Java 的 SystemInfoUtil 类

作者：ykz200
原文链接：https://ld246.com/article/1498712210586
来源网站：链滴

本文由黑壳网原创

本文来源Java的SystemInfoUtil类 - 黑壳网

壳叔搞笑一刻

单位有个女同事，平时爱占小便宜。

今天一上班，辫子上绑着根捆蛋糕的绳子。

我们问她:今天怎么绑这么一根东西。

她回答到:今天我生日，给你们提个醒，别忘了给我发红包。

我去，早知道不问她了……

今天闲着没事，正好公司项目后台想显示一下服务器的各种信息，感觉不至于那么low，所以根据网的一些资料，整理出这个么一个工具类。

其实代码 so easy 重在理解。就像解决错误一样，你不能一味的去各种试一试改一改，你得去了解这错误，才能对症下药。

工具类简单说明

getSystemInfo() 获取系统环境等相关信息

其实获取服务器环境的信息代码简单，大多数都是从Java的System 这个大佬身上获取出来的。没什可说的只要根据需求取值就可以了。不一定所有的都会用上所以根据需要适当调整

getDatabaseMajorVersion() 获取数据库的主要版本等信息

通过jdbc 读取数据库数据信息~ 只不过这里的以下代码是从jdbc.properties 配置文件中获取的, 根自己这个地方自己进行适当的修改。

getProjectInfo() 获取项目版本信息

这个地方是直接读取配置文件,将相关的信息读取出来。

```
/**
 * 从配置文件读取 链接数据库信息
 * 然后建立连接
 * 用适当的驱动程序连接到DBMS，看下面的代码[自行修改您所连接的数据库相关信息]:
 */
props.load(SystemInfoUtil.class.getClassLoader().getResourceAsStream("jdbc.properties"));
String url = props.getProperty("jdbcUrl");
String user = props.getProperty("username");
String password = props.getProperty("password");
String driver = props.getProperty("driverClasss");
```

详细代码

SystemInfoUtil类

```
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;

import java.io.IOException;
import java.lang.reflect.Method;
```

```
import java.net.InetAddress;
import java.net.UnknownHostException;
import java.sql.Connection;
import java.sql.DatabaseMetaData;
import java.sql.DriverManager;
import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.HashMap;
import java.util.Locale;
import java.util.Map;
import java.util.Properties;
import java.util.logging.SimpleFormatter;

/**
 * 获取系统信息以及服务器信息
 * Created by kzyuan on 2017/6/22.
 */
public class SystemInfoUtil {

    private static Logger logger = LoggerFactory.getLogger(SystemInfoUtil.class);

    /**
     * 服务器环境相关信息
     *
     * @return
     */
    public Map getSystemInfo() {
        Map<String, String> map = new HashMap<String, String>();

        /**
         * java.version Java 运行时环境版本
         */
        map.put("java_version", System.getProperty("java.version"));
        /**
         * java.vendor Java 运行时环境供应商
         */
        map.put("java_vendor", System.getProperty("java.vendor"));
        /**
         * java.vendor.url Java 供应商的 URL
         */
        map.put("java_vendor_url", System.getProperty("java.vendor.url"));
        /**
         * java.home Java 安装目录
         */
        map.put("java_home", System.getProperty("java.home"));
        /**
         * java.vm.specification.version Java 虚拟机规范版本
         */
        map.put("java_vm_specification_version", System.getProperty("java.vm.specification.versi
n"));
        /**
         * java.vm.specification.vendor Java 虚拟机规范供应
         */
        map.put("java_vm_specification_vendor", System.getProperty("java.vm.specification.vend
```

```
r"));
        /**
         *  java.vm.specification.name Java 虚拟机规范名称
         */
        map.put("java_vm_specification_name", System.getProperty("java.vm.specification.name"
);

        /**
         * java.vm.version   Java 虚拟机实现版本
         */
        map.put("java_vm_version", System.getProperty("java.vm.version"));
        /**
         * java.vm.vendor   Java 虚拟机实现供应商
         */
        map.put("java_vm_vendor", System.getProperty("java.vm.vendor"));
        /**
         * java.vm.name   Java 虚拟机实现名称
         */
        map.put("java_vm_name", System.getProperty("java.vm.name"));
        /**
         * java.specification.version   Java 运行时环境规范版本
         */
        map.put("java_specification_version", System.getProperty("java.specification.version"));
        /**
         * java.specification.vendor   Java 运行时环境规范供应商
         */
        map.put("java_specification_vendor", System.getProperty("java.specification.vendor"));
        /**
         * java.specification.name   Java 运行时环境规范名称
         */
        map.put("java_specification_name", System.getProperty("java.specification.name"));
        /**
         * java.class.version   Java 类格式版本号
         */
        map.put("java_class_version", System.getProperty("java.class.version"));
        /**
         * java.class.path   Java 类路径
         */
        map.put("java_class_path", System.getProperty("java.class.path"));
        /**
         * java.library.path   加载库时搜索的路径列表
         */
        map.put("java_library_path", System.getProperty("java.library.path"));
        /**
         * java.io.tmpdir   默认的临时文件路径
         */
        map.put("java_io_tmpdir", System.getProperty("java.io.tmpdir"));
        /**
         * java.compiler   要使用的 JIT 编译器的名称
         */
        map.put("java_compiler", System.getProperty("java.compiler"));
        /**
         * java.ext.dirs   一个或多个扩展目录的路径
         */
```

```
map.put("java_ext_dirs", System.getProperty("java.ext.dirs"));
/**
 * os.name  操作系统的名称
 */
map.put("os_name", System.getProperty("os.name"));
/**
 * os.arch  操作系统的架构
 */
map.put("os_arch", System.getProperty("os.arch"));
/**
 * os.version  操作系统的版本
 */
map.put("os_version", System.getProperty("os.version"));
/**
 * file.separator  文件分隔符（在 UNIX 系统中是“/”）
 */
map.put("file_separator", System.getProperty("file.separator"));
/**
 * path.separator  路径分隔符（在 UNIX 系统中是“:”）
 */
map.put("path_separator", System.getProperty("path.separator"));
/**
 * line.separator  行分隔符（在 UNIX 系统中是“/n”）
 */
map.put("line_separator", System.getProperty("line.separator"));
/**
 * user.name  用户的账户名称
 */
map.put("user_name", System.getProperty("user.name"));
/**
 * user.home  用户的主目录
 */
map.put("user_home", System.getProperty("user.home"));
/**
 * user.dir  用户的当前工作目录
 */
map.put("user_dir", System.getProperty("user.dir"));

/**
 * file_encoding 获取JVM编码格式
 */
map.put("file_encoding", System.getProperty("file.encoding"));

/**
 *  获取操作系统的编码
 */
map.put("sun_jnu_encoding", System.getProperty("sun.jnu.encoding"));
InetAddress netAddress = getInetAddress();
/**
 *  host_ip 获取本地IP
 */
map.put("host_ip", getHostIp(netAddress));

/**
```

```
     * host_name 获取本地主机名
     */
    map.put("host_name", getHostName(netAddress));


    Locale locale = Locale.getDefault();
    /**
     * language 获取语言
     */
    map.put("language",locale.getLanguage());

    /**
     * country 获取语言国家
     */
    map.put("country",locale.getCountry());
    return map;
}

/**
 * 获取本地主机
 *
 * @return
 */
public static InetAddress getInetAddress() {
    try {
        return InetAddress.getLocalHost();
    } catch (UnknownHostException e) {
        logger.error("UnknownHostException", e);
    }
    return null;
}

/**
 * 通过InetAddress获取本地Ip
 *
 * @param netAddress
 * @return
 */
public static String getHostIp(InetAddress netAddress) {
    if (null == netAddress) {
        return null;
    }
    String ip = netAddress.getHostAddress();
    return ip;
}

/**
 * 通过InetAddress获取本地主机的名字
 *
 * @param netAddress
 * @return
 */
public static String getHostName(InetAddress netAddress) {
    if (null == netAddress) {
```

```
            return null;
        }
        String name = netAddress.getHostName();
        return name;
    }

    /**
     * 通过Jdbc的方式获取数据库的版本
     *
     * @return
     */
    protected Map getDatabaseMajorVersion() {
        Map<String, String> map = new HashMap<String, String>();
        PropertyUtil propertyUtil = new PropertyUtil();
        Properties props = new Properties();

        try {

            /**
             * 从配置文件读取 链接数据库信息
             * 然后建立连接
             * 用适当的驱动程序连接到DBMS，看下面的代码[自行修改您所连接的数据库相关信息]：
             */
            props.load(SystemInfoUtil.class.getClassLoader().getResourceAsStream("jdbc.propertie
"));
            String url = props.getProperty("jdbcUrl");
            String user = props.getProperty("username");
            String password = props.getProperty("password");
            String driver = props.getProperty("driverClasss");
            // 加载驱动程序
            // 下面的代码为加载JDBD-ODBC驱动程序
            Class.forName(driver);
            // 用url创建连接
            Connection con = DriverManager.getConnection(url, user, password);
            // 获取数据库的信息
            DatabaseMetaData dbMetaData = con.getMetaData();

            /**
             * 返回一个String类对象，代表数据库的URL
             */
            map.put("URL", dbMetaData.getURL());

            /**
             * 返回连接当前数据库管理系统的用户名。
             */
            map.put("userName", dbMetaData.getUserName());

            /**
             * 返回一个boolean值，指示数据库是否只允许读操作。
             */
            map.put("isReadOnly", dbMetaData.isReadOnly() + "");

            /**
             * 返回数据库的产品名称。
```

```
            */
            map.put("DatabaseProductName", dbMetaData.getDatabaseProductName());

            /**
             * 返回数据库的版本号。
             */
            map.put("DatabaseProductVersion", dbMetaData.getDatabaseProductVersion());


            Method gdbmvMethod = DatabaseMetaData.class.getMethod("getDatabaseMajorVer
ion", null);
            /**
             * 数据库真正的版本号
             */
            map.put("version", gdbmvMethod.invoke(dbMetaData, null)+"");

            /**
             * 返回驱动驱动程序的名称。
             */
            map.put("DriverName", dbMetaData.getDriverName());

            /**
             * 返回驱动程序的版本号。
             */
            map.put("DriverVersion", dbMetaData.getDriverVersion());
            // 关闭连接
            con.close();
        } catch (Exception e) {
            // 输出异常信息
            logger.error("Exception", e);
        }
        return map;
    }

    /**
     * 项目信息
     *
     * @return
     */
    protected Map getProjectInfo() {
        Map<String, String> map = new HashMap<String, String>();
        Properties props = new Properties();
        try {
            /**
             * 从配置文件中获取相关网站信息
             */
            props.load(SystemInfoUtil.class.getClassLoader().getResourceAsStream("common.pro
erties"));
        } catch (IOException e) {
            logger.error("IOException", e);
        }

        /**
         * 项目版本
```

```
         */
        map.put("project_version",props.getProperty("project.version"));
        /**
         * 项目编码
         */
        map.put("project_build_sourceEncoding",props.getProperty("project.build.sourceEncodi
g"));
        SimpleDateFormat time=new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");

        /**
         * 项目启动日期
         */
        map.put("start_date",time.format(new Date()));
        return map;
    }


}
```

代码下载地址

[url=https://bhusk99.ctfile.com/fs/12729178-209651307]SystemInfoUtil.java[/url]