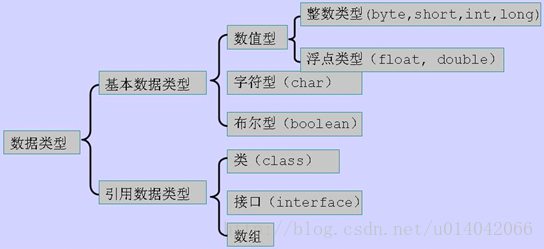
**1.java中几种基本类型各占用多少字节？**



**基本类型：**

byte 1字节 8 bit

short 2字节

int 4字节

long 8字节

float 4字节

double 8字节

char 2字节

boolean 1bit

**对象引用数据类型：**

class

interface

数组

枚举

**JVM4种引用类型：**

强引用，不会被回收

软引用，内存不足时回收

弱引用，一个对象扫描到只有弱引用会被回收

虚引用，任何时候都可能被回收

基本数据类型直接存储在栈上

引用类型对象存储在堆上

**2.整数的存储**

正数：最高位为符号位0，以二进制存储。

负数：最高位为符号位1，以补码形式存储。

-n=(n取反)+1

3=0000 0011，取反1111 1100，加1得11111101=-3

-3=1111 1101，取反0000 0010，加1得0000 0011=3

**3.浮点数的存储**

浮点数存储分为2部分，小数点左边整数和小数点右面的小数。

例如：78.375

小数点左边的整数和整数的存储一致。

https://img2018.cnblogs.com/blog/1655563/201904/1655563-20190409153745773-384204118.png

小数点右边的小数

https://img2018.cnblogs.com/blog/1655563/201904/1655563-20190409153914709-2146580616.png

4.String， Stringbuffer， StringBuilder 的区别。

String 字符串常量(final修饰，不可被继承)，String是常量，当创建之后即不能更改。(可以通过StringBuffer和StringBuilder创建String对象(常用的两个字符串操作类)。)

==StringBuffer 字符串变量（线程安全）,==其也是final类别的，不允许被继承，其中的绝大多数方法都进行了同步处理，包括常用的Append方法也做了同步处理(synchronized修饰)。其自jdk1.0起就已经出现。其toString方法会进行对象缓存，以减少元素复制开销。

public synchronized String toString() {

if (toStringCache == null) {

toStringCache = Arrays.copyOfRange(value, 0, count);

}

return new String(toStringCache, true);

}

==StringBuilder 字符串变量（非线程安全）==其自jdk1.5起开始出现。与StringBuffer一样都继承和实现了同样的接口和类，方法除了没使用synch修饰以外基本一致，不同之处在于最后toString的时候，会直接返回一个新对象。

public String toString() {

// Create a copy, don’t share the array

return new String(value, 0, count);

}

5. 讲讲类的实例化顺序，比如父类静态数据，构造函数，字段，子类静态数据，构造函数，字段，当 new 的时候， 他们的执行顺序。