

Different output formats available in RMarkdown

This reading will explore the different types of output formats you can produce with R Markdown.

Setting the output of an R Markdown document

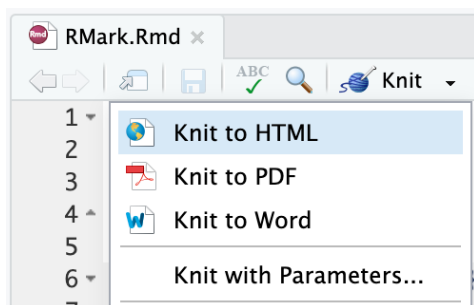
When working in RStudio, you can set the output of a document in R Markdown by changing the YAML header. For example, this code creates an HTML document:

```
---  
title: "Demo"  
output: html_document  
---
```

This code creates a PDF document:

```
---  
title: "Demo"  
output: pdf_document  
---
```

The **Knit** button in the RStudio source editor renders a file to the first format listed in its output field (HTML is the default). You can render to additional formats by clicking the dropdown menu next to the knit button.



Documents

Along with the default `html_document` output, there are a number of other types of documents you can create in R Markdown:

<code>pdf_document</code>	Creates a PDF with LaTeX (an open source document layout system). If you don't already have LaTeX, RStudio will automatically prompt you to install it.
<code>word_document</code>	Creates Microsoft Word documents (.docx)
<code>odt_document</code>	Creates OpenDocument Text documents (.odt)
<code>rtf_document</code>	Creates Rich Text Format documents (.rtf)
<code>md_document</code>	Creates a Markdown document
<code>github_document</code>	Creates a GitHub document. This is a customized version of a Markdown document designed for sharing on GitHub.

For a detailed guide to creating different types of R Markdown documents, check out the [Documents](#) chapter in *RMarkdown: the Definitive Guide*.

Notebooks

A **notebook** (`html_notebook`) is a variation on an HTML document (`html_document`). Overall, the output formats are similar; the main difference between them is that the rendered output of a notebook always includes an embedded copy of the source code.

Notebooks and HTML documents also have different purposes. HTML documents are good for communicating with stakeholders. Notebooks are better for collaborating with other data analysts or data scientists.

To learn more, check out the section on [Notebooks](#) on the R Markdown website.

Presentations

You can also use R Markdown to produce presentations. Automatically inserting the results of your R code into a presentation can save you lots of time.

R Markdown renders to the following presentation formats:

<code>beamer_presentation</code>	For PDF presentations with beamer
<code>ioslides_presentation</code>	For HTML presentations with ioslides
<code>slidy_presentation</code>	For HTML presentations with Slidy
<code>powerpoint_presentation</code>	For PowerPoint presentations
<code>revealjs::revealjs_presentation</code>	For HTML presentations with reveal.js (a framework for creating presentations using HTML, which requires the reveal.js package)

To learn more, check out the section on [slide presentations](#) on the R Markdown website.

Dashboards

Dashboards are a useful way to quickly communicate a lot of information. The [flexdashboard](#) package lets you publish a group of related data visualizations as a dashboard. Flexdashboard also provides tools for creating sidebars, tabsets, value boxes, and gauges.

To learn more, visit the [flexdashboard](#) page and the [Dashboards](#) section on the R Markdown website.

Shiny

Shiny is an R package that lets you build interactive web apps using R code. You can embed your apps in R Markdown documents or host them on a webpage.

To call Shiny code from an R Markdown document, add `runtime: shiny` to the YAML header:

```
---
title: "Shiny Web App"
output: html_document
runtime: shiny
---
```

To learn more about Shiny and how to use R code to add interactive components to an R Markdown document, check out the [Shiny](#) tutorial on the RStudio website.

Other formats

Other packages provide even more output formats:

- The [bookdown](#) package is helpful for writing books and long-form articles.
- The [prettydoc](#) package provides a range of attractive themes for R Markdown documents.
- The [rticles](#) package provides templates for various journals and publishers.

Visit the [RStudio Formats](#) page on the R Markdown website for a more comprehensive list of output formats.

More resources

- Check out this [gallery](#) from the R Markdown website for tons of examples of the outputs you can create with R Markdown.
- This reading pulled information from the discussion of output formats in *R for Data Science*. For more detailed descriptions, check out the chapter on “[R Markdown Formats](#).”