1. **java中void main(String[]args)是什么意思？**
2. **数字（byte，short，int，long，float，double）**

**2.1基本数字：**

byte：1字节，8位

short：2字节，16位

int：4字节，16位

long：8字节，32位

float：数字+f或F或不加

double：数字+d或D或不加

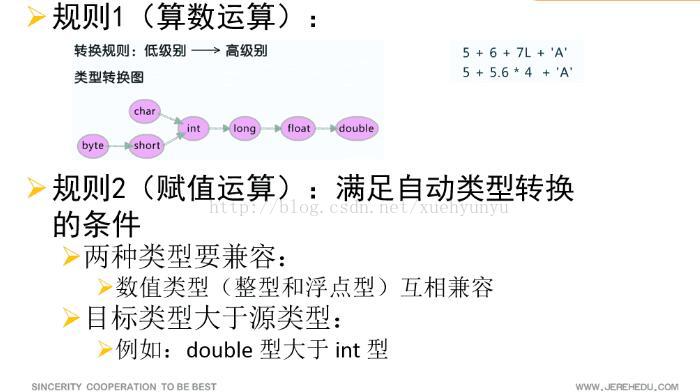
**2.2二进制，八进制，十六进制**

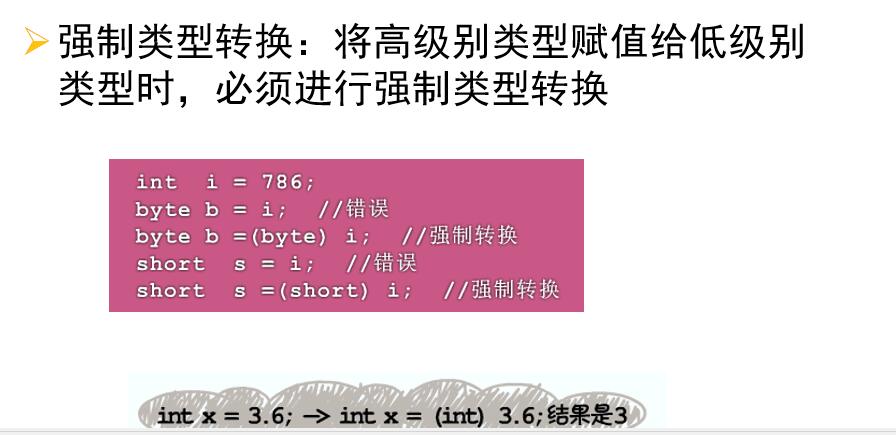
二进制：0b或者0B开头+0或1

八进制：0开头+0-7

十六进制：0x或0X开头+0-9和a-f

**2.3数据转换规则**





1. **运算符**

**3.1Java的位运算符详解实例——与（&）、或（|）、非（~）、异或（^）**

[**https://www.cnblogs.com/shuaiding/p/11124974.html**](https://www.cnblogs.com/shuaiding/p/11124974.html)

**3.1.1 与（&）：既可以比较数字又可以比较布尔值**

**比较数字时：**先将数字转换为二进制，两个操作数中位都为1，结果才为1，否则结果为0。

**128&129 = 128**

129转换成二进制就是10000001， 128转换成二进制就是10000000。根据与运算符的运算规律，只有两个位都是1，结果才是1，可以知道结果就是10000000，即128。

**63&64=0**

* + 1. **或（|）**

与“与（&）”类似，将两个数转化为二进制，两个位只要有一个为1，那么结果就是1，否则就为0。

**例如：128|129=129**

**3.1.3 非（~）**

将两个数转化为二进制，如果位为0，结果是1，如果位为1，结果是0。

**~a = -(a+1)**

**正负数通用！**

**补充：Java中原码、补码与反码！**

<https://www.baidu.com/link?url=8MLVxmooqB3s1rMEjJiWKG-F7feEBbF_zo_AAbOtj1jy3xw8dFwR_eLk16HdGK_2OE30PmXNmJrb2J9ktLwBayVHYqXwNJngACz6gax-zBy&wd=&eqid=ed0e1ab10004cce1000000065e48ec23>

对于正数：原码=反码=补码

对于负数：

原码=正数的原码-最高位0变成1

反码=负数的原码-除最高位1不变，其他位01颠倒

补码=负数的反码+1

负数的运算：不使用减法，使用负数的补码作运算。

**3.1.3 异或（^）**

两个数字转换为二进制，两个操作数的位中，相同则结果为0，不同则结果为1。

2^15 = 13