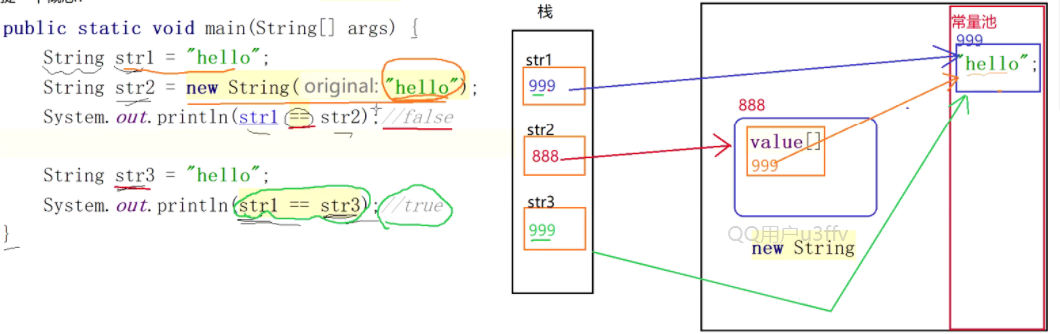
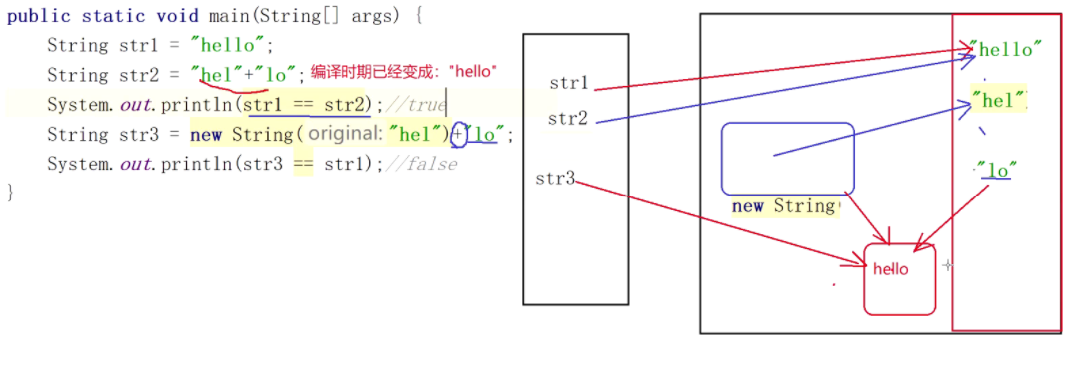
**1.对象值相等的比较**

**==** 与 **equals** 的区别：

1. ”==”：比较的是两个字符串对象地址。

2. “equals（）”：不覆写时与“==”一样，比较地址，覆写之后比较字符串的内容





**2.对象值大于、等于、小于的比较**

Comparable——基于自然顺序(写死的）

< 0: 表示 this 指向的对象小于 o 指向的对象

== 0: 表示 this 指向的对象等于 o 指向的对象

> 0: 表示 this 指向的对象等于 o 指向的对象

public interface Comparable<E> {
int compareTo(E o);
}

Comparator——基于比较器

< 0: 表示 o1 指向的对象小于 o2 指向的对象

== 0: 表示 o1 指向的对象等于 o2 指向的对象

> 0: 表示 o1 指向的对象等于 o2 指向的对象

public interface Comparator<T> {
int compare(T o1, T o2);
}

**两个接口区别：**

comparable接口基于自然顺序，直接使用类实现接口，覆写compareTo方法

p.compareTo(q);直接调用

comparator接口基于比较器，需要根据不同的比较基准，写相应的类实现comparator接口，覆写compare方法

comparator<T> 名称=new 类名（）；
名称.compareTo（p，q)

3.StringInteger中包含接口comparable 所以他两为自然顺序（规定）

为什么使用comparator接口？

因为有些类没有自然顺序或者是自然顺序不否合需要

比如：Person的自然顺序是以age为基准的想要其他的就必须自己建比较器