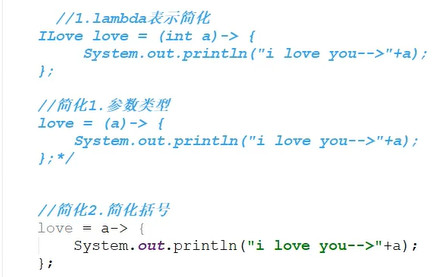
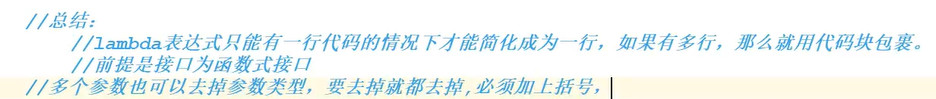
Lambda表达式实际上就是简化接口实现内部匿名类的写法。（Java能自动进行接口类型的识别，可以省略具体的接口类型，只需要写指定具体方法的参数值即可。（参数值，参数值……）->{具体实现的函数体}）

[8. 方法引用入门-Lambda函数式编程-攀博课堂自学Java教程 (pbteach.com)](http://www.pbteach.com/course/20/1412773983350165506.html)







[【IntelliJ IDEA】快捷键 - Angel挤一挤 - 博客园 (cnblogs.com)](https://www.cnblogs.com/sxdcgaq8080/p/7650804.html)

**在注解@FuctionalInterface的javadoc中如下说明：An informative annotation type used to indicate that an interface type declaration is intended to be a functional interfaceas defined by the Java Language Specification. Conceptually, a functional interface has exactly one abstract method. Since default methods have an implementation, they are not abstract. If an interface declares an abstract method overriding one of the public methods ofjava.lang.Object, that also does not count toward the interface's abstract method count since any implementation of the interface will have an implementation fromjava.lang.Objector elsewhere.从中我们可以得知函数式接口的几点特征：函数式接口只有一个抽象方法default方法某默认实现，不属于抽象方法接口重写了Object的公共方法也不算入内（签名相同）所以，Comparator虽然有两个抽象方法：int compare(T o1, T o2);boolean equals(Object obj);其中 equals为Object的方法，不算入内，所以Comparator可以作为函数式接口。**