

```
@FunctionalInterface
public interface Workable {
    void work(String name, String job);
}
```

```
@FunctionalInterface
public interface Speakable {
    void speak(String content);
}
```

```
public class Person {
    public void action1(Workable workable) {
        workable.work("홍길동", "프로그래밍");
    }

    public void action2(Speakable speakable) {
        speakable.speak("안녕하세요");
    }
}
```

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

        //매개변수가 두 개일 경우
        person.action1((name, job) -> {
            System.out.print(name + "이 ");
            System.out.println(job + "을 합니다.");
        });
        person.action1((name, job) -> System.out.println(name + "이 " +
            job + "을 하지 않습니다.));

        //매개변수가 한 개일 경우
        person.action2(word -> {
            System.out.print "\"" + word + "\"";
            System.out.println("라고 말합니다.");
        });
        person.action2(word -> System.out.println "\"" + word + "\"라고 외칩니다.));
    }
}
```

실행 결과

```
홍길동이 프로그래밍을 합니다.
홍길동이 프로그래밍을 하지 않습니다.
"안녕하세요"라고 말합니다.
"안녕하세요"라고 외칩니다.
```
