lambda表达式（为支持函数式编程）：将接口的抽象方法实现（**接口中有且仅有一个抽象方法**），可由实现接口的匿名内部类优化而来。

1.如果抽象方法的实现只有一条命令，大括号可以省略

2.如果抽象方法的参数只有一个，参数列表的括号可以省略

3.如果抽象方法中有返回值，并且只有一条语句，就是此返回值语句

4．多条语句（返回值+实现语句/多条实现语句）的情况下，大括号不能省略

5.在lambda表达式所在的作用域下，参数列表中的参数名不能和作用域中的其他变量名冲突

标准写法，不简写：

接口名 对象名=（[参数列表]）->{实现语句；};

方法引用（与lambda表达式搭配使用）：本质上就是给现有方法起了个别名

1. 引用类中静态方法

类名称：：静态方法名称

1. 引用某个对象的方法

实例化对象：：普通方法

1. 引用类中普通方法

类名称：：普通方法名

1. 引用类中构造方法

类名称：：new

如果接口中只有一个抽象方法，并且希望其以后被用作lambda表达式，可在接口类前加上注解@FunctionalInterface，这样IDE就会自动检测是否只有一个抽象方法,存在两个抽象方法以上就会报错。

现代编程语言：能充分利用计算机资源的多核cpu，使硬件充分发挥作用；Stream、并发、并行计算

System.out.println();中out是对象