

# Java打开文件的所有方式收录

文章分类: [JavaSE](#); 标签: [JavaCodeSnippet](#); 作者: [Hackyle](#);

更新时间: Tue Aug 29 13:48:55 CST 2023

1. [user.dir](#)
2. [ClassLoader](#)
3. [IDEA](#)
4. [Tomcat](#)
5. [Spring加载文件](#)
6. [读取resources下的文件](#)

本文的主要内容是介绍几种常用的文件打开方式，可以解决实际项目开发中很多的文件打开场景

- 基于环境变量`user.dir`找到文件位置
- 基于`ClassLoader`找到文件的位置
- 在`IDEA`中打开文件
- 在`Tomcat`中打开文件
- 基于`Spring`找到文件的位置
- 基于`Maven`打开`resources`目录下的文件

## 打开文件的一般思路

1. 确定文件位置
2. 使用`java.io.File`打开文件，使用`InputStream`、`Reader`载入文件流，或者`OutputStream`、`Writer`写出文件流

## 内容导览

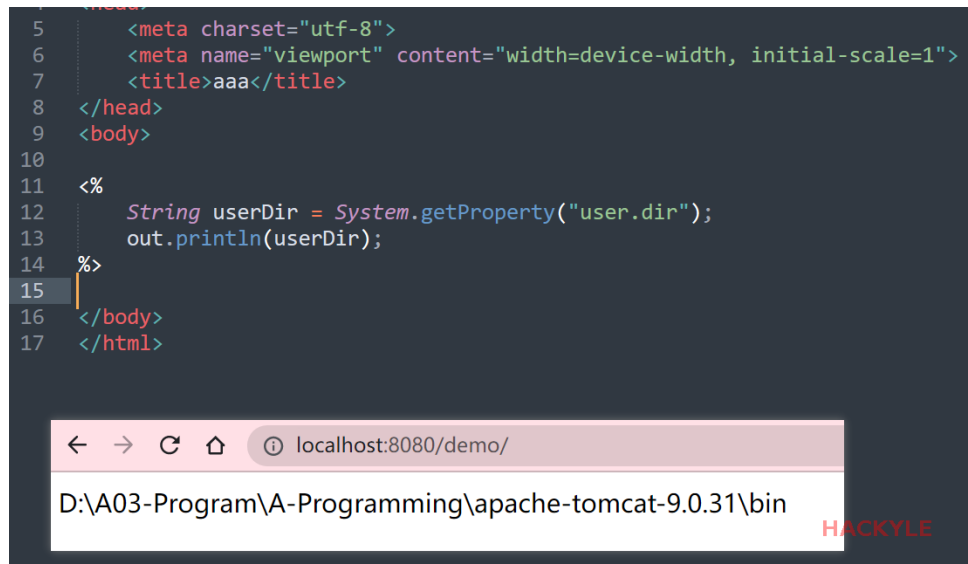
- [user.dir](#)
- [ClassLoader](#)
- [IDEA](#)
- [Tomcat](#)
- [Spring加载文件](#)
- [读取resources下的文件](#)

## user.dir

- `dir`系统属性是 JVM 启动的时候设置的，通常是 Java 虚拟机的调用目录，即执行 `java` 字节码所在的目录。
- 如果存在包，那么 `"/` 为包的上一级目录。如果是`jar`包，则 `"/` 为当前`jar`包所在目录。
- 对于`Tomcat/jboss` 容器，`dir` 是 `%web容器目录%/bin/` 目录，因为这个目录就是我们启动 `web` 容器的地方。
- 在`Eclipse`和`IDEA`，`getProperty("user.dir")`为工程的根目录

```
1 public class OpenFileDemo {
2     public static void main(String[] args) throws Exception {
3         String userDir = System.getProperty("user.dir");
4         System.out.println("当前程序(不含包名)运行所在目录: " + userDir); //D:\A03-P
5     }
6 }
```

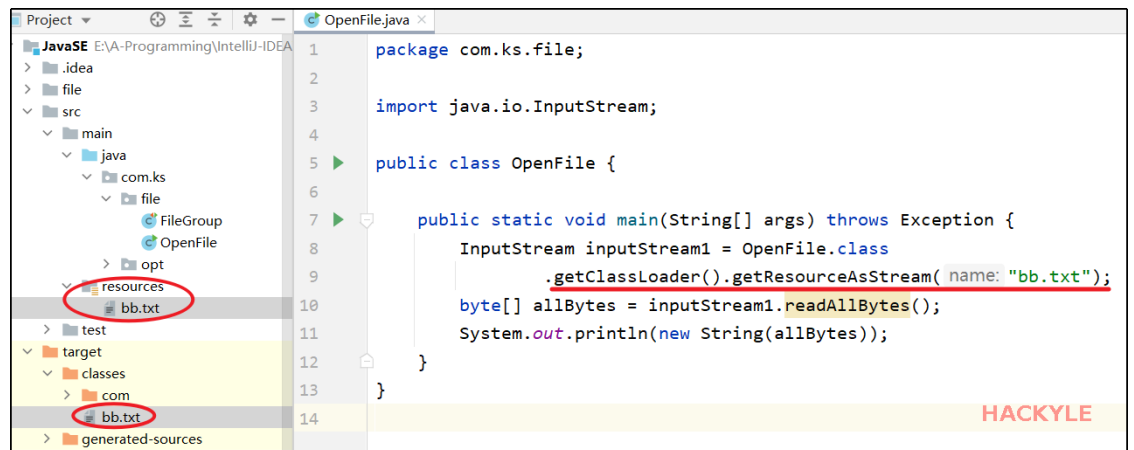
## 在Tomcat容器中的user.dir为bin目录



## ClassLoader

- 读取字节码路径下的文件
  1. 类名.getResource(“资源名称”), 返回URL
    1. 绝对路径以 / 开头, 从classpath 或 jar 包根目录下开始搜索;
    2. 相对路径是相对当前class 所在的目录, 允许使用 .. 或 . 来定位文件。
  2. 类名.getClassLoader().getResourceAsStream(), 返回InputStream。只能使用绝对路径, 而且不用以 / 开头。
- 这两种方式读取资源文件, 不会依赖于dir, 也不会依赖于具体部署的环境, 是推荐的做法
- ClassLoader 不能读取太大的文件, 它适合读取项目的配置文件

### ClassLoader加载字节码所在目录下的文件



获取当前字节码文件的绝对路径

```
import java.io.File;
import java.net.URL;
```

```
public class OpenFileDemo {
    public static void main(String[] args) throws Exception {
        //在静态方法中使用".class.getClassLoader()"获取类加载器
        String path = OpenFileDemo.class.getClassLoader().getResource("").getPath(
        System.out.println(path); //输出: /D:/A03-Program/A-Programming/IDEA/boot/
        //File file = new File(path + "aa.txt");

        new OpenFileDemo().test01();
    }

    public void test01() {
```

```

19         System.out.println(path);
20         String filePath = URLDecoder.decode(path, "UTF-8"); //如果路径中带有中文会被
21
22         String fileName = "aa.txt";
23         File file = new File(filePath + fileName);
24         System.out.println(file.exists()); //true
25     }
26 }

```

## IDEA

### IDEA默认从项目根目录查找文件

```

1 public static void readFile() throws Exception {
2     //IDEA中还是以工程的根目录为相对文件的根目录
3     //在项目的根目录下建立一个文件夹file，在其文件夹下建立文件hello.txt
4     File file = new File("file/hello.txt");
5
6     System.out.println("文件的绝对路径: " + file.getAbsolutePath());
7     BufferedInputStream in = new BufferedInputStream(new FileInputStream(file));
8     byte[] buffer = new byte[1024];
9     int read = in.read(buffer);
10    System.out.println("文件内容: " + new String(buffer, 0, read, StandardCharsets
11
12    //Output:
13    //文件的绝对路径: D:\E-DataProject\JSE\file\hello.txt
14    //文件内容: This is a test text.
15 }

```

## Tomcat

### 获取Tomcat的运行目录

- new File(".").getAbsolutePath() 当前文件所在目录
- getSession().getServletContext().getRealPath("") 项目所在目录
- getProperty("catalina.home") Tomcat根目录
- getProperty("user.dir") Tomcat的bin目录，因为Tomcat是从该目录启动的

```

<%@ page pageEncoding="UTF-8" contentType="text/html; charset=UTF-8" %>
<%@ page import="java.io.File" %>

<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <title>PN</title>
</head>
<body>
<%
    File file01 = new File(".");
    pageContext.setAttribute("file01", file01.getAbsolutePath());
    File file02 = new File("./");
    pageContext.setAttribute("file02", file02.getAbsolutePath());
    File file03 = new File("../");

```

```

20 %>
21 <!-- JSP表达式 -->
22 catalina.home: <%= System.getProperty("catalina.home") %> <br/>
23 user.dir: <%= System.getProperty("user.dir") %> <br/>
24 requestServletPath: ${requestServletPath} <br/>
25
26 file01: ${file01} <br/>
27 file02: ${file02} <br/>
28 file03: ${file03} <br/>
29
30 <!--
31     Output
32     catalina.home: F:\A04-Storage\PN-Tomcat\apache-tomcat-9.0.31-windows-x64
33     user.dir: F:\A04-Storage\PN-Tomcat\apache-tomcat-9.0.31-windows-x64\bin
34     requestServletPath: F:\A04-Storage\PN-Tomcat\apache-tomcat-9.0.31-windows-x64\
35
36     file01: F:\A04-Storage\PN-Tomcat\apache-tomcat-9.0.31-windows-x64\bin\.
37     file02: F:\A04-Storage\PN-Tomcat\apache-tomcat-9.0.31-windows-x64\bin\.
38     file03: F:\A04-Storage\PN-Tomcat\apache-tomcat-9.0.31-windows-x64\bin\.
39 -->
40 </body>
41 </html>

```

## Spring加载文件

```

1 import org.springframework.core.io.ClassPathResource;
2
3 //加载resources/static/template目录下的文件
4 ClassPathResource resource = new ClassPathResource("static/template/report_templat
5 File file = resource.getFile();
6
7
8 import org.springframework.util.ResourceUtils;
9 File rootPath = new File(ResourceUtils.getURL("classpath:").getPath()); //SpringBc
10 File templatePath = new File(rootPath.getAbsolutePath(), "/excel_template/userInfo.

```

```
import org.springframework.core.io.ClassPathResource;
```

```
import java.io.BufferedInputStream;
import java.io.File;
import java.io.InputStream;
import java.net.URL;
import java.net.URLDecoder;
import java.nio.charset.StandardCharsets;
```

```
public class OpenFileDemo {
    public static void main(String[] args) throws Exception {
        OpenFileDemo openFileDemo = new OpenFileDemo();
        openFileDemo.method01();
        openFileDemo.method02();
    }
}
```

```
        openFileDemo.method05();

        openFileDemo.method06();
        openFileDemo.method07();
    }

    /**
     * getPath(): 直接通过文件名来获取路径
     * 使用单元测试@Test, 获取的是Junit的jar包所在位置
     */
    public void method01() {
        URL resource = this.getClass().getClassLoader().getResource(""); //注意get
        String path = resource.getPath();
        String filePath = URLDecoder.decode(path, StandardCharsets.UTF_8); //如果路

        //打开resources/data/aa.txt下的文件
        File file = new File(filePath + "data/aa.txt");
        System.out.println(filePath + ": " +file.exists());
    }

    public void method02() {
        //打开resources/data/aa.txt下的文件
        URL resource = this.getClass().getClassLoader().getResource("data/aa.txt")
        String path = resource.getPath();
        String filePath = URLDecoder.decode(path, StandardCharsets.UTF_8); //如果路

        File file = new File(filePath);
        System.out.println(filePath + ": " +file.exists());
    }

    /**
     * getFile():等价于getPath
     * 如果是文件路径的话getFile和getPath效果是一样的, 如果是URL路径的话getPath是带有参
     */
    public void method03() {
        //打开resources/data/aa.txt下的文件
        URL resource = this.getClass().getClassLoader().getResource("data/aa.txt")
        String path = resource.getFile();
        String filePath = URLDecoder.decode(path, StandardCharsets.UTF_8); //如果路

        File file = new File(filePath);
        System.out.println(filePath + ": " +file.exists());
    }

    /**
     * getResourceAsStream: 直接获取到InputStream
     * 基于Spring
     */
    public void method04() throws Exception {
        //打开resources/data/aa.txt下的文件
        InputStream in = this.getClass().getClassLoader().getResourceAsStream("dat
        BufferedInputStream is = new BufferedInputStream(in);
        System.out.println(new String(is.readAllBytes(), StandardCharsets.UTF_8));
    }

    /**
     * 通过ClassPathResource获取文件流
```

```

79      .....
80      ClassPathResource classPathResource = new ClassPathResource("data/aa.txt")
81      InputStream in = classPathResource.getInputStream();
82      BufferedInputStream is = new BufferedInputStream(in);
83      System.out.println(new String(is.readAllBytes(), StandardCharsets.UTF_8));
84  }
85
86  /**
87   * 获取项目的绝对路径，不能用于服务器获取
88   * 不要使用!!!
89   */
90  public void method06() throws Exception {
91      String projectPath = System.getProperty("user.dir");
92      String filePath = projectPath + "/src/main/resources/data/aa.txt";
93      File file = new File(filePath);
94      System.out.println(filePath + ": " + file.exists());
95  }
96
97  /**
98   * 获取当前字节码文件的绝对路径
99   * 不能用于服务器获取
100  */
101  public void method07() throws Exception {
102      File dir = new File(".");
103      File canonicalFile = dir.getCanonicalFile();
104      String absolutePath = dir.getAbsolutePath();
105      System.out.println(absolutePath);
106  }
107  }
```

版权声明：非明确标注皆为原创文章，遵循CC 4.0 BY-SA版权协议，转载请附上本文链接及此声明。  
 原文链接：<https://blog.hackyle.com/article/java/open-file>

留下你的评论

Name:

Email:

Link:

File Edit View Format Tools Table Help

↶ ↷ **B** *I* U ~~S~~ A *ℱ* Ω ☺ ...

Input comment, please

© Copy Right: 2022 HACKYLE. All Rights Reserved

Designed and Created by HACKYLE SHAW

备案号: 浙ICP备20001706号-2