

JC2002 Java 编程 - 实践 9（第 11 天）

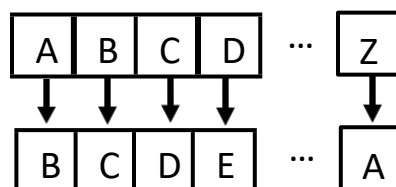
本实用程序将引导您完成一些与 Java 中字符串操作有关的练习。其中大部分都是简单的短程序。对于后面的练习，您可能希望使用 Maven，这样也可以集成测试。如果您使用 Maven，请使用以前实践中的说明。对于今天的所有练习，你应该使用 Java **字符串** 类中已有的方法。

1. 可以从单词倒序开始。编写一个小程序，从用户输入字符串，然后按相反顺序打印出单词。
2. 使用 **regionMatches()** 方法比较字符串的部分内容。编写一个小程序，接收用户输入的两个字符串，然后报告其中是否有匹配的部分。在这项任务中，您可以忽略大小写（大写或小写）。
3. 编写一个程序，从键盘输入一行文本，然后打印一个显示字母出现频率的表格，如下面的输出示例所示。如果某个字母的出现次数为零，则不打印任何内容。大写字母和小写字母应一起计算。

输出示例

```
a: 5
b: 2
c: 1
d: 4
...
z: 1
```

4. 假设个人身份代码的格式为 **DDMMYY-XXXX**，其中 DD 是出生日期（01-31）的数字，MM 是出生月份（01-12）的数字，YY 是出生年份（00-99）的最后两位数字，XXXX 是由四个字符组成的个人标识符，可以包含数字和大写字母。使用 **matches()** 方法和正则表达式（*Regex*）编写一个小程序，测试给定输入是否为有效的个人身份代码。
5. 如下图所示，编写一个程序，从用户处读取一个单词，并使用移位密码进行加密，其中字母被字母表中的下一个字母替换。



提示： 您可以使用以下方法将字符 **a-z** 转换为整数 **0-25**：

```
int i = Character.digit( c, 36 ) - 10;
```

您可以使用以下方法将整数 **0-25** 转换为字符 **a-z**：

```
char c = Character.forDigit( i + 10, 36 );
```

您还可以尝试不同的移动方式，例如将字母移动两个位置、三个位置等。