

java认证:浅谈Java中的编码理论Java认证考试 PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/582/2021\\_2022\\_java\\_E8\\_AE\\_A4\\_E8\\_AF\\_81\\_c104\\_582213.htm](https://www.100test.com/kao_ti2020/582/2021_2022_java_E8_AE_A4_E8_AF_81_c104_582213.htm) Java是使用Unicode字符集的，意思是在内存运行的时候是这样的。当Java源程序被编译为.class文件的时候，是以UTF-8字符集存储字符的。比如字符串"汉语"的Unicode字符为：0x49 6C ED 8B。在.class文件中则变成了：0xE6 B1 89 E8 AF AD 0D 0A。实际上如果用以下程序输出“汉语”二个字的16进制字节序列：

```
String str = "汉语";
try{byte[] b = str.getBytes("Unicode").for(int i :
b){System.out.println(Integer.toHexString(i)).} }catch(Exception
ex){ex.printStackTrace().} 
```

会发现，实际输出的是 0xFE FF 6C 49 8B ED。多出来的0xFE FF，是Unicode字符串的BOM（Byte Order Mark）。但是为什么输出的顺序（0xFE FF 6C 49 8B ED）和文件中的顺序不一致（0xFF FE 49 6C ED 8B）？

```
public
byte[] getBytes(Charset charset) 
```

使用指定的字符集将此String解码为字节序列，并将结果存储到一个新的字节数组中。如果不指定参数，则使用系统默认的字符集，对于简体中文系统一般为GBK。看程序：

```
class EncodingExer {public static void
main(String[] args) {String str = "汉语". //GBK编码为： 0xBABA
0xD3EFbyte[] b = str.getBytes(). for(int i :
b){System.out.println(Integer.toHexString(i)). //0xBA 0xBA 0xD3
0xEF} }} 
```

在这个过程中，由Unicode字符得到字符串“汉语”，然后解码得到了GBK字节序列。与String.getBytes（）相对应的是new String（byte[] bytes，Charset charset），它的作用是用字节序列生成字符串。如果想由刚才的GBK字节序列得到

原来的字符串“汉语”，我们需要做的是：`new String ( b )`就可以了。将**b**按照GBK编码得到字符串“汉语”，然后转化为Unicode字符。如果是`str.getBytes ( "iso-8859-1" )`，那么还原的时候就是`new String ( b , "iso-8859-1" )`。总之就是用什么方式编码，就用什么方式解码。更多优质资料尽在百考试题论坛 百考试题在线题库 java认证更多详细资料 100Test 下载频道 开通，各类考试题目直接下载。详细请访问 [www.100test.com](http://www.100test.com)