

## 第六周——输入/输出

### ! Important

第六周题目不完成，这里只提供前三题。

### 题目目的

- （一）掌握 `scan` 函数、`readline` 函数和 `readLines` 函数读取键盘输入。
- （二）掌握 `scan` 函数、`read.table` 函数和 `read.csv` 函数读取文本文件。
- （三）掌握 `write.table` 函数和 `write.csv` 函数保存结构化数据。
- （四）掌握 `print` 函数和 `cat` 函数的使用。
- （五）掌握环境数据的保存与加载。

### 题目

题目一：环境数据的保存与加载。创建脚本文件 `test0601.R`，并完成下面操作。

定义矩阵 `mat` 为

	[,1]	[,2]	[,3]	[,4]
[1,]	5	10	2	17
[2,]	3	8	9	7
[3,]	9	15	13	9

定义字符型变量 `str` 为 `c(“咳嗽”, “头痛”, “低烧”)`, 定义数值型变量 `num` 为 178。

用 `save.image` 保存所有变量, 文件名 `save-all.txt`, 用 `dir` 查看当前工作文件下是否有该文件。

用 `save` 保存 `mat` 和 `str`, 文件名为 `save-part.txt`, 用 `dir` 查看当前工作文件下是否有该文件。

用 `rm` 函数删除所有变量。

用 `ls` 查看当前环境对象, 是否还有变量名列出? 计算 `num` 除以 100 的结果, 会有什么错误?

从文件 `save-all.txt` 中加载数据, 计算 `num` 除以 100 的结果, 还有错误出现吗?

## 题目二：从命令窗口中交互式输入数据。创建脚本文件 `test0602.R`, 完成下面操作。

用 `scan` 函数从命令窗口中读入向量 `{7,13,10,9}`, 赋值给变量 `x`, 然后计算该向量的和与平均值。

用 `readline` 函数读入价格 2319, 其中提示信息为 “请输入价格: ”; 用 `readline` 函数读入数量 12, 其中提示信息为 “请输入数量: ”; 最后计算总金额, 输出格式为:

-----

您应付金额为: ???

付款时间为: \*\*\*

-----

其中??? 为计算出来的结果，\*\*\* 为系统当前日期时间。

用 readLines 函数从命令窗口中输入两个字符串，分别为“R 程序设计”、“Python 程序设计”。

**题目三：屏幕输出与定向。创建脚本文件 test0603.R，完成下面操作。**

字符串向量 poem 包含四个字符串，分别为：“秦时明月汉时关，”，“万里长征人未还。”，“但使龙城飞将在，”，“不教胡马度阴山。” 请用赋值语句定义变量 poem。

请用 cat 函数按下面格式输出。

秦时明月汉时关，  
万里长征人未还。  
但使龙城飞将在，  
不教胡马度阴山。

用 sink 函数把上面的输出定向输出到文件 sink.file.txt 中。

## 答案及解析

**题目一：**

```
# (1)
mat <- matrix(c(5, 3, 9, 10, 8, 15, 2, 9, 13, 17, 7, 9)
              , nrow = 3, ncol = 4) # 定义一个 3x4 的矩阵
str <- c(" 咳嗽", " 头痛", " 低烧")
num <- 178

# (2)
save.image("save-all.txt")
dir()
```

```
[1] "save-all.txt"           "save-part.txt"
[3] "sink.file.txt"          "第六周——输入、输出.html"
[5] "第六周——输入、输出.pdf" "第六周——输入、输出.qmd"
[7] "第六周——输入、输出.rmarkdown"
```

```
# (3)
save(mat, str, file = "save-part.txt")
dir()
```

```
[1] "save-all.txt"           "save-part.txt"
[3] "sink.file.txt"          "第六周——输入、输出.html"
[5] "第六周——输入、输出.pdf" "第六周——输入、输出.qmd"
[7] "第六周——输入、输出.rmarkdown"
```

```
# (4)
rm(list = ls())

# (5)
ls()
```

```
character(0)
```

```
#num / 100
# 将上面的 # 删除会报错

# (6)
load("save-all.txt")
num / 100
```

```
[1] 1.78
```

题目二：

```
#(1)
scan() ->x
sum(x)
```

```
[1] 0
```

```
mean(x)
```

```
[1] NaN
```

```
#(2)
price <- as.integer(readline(prompt = '请输入价格:'))
```

请输入价格:

```
num <- as.integer(readline(prompt = " 请输入数量:"))
```

请输入数量:

```
total <- price*num
current_time <- format(Sys.time(),
                        "%Y 年%m 月%d 日 %H 时%M 分%S 秒")
cat("\n-----\n")
```

-----

```
cat(" 您应付金额为: ",total,'\n')
```

您应付金额为: NA

```
cat(" 付款时间为:",current_time)
```

付款时间为: 2025年08月31日 15时35分45秒

```
cat("\n-----\n")
```

-----

### 题目三:

```
# 实验三：屏幕输出与定向
```

```
# 定义字符串向量 poem
```

```
poem <- c(  
  " 秦时明月汉时关，"  
  " 万里长征人未还。"  
  " 但使龙城飞将在，"  
  " 不教胡马度阴山。"  
)
```

```
# 用 cat 函数按格式输出
```

```
cat(poem, sep = "\n")
```

秦时明月汉时关，

万里长征人未还。

但使龙城飞将在，

不教胡马度阴山。

```
# 使用 sink 函数将输出定向到文件 sink.file.txt
```

```
sink("sink.file.txt") # 开始定向输出
```

```
cat(poem, sep = "\n") # 输出内容
```

```
sink() # 结束定向输出，恢复正常输出
```