【案例6-5】 模拟百度翻译器

### 【案例介绍】

**1.任务描述**

相信百度翻译对于大家来说并不陌生，本案例要求编写一个程序模拟百度翻译。用户输入英文之后搜索程序中对应的中文，如果搜索到对应的中文就输出搜索结果，反之给出提示。本案例要求使用Map集合实现英文与中文的存储。

**2.运行结果**

程序运行结果如图6-1所示：

手机屏幕截图

描述已自动生成

图6-1 运行结果图

### 【案例任务】

* 学会分析“模拟百度翻译”任务的实现思路。
* 根据思路独立完成“模拟百度翻译”任务的源代码编写、编译及运行。
* 掌握Map集合特点及常用方法的使用。

### 【案例思路】

（1）百度翻译主要用于翻译对应的意思。这是一种映射关系。因此可以用 Map集合来实现，所以首先就是定义Map集合，存储数据。

（2）用键盘录入功能获取我们要翻译的单词。

（3）定义一个方法，在该方法中实现对单词的查询操作，并且根据不同情况给出相关提示。

（4）调用查询的方法，实现翻译。并将结果输出到控制台。

### 【案例实现】

程序实现代码如文件6-1所示。

文件6-1 example7.java

1. package com.itheima;
2. import java.util.HashMap;
3. import java.util.Scanner;
4. public class example7 {
5. public static void main(String[] args) {
6. //定义Map集合用于存储词典的数据
7. HashMap<String, String> word = new HashMap<String, String>();
8. while(true) {
9. //集合合添加词典的数据。
10. word.put("apple", "苹果");
11. word.put("banner", "香蕉");
12. word.put("book", "书");
13. word.put("telephone", "电话");
14. word.put("cat", "猫");
15. word.put("dog", "狗");
16. word.put("student", "学生");
17. word.put("teacher", "老师");
18. //键盘录入功能获取我们要翻译的单词
19. Scanner sc = new Scanner(System.in);
20. System.out.println("请您输入您要翻译的单词：");
21. String w = sc.nextLine();
22. //调用判断打印输出
23. Select(w, word);
24. }
25. }
26. //定义方法对键盘录入的数据进行判断
27. public static String Select(String w, HashMap<String, String> word) {
28. if(w.isEmpty()) {
29. System.out.println("请输入您要翻译的单词：");
30. return "";
31. }else if(!word.containsKey(w)) {
32. System.out.println("对不起，您要翻译的单词不存在，请重新输入：");
33. }else{
34. String chinese = word.get(w);
35. System.out.println(w+"翻译成中文意思为："+chinese);
36. }
37. return w;
38. }
39. }

在文件6-1中，第7行代码用于定义HashMap集合。第10~17行代码用Map集合的put()方法向集合中添加词典的初始化数据。第19~21行代码键盘录入我们要翻译的英文单词，第23行代码调用Select()方法实现集合中判断是否包含键盘输入的单词，如果有则输入对应的中文意思，反之则给出相应提示。