

```
@FunctionalInterface
public interface Calculable {
    double calc(double x, double y);
}
```

```
public class Person {
    public void action(Calculable calculable) {
        double result = calculable.calc(10, 4);
        System.out.println("결과: " + result);
    }
}
```

실행 결과

결과: 14.0

결과: 14.0

결과: 14.0

```
public class LambdaExample {
    public static void main(String[] args) {
        Person person = new Person();

        //실행문이 두 개 이상일 경우
        person.action((x, y) -> {
            double result = x + y;
            return result;
        });

        //리턴문이 하나만 있을 경우(연산식)
        //person.action((x, y) -> {
        //    return (x + y);
        //});
        person.action((x, y) -> (x + y));

        //리턴문이 하나만 있을 경우(메소드 호출)
        //person.action((x, y) -> {
        //    return sum(x, y);
        //});
        person.action((x, y) -> sum(x, y));
    }

    public static double sum(double x, double y) {
        return (x + y);
    }
}
```