计算机常见编码

一. 有关编码的基础知识

1. 位 bit 最小的单元
   字节 byte 机器语言的单位
   1byte=8bits
   1KB=1024byte
   1MB=1024KB
   1GB=1024MB
2. 二进制 binary
   八进制 octal
   十进制 decimal
   十六进制 hex
3. **字符**：是各种文字和符号的总称，包括各个国家的文字，标点符号，图形符号，数字等。
   字符集：字符集是多个符号的集合，每个字符集包含的字符个数不同。
   字符编码：字符集只是规定了有哪些字符，而最终决定采用哪些字符，每一个字符用多少字节表示等问题，则是由编码来决定的。计算机要准确的处理各种字符集文字，需要进行字符编码，以便计算机能够识别和存储各种文字。

二. 常见字符集的编码介绍：

常见的字符集有：ASCII 字符集，GB2312 字符集，BIG5 字符集，GB18030 字符集，Unicode 字符集，下面一一介绍：

1. ASCII 字符集：

- **定义：**

美国信息互换标准代码，是基于罗马字母表的一套电脑编码系统，主要显示英语和一些西欧语言，是现今最通用的单字节编码系统。

- **包含内容：**

控制字符（回车键，退格，换行键等）
可显示字符（英文大小写，阿拉伯数字，西文符号）
扩展字符集（表格符号，计算符号，希腊字母，拉丁符号）

- **编码方式：**

第 0-31 号及 127 号是控制字符或通讯专用字符；
第 32-126 号是字符，其中 48-57 号为 0-9 十个阿拉伯数字，65-90 号为 26 个大写英文字母，97-122 号为 26 个英文小写字母，其余为一些标点符号，运算符号等。
在计算机存储单元中，一个 ASCII 码值占一个字节（8 个二进制位），最高位是用作奇偶检验位。【奇偶校验是指：在代码传送的过程中，用来检验是否出错的一种方法。】奇偶校验分为奇校验和偶校验。奇校验规定：正确的代码一个字节中 1 的个数必须是奇数，若非奇数，则在最高位添 1；偶校验规定：正确的代码一个字节中 1 的个数必须是奇数，若非奇数，则在最高位添 1。

2. GB2312 字符集：

- **定义：**

信息交换用汉字编码字符集。是**中国标准的简体中文字符集**，它所收录的汉字已经覆盖 99.75%的使用频率，在中国大陆和新加坡广泛使用。

- **包含内容：**

GB2312 收录了简化汉字及一般字符，序号，数字，拉丁字母，日文假名，希腊字母，俄文字母，汉语拼音符号，汉语注音字母，共 7445 个图形字符。其中包括 6763 个汉字，一级汉字 3755 个，二级汉字 3008 个。

- **编码方式：**

GB2312 对所收汉字进行了“分区”处理，每区含有 94 个汉字或者符号，这种表示方法也叫做“区位码”。

它是用双字节表示的，前面的字节为第一字节，又称“高字节”，后面的为第二字节，“低字节”。

高位字节，把 01-87 区的区号加上 0xA0（相当于数字 160）；低位字节把 01-94 区的区号加上 0xA0（相当于数字 160）。 举个简单的小例子：第一个汉字——“啊”，它的区号为 16，位号 01，则区位码是 1601。则高字节位：16+0xA0=0xB0；低字节位：01+0xA0=0xA1，所以“啊”的汉字处理编码为 0xB0A1。

3. GBK 字符集：

- **定义：**

GBK 是 GB2312 字符集的扩展（K）(**中国的中文编码表升级，融合了更多的中文文字符号。**)，它收录了 21886 个符号，它分为汉字区和图形符号区，汉字区包括 21003 个字符。GBK 字符集主要扩展了繁体中文字的支持。

4. BIG5 字符集：

- **定义：**

又称大五码，由台湾五家软件公司创立。因为当时台湾没有一个标准的字符集，而且 GB2312 又没有收录繁体字，所以才推出了 BIG5。

- **包含内容：**

BIG5 字符集共收录了 13053 个中文字，该字符集在台湾使用。但是没有考虑到社会上流通的人名，地方用字，方言用字，化学及生物科等用字，没有包含日文平假名及片假字母。

- **编码方式：**

BIG5 也采用双字节存储方法，一两个字节编码一个字。高位字节的编码范围是 0xA1-0xF9,低位字节的编码范围是 0xA1-0xFE。

5. GB18030 字符集：

- **定义：**

GB18030 字符集标准解决汉字，日文假名，朝鲜语和中国少数民族文字组成的大字符集计算机编码问题。

- **包含内容：**

该标准的字符总编码空间超过 150 万个编码位，收录了 27484 个汉字，覆盖

中文，日文，朝鲜语和中国少数民族文字。满足中国大陆，香港，台湾，日本和韩国等东南亚地区信息交换多文种，大字量，多用途，统一编码格式的要求。

- **编码方式：**

GB8030 标准采用单字节，双字节和四字节三种方式对字符编码。 单字节部分使用 0x00-0x7F 码（对应于 ASCII 码的相应码）；
双字节部分，首字节码从 0x81-0xFE,尾字节码分别是 0x40-0x7E 和 0x80-0xFE。
四字节部分采用 0x30-0x39 作为双字节编码扩充的后缀，这样扩充的四字节编码，其范围是 0x81308130-0x0xFE39FE39,其中第一，三个字节编码位均为 0x81-0xFE，第二，四个为 0x30-0x39。

6. **ISO8859-1**：拉丁码表。欧洲码表

用一个字节的 8 位表示。

7. Unicode 字符集：

- **定义：**

（国际标准码，融合了多种文字。所有文字都用两个字节来表示,Java 语言使用的就是 unicode） University multiple-object coded character set（通用多八位编码字符集），支持世界上超过 650 种语言的国际字符。Unicode 允许在同一服务器上混合使用不同语言，它为每种语言的每个字符设定了统一并且唯一的二进制编码，以满足跨平台，跨语言进行文本转换，处理的要求。

- **编码方式：**

Unicode 标准始终使用十六进制数字，固定使用 2 个字节来表示一个字符，共可以表示 65536 个字符。而且书写时在前面加上前缀“U+”，例如 A 的编码是 004116，则书写成“U+0041”。

- **Unicode 字符集包含的编码方案：**
  - **UTF-8：（最多用三个字节来表示一个字符。）**

    UTF8 是 unicode 其中的一个使用方式。UTF 的意思是：unicode translation format，即把 unicode 转作某种格式的意思。UTF-8 使用可变长度字节来存储 unicode 字符，如 ASCII 字母还是采用一个字符来存储，希腊字母等采用 2 个字符来存储，而常用的汉字要使用 3 字节，辅助平面字符则使用 4 字节。
  - **UTF-16：**

    使用一个或两个未分配的 16 位代码单元的序列对 unicode 代码点进行编码，即 2 个字节表示一个字符。
  - **UTF-32：**

    将每一个 unicode 代码点表示为相同值的 32 位整数。
  - **关于 unicode 编码的一个问题：**

    使用记事本另存为时，可以在 ANSI，GBK,Unicode，unicode big endian 和 UTF-8 这几种编码之间相互转换。同样是 txt 文件，windows 是怎么识别编码的呢？

    答：平时注意的话可以发现 Unicode，unicode big endian 和 UTF-8 编码的 txt 文件的开头会多出几个字节，分别是(FF,FE),（FE,FF），（EF,BB,BF）。那么这些标记都是基于什么标准呢？

ANSI 字符集：ASCII 字符集，以及由此派生并兼容的字符集。

UTF-16 与 UTF-8：如“连通”两个字，在 UTF-16 中为：DE 8F 1A 90，两个字节决定一个汉字；在 UTF-8 中则为：E8 BF 9E E9 80 9A，即 3 个字节决定一个字符。

当一个软件打开一个文本时，首先是要决定这个文本究竟是使用哪种字符集的哪种编码保存的，软件一般采用三种方式来决定文本的字符集和编码：检测文件头标识，提示用户选择，根据一定的规则猜测。不同编码方式的开头字节如下：

EF BB BF	UTF-8
FF FE	UTF-16，little endian
FE FF	UTF-16，big endian
FF FE 00 00	UTF-32，little endian
00 00 FE FF	UTF-32，big endian

注：endian 是指字节序，big endian（大尾）和 little endian（小尾）是 CPU 处理多字节数的不同方式。例如“汉”的 unicode 编码是 6C49，写到文件中，如果将 6C 写在前面就是 big endian，将 49 写在前面就是 little endian。

8. 总结：

从 ASCII，GB2312，GBK 到 GB18030，这些编码方法是向下兼容的，即同一个字符在这些方案中总是有相同的编码，后面的标准支持更多的字符。在这些编码中，英文和中文可以统一的处理。区分中文编码的方法是高字节的最高位不为 0。

计算机使用的缺省编码方式就是计算机的内码。有的中文 windows 的缺省内码还是 GBK，可以通过 GB18030 升级包升级到 GB1030。不过相对 GBK 新增的字符，普通人很难用到的，通常我们用 GBK 来指代中文 windows 内码。

GB2312 的原文是区位码，从区位码到内码，需要在高字节和低字节上分别加上 A0。