# java中Unicode码与中文的转化

## 什么是unicode码

unicode（统一码、万国码、单一码）是计算机科学领域里的一项业界标准,包括字符集、编码方案等。Unicode 是为了解决传统的字符编码方案的局限而产生的，它为每种语言中的每个字符设定了统一并且唯一的二进制编码，以满足跨语言、跨平台进行文本转换、处理的要求。1990年开始研发，1994年正式公布。unicode码是一种标准，utf-8是一种unicode的实现方式之一。

## Unicode码与中文的转化

Unicode码和中文的转化一般有两种方式：

1. 通过jdk自带的工具native2ascii，通过命令行进行中文和unicode码的转化。
2. 通过java代码进行转化。

此处，关于第一种方式，不介绍，介绍一下通过java代码的转化，测试代码如下：

package com.tooklili.service.test.dataoke;

import org.apache.commons.lang.StringUtils;

import org.junit.Test;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

/\*\*

\* unicode码转换测试

\*

\* @author shuai.ding

\*

\* @date 2017年5月31日下午5:41:15

\*/

public class UnicodeTest {

private static final Logger LOGGER = LoggerFactory.getLogger(UnicodeTest.class);

/\*\*

\* unicode编码测试

\*

\* @author shuai.ding

\*/

@Test

public void unicodeEncodeTest() {

String s = "你好";

String unicode = string2Unicode(s);

LOGGER.info(unicode);

}

/\*\*

\* unicode解码测试

\*

\* @author shuai.ding

\*/

@Test

public void unicodeDecodeTest() {

// 在java中"\"是转义字符，所以防止转义，必须写成下面的形式

// String s = "\\\u0061\\u0041\\\u4f60\\\u597d\\\u0024\\\u006e\\\u0067\\\u006e\\\u0031";

String s = "\\u5168\\u7ad9\\u63a5\\u53e3\\u5df2\\u5347\\u7ea7\\u4e3a\\u5206\\u9875\\u6a21\\u5f0f\\uff0c\\u6bcf\\u987550\\u6761\\u6570\\u636e\\uff0c\\u5206\\u9875\\u53c2\\u6570\\uff1a&page";

System.out.println(s);

String str = unicode2String(s);

LOGGER.info(str);

}

/\*\*

\* 将字符串转化成unicode码

\* @author shuai.ding

\* @param string

\* @return

\*/

private String string2Unicode(String string) {

if (StringUtils.isBlank(string)) {

return null;

}

char[] bytes = string.toCharArray();

StringBuffer unicode = new StringBuffer();

for (int i = 0; i < bytes.length; i++) {

char c = bytes[i];

// 标准ASCII范围内的字符，直接输出

if (c >= 0 && c <= 127) {

unicode.append(c);

continue;

}

String hexString = Integer.toHexString(bytes[i]);

unicode.append("\\u");

// 不够四位进行补0操作

if (hexString.length() < 4) {

unicode.append("0000".substring(hexString.length(), 4));

}

unicode.append(hexString);

}

return unicode.toString();

}

/\*\*

\* 将unicode码转化成字符串

\* @author shuai.ding

\* @param unicode

\* @return

\*/

private String unicode2String(String unicode) {

if (StringUtils.isBlank(unicode)) {

return null;

}

StringBuilder sb = new StringBuilder();

int i = -1;

int pos = 0;

while ((i = unicode.indexOf("\\u", pos)) != -1) {

sb.append(unicode.substring(pos, i));

if (i + 5 < unicode.length()) {

pos = i + 6;

sb.append((char) Integer.parseInt(unicode.substring(i + 2, i + 6), 16));

}

}

//如果pos位置后，有非中文字符，直接添加

sb.append(unicode.substring(pos));

return sb.toString();

}

}