

Chapter 1

1.1 添加 UI 控件

- `fluidPage()` 是一个布局函数，用于设置页面的基本视觉结构。
- `selectInput()` 是一个输入控件，允许用户通过提供值与应用程序交互。
- `verbatimTextOutput()` 和 `tableOutput()` 是输出控件，告诉 Shiny 将渲染输出放在哪里。`verbatimTextOutput()` 显示代码并 `tableOutput()` 显示表格。

1.2

您可以通过包装一段代码并将 `reactive({...})` 其分配给变量来创建反应式表达式，并且可以通过像函数一样调用它来使用反应式表达式。但是，虽然看起来您正在调用函数，但响应式表达式有一个重要的区别：它仅在第一次调用时运行，然后缓存其结果，直到需要更新为止。

Chapter 2

基础 UI

2.1 输出

请注意，有两个渲染函数的行为略有不同：

- `renderText()` 将结果组合成一个字符串，并且通常与 `textOutput()`
- `renderPrint()` 打印结果，就像您在 R 控制台中一样，并且通常与 `verbatimTextOutput()`

2.1.1 表格

有两种用于在表中显示数据框的选项：

- `tableOutput()` 与 `renderTable()` 渲染一个静态数据表，一次性显示所有数据。
- `dataTableOutput()` 与 `renderDataTable()` 呈现一个动态表，显示固定数量的行以及用于更改哪些行可见的控件。

`tableOutput()` 对于小型、固定的 summary（例如模型系数）最有用；如果您想向用户公开完整的数据框，则 `dataTableOutput()` 最合适。

2.1.2 绘图

您可以使用 `plotOutput()` 和 `renderPlot()` 显示任何类型的 R 图形（base、ggplot2 或其他）。

2.1.3 下载

您可以让用户使用 `downloadButton()` 或 `downloadLink()` 来下载文件。