

Resource和其实现类的关系图

两个例子:

Spring 的Resource接口及其实现类可以脱离Spring框架使用，比JDK的资源访问api更加好用，和强大。

几个例子：

FileSystemResource,以文件系统的绝对路径的方式进行访问。

ClassPathResource，以类路径的方式进行访问。

ServletContextResource，以相对于web应用根目录的方式进行访问。

资源地址表达式：

clsspath:com/controller/test.java 相对于类的根路径

file:com/controller/test.java 可以是相对路径，也可以是绝对路径

<http://www.baidu.com/resource/test.xml> 使用UrlResource从Web服务器上获取资源。

<ftp://www.baidu.com/resource/test.xml>使用UrlResource从FTP服务器上获取资源。

String path = "E:/java/abc.txt"; //绝对路径

Resource resource1 = **new** FileSystemResource(path);

System.*out*.println("resource1 : "+resource1.getFilename());

File f = **new** File("text.txt");

System.*out*.println(f.getCanonicalPath());//E:\workspace\dining1.0\backend\text.txt

System.*out*.println(f.getAbsolutePath());//E:\workspace\dining1.0\backend\text.txt

Resource resource2 = **new** FileSystemResource(f);

System.*out*.println("resource2 : "+resource2.getFilename());

Resource res=**new** ClassPathResource("com/dspsemi/backend/controller/account/file.txt");//相对路径

EncodedResource enc=**new** EncodedResource(res,"UTF-8");

String cont=FileCopyUtils.*copyToString*(enc.getReader());

System.*out*.println(cont);

ResourcePatternResolver resolver=**new** PathMatchingResourcePatternResolver();

Resource resources[]=resolver.getResources("classpath\*:/com/dspsemi/backend/controller/account/\*");

**for**(Resource resource:resources){

System.*out*.println(resource.getFilename());//注意这边的文件后缀.class文件

}

Console:

resource1 : abc.txt

E:\workspace\dining1.0\backend\text.txt

E:\workspace\dining1.0\backend\text.txt

resource2 : text.txt

this is content of file.txt

BillController.class

file.txt

ReconciliationController.class

SourceSystemTest.class

VoucherController.class