通过**Class类的getResourceAsStream()方法来获取**

**例如：**resource打包成jar包，我们读取里面的res.txt可如下，

[复制代码](javascript:void(0);)

/返回读取指定资源的输入流

InputStream is=this.getClass().getResourceAsStream("/resource/res.txt");

BufferedReader br=new BufferedReader(new InputStreamReader(is));

String s="";

while((s=br.readLine())!=null)

System.out.println(s);

}

[复制代码](javascript:void(0);)