

Quarto for Academic Writing

A Complete Guide to Creating Professional Documents

Your Name

2026-08-30

Table of contents

1	Introduction to Quarto Document Writing	3
2	Setting Up Your Document	3
2.1	The YAML Header	3
2.2	Document Structure	4
3	Working with R Code	5
3.1	Loading Packages	5
3.2	Code Chunk Options	5
3.3	Inline R Code	6
4	Creating Tables	6
4.1	Simple Summary Tables with gt	6
4.2	Grouped Summary Tables	7
4.3	Regression Tables with modelsummary	8
4.4	Multiple Regression Models in One Table	8
5	Mathematical Equations	9
5.1	Inline Math	9
5.2	Display Math (Numbered)	9
5.3	Display Math (Unnumbered)	10
5.4	Common LaTeX Math Symbols	10
6	Cross-Referencing	10
6.1	Referencing Tables	10
6.2	Referencing Figures	11
6.3	Referencing Sections	11

7	Advanced Formatting	12
7.1	Callout Blocks	12
7.2	Column Layouts	12
7.3	Tabbed Content	13
7.4	Custom Styling	13
8	Complete Example: Regression Analysis	14
9	Tips for Academic Writing	16
9.1	Do's	16
9.2	Don'ts	16
10	Workflow Tips	16
10.1	Iterative Rendering	16
10.2	Code Organization	17
10.3	Reproducibility Checklist	17
11	Output Formats	17
11.1	HTML (Interactive)	17
11.2	PDF (Print)	17
11.3	Word (Collaboration)	18
12	Common Issues and Solutions	18
12.1	Tables Not Appearing	18
12.2	Cross-References Not Working	18
12.3	Math Not Rendering	19
12.4	Code Chunks Producing Errors	19
13	Practice Exercise	19
14	Additional Resources	20
14.1	Documentation	20
14.2	Cheat Sheets	20
14.3	Community	20
15	Quick Reference	20
15.1	Essential YAML Options	20
15.2	Code Chunk Options	21
15.3	Cross-References	21
15.4	Math Symbols	21
16	Appendix: Complete Working Example	22

1 Introduction to Quarto Document Writing

This tutorial will teach you how to create professional academic documents using Quarto. We'll recreate a complete regression analysis document, learning key features along the way.

What You'll Learn

- Document structure and YAML configuration
- Working with R code chunks
- Creating tables with `gt` and `modelsummary`
- Mathematical equations with LaTeX
- Cross-referencing figures and tables
- Professional formatting and styling

2 Setting Up Your Document

2.1 The YAML Header

Every Quarto document starts with a YAML header enclosed in `---`. This controls the document's metadata and output format.

Basic YAML:

```
---  
title: "Your Document Title"  
author: "Your Name"  
date: today  
format: html  
---
```

Advanced YAML with Multiple Formats:

```
---  
title: "Regression with Interaction Variables"  
author: "Your Name"  
date: today  
format:  
  html:  
    toc: true  
    toc-depth: 3  
    number-sections: true
```

```
code-fold: show
theme: cosmo
pdf:
  toc: true
  number-sections: true
docx:
  toc: true
---
```

Pro Tip

- Use today for automatic date updates
- `toc: true` adds a table of contents
- `number-sections: true` auto-numbers your sections
- `code-fold: show` makes code collapsible

2.2 Document Structure

A well-structured academic document typically includes:

1. **Introduction** - Context and objectives
2. **Methodology** - Data and methods
3. **Results** - Analysis and findings
4. **Discussion** - Interpretation
5. **Conclusion** - Summary and implications

```
# Introduction

Your introduction text here...

## Background

More specific context...

# Methodology

## Data Description

## Statistical Methods

# Results
```

```
# Discussion
```

```
# Conclusion
```

3 Working with R Code

3.1 Loading Packages

Start your document by loading required packages in a code chunk:

```
```{r}
#| label: setup
#| message: false
#| warning: false

library(tidyverse) # Data manipulation
library(gt) # Beautiful tables
library(modelsummary) # Regression tables
library(carData) # Sample data
```
```

! Best Practice

Use `#| message: false` and `#| warning: false` to suppress package loading messages in your final document.

3.2 Code Chunk Options

Code chunks use `#|` (hashpipe) for options:

```
```{r}
#| label: descriptive-name
#| echo: true # Show code
#| eval: true # Run code
#| warning: false # Hide warnings
#| message: false # Hide messages
#| fig-cap: "Caption for figure"
```

```
Your code here
```
```

Common options:

| Option | Values | Purpose |
|---------|------------|--------------------|
| echo | true/false | Show/hide code |
| eval | true/false | Run/don't run code |
| include | true/false | Include output |
| message | true/false | Show messages |
| warning | true/false | Show warnings |
| error | true/false | Show errors |

3.3 Inline R Code

Mix R code with text using backticks:

```
The dataset contains `r nrow(Salaries)` observations and
`r ncol(Salaries)` variables.

The average salary is $`r round(mean(Salaries$salary), 2)`.
```

This will render as: “The dataset contains 397 observations and 6 variables.”

4 Creating Tables

4.1 Simple Summary Tables with gt

The `gt` package creates publication-quality tables:

```
```{r}
#| label: tbl-summary
#| tbl-cap: "Summary Statistics by Gender"

Salaries %>%
 group_by(sex) %>%
 summarise(avg_salary = mean(salary)) %>%
 gt() %>%
```

```

fmt_currency(columns = avg_salary, decimals = 1) %>%
cols_label(
 sex = "Gender",
 avg_salary = "Average Salary"
)
```

```

💡 GT Table Formatting Functions

- `fmt_currency()` - Format as currency
- `fmt_number()` - Number formatting
- `fmt_percent()` - Percentage formatting
- `cols_label()` - Rename columns
- `tab_header()` - Add title and subtitle

4.2 Grouped Summary Tables

Create tables with multiple grouping variables:

```

```{r}
#| label: tbl-grouped
#| tbl-cap: "Salary by Gender and Discipline"

Salaries %>%
 group_by(sex, discipline) %>%
 summarise(avg_salary = mean(salary), .groups = "drop") %>%
 gt() %>%
 fmt_currency(columns = avg_salary, decimals = 2) %>%
 cols_label(
 sex = "Gender",
 discipline = "Discipline",
 avg_salary = "Average Salary"
) %>%
 tab_header(
 title = "Faculty Salaries",
 subtitle = "By Gender and Discipline"
)
```

```

4.3 Regression Tables with modelsummary

The `modelsummary` package creates professional regression tables:

```
```{r}
#| label: tbl-regression
#| tbl-cap: "Regression Results"

Fit model
model1 <- lm(salary ~ sex, data = Salaries)

Create table
modelsummary(
 model1,
 stars = TRUE,
 gof_map = c("nobs", "r.squared", "adj.r.squared", "rmse")
)
```
```

4.4 Multiple Regression Models in One Table

Compare multiple models side-by-side:

```
```{r}
#| label: tbl-all-models
#| tbl-cap: "Comparison of Regression Models"

Fit multiple models
model1 <- lm(salary ~ sex, data = Salaries)
model2 <- lm(salary ~ sex + discipline, data = Salaries)
model3 <- lm(salary ~ sex * discipline, data = Salaries)
model4 <- lm(salary ~ sex * discipline + rank +
 yrs.since.phd + yrs.service, data = Salaries)

Create comparison table
models <- list(
 "(1)" = model1,
 "(2)" = model2,
 "(3)" = model3,
 "(4)" = model4
)
```

```

modelsummary(
 models,
 stars = TRUE,
 gof_map = c("nobs", "r.squared", "adj.r.squared",
 "aic", "bic", "logLik", "F", "rmse"),
 coef_rename = c(
 "sexMale" = "Male",
 "disciplineB" = "Discipline B",
 "sexMale:disciplineB" = "Male × Discipline B"
)
)
...

```

### **i** Customizing Regression Tables

- `stars = TRUE` - Add significance stars
- `gof_map` - Select goodness-of-fit statistics
- `coef_rename` - Rename coefficients for clarity
- `output = "gt"` - Use gt for more formatting options

## 5 Mathematical Equations

### 5.1 Inline Math

Use single dollar signs for inline equations:

The coefficient  $\beta_1$  represents the effect of gender on salary.

The p-value is  $p < 0.05$ .

### 5.2 Display Math (Numbered)

Use the `equation` environment with tags for numbered equations:

```

$$
salary_i = \beta_0 + \beta_1 sex_i + \epsilon_i \tag{1}
$$

```

This creates:

$$salary_i = \beta_0 + \beta_1 sex_i + \epsilon_i \quad (1)$$

### 5.3 Display Math (Unnumbered)

Use double dollar signs without tags:

```
$$
Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \epsilon
$$
```

### 5.4 Common LaTeX Math Symbols

Symbol	Code	Example
Greek letters	<code>\alpha, \beta, \gamma</code>	$\alpha, \beta, \gamma$
Subscript	<code>x_i</code>	$x_i$
Superscript	<code>x^2</code>	$x^2$
Fractions	<code>\frac{a}{b}</code>	$\frac{a}{b}$
Sum	<code>\sum_{i=1}^n</code>	$\sum_{i=1}^n$
Integral	<code>\int_a^b</code>	$\int_a^b$
Hat	<code>\hat{y}</code>	$\hat{y}$
Bar	<code>\bar{x}</code>	$\bar{x}$

#### Math Mode Tips

- Use `\text{text}` for text within equations:  $\beta_{\text{male}}$
- Use `\ldots` for ellipsis:  $1, 2, 3, \dots, n$
- Enclose matrices in `\begin{bmatrix} \dots \end{bmatrix}`

## 6 Cross-Referencing

### 6.1 Referencing Tables

Add labels to tables with `\label{tbl-name}`:

```

```{r}
#| label: tbl-summary
#| tbl-cap: "Summary Statistics"

# Your table code
```

See @tbl-summary for descriptive statistics.

```

## 6.2 Referencing Figures

Add labels to figures with `#| label: fig-name:`

```

```{r}
#| label: fig-scatter
#| fig-cap: "Salary vs Years Since PhD"

ggplot(Salaries, aes(x = yrs.since.phd, y = salary)) +
  geom_point() +
  theme_minimal()
```

As shown in @fig-scatter, there is a positive relationship.

```

## 6.3 Referencing Sections

Add IDs to section headers:

```

Introduction {#sec-intro}

Methods {#sec-methods}

As discussed in @sec-intro, our objective is...

```

### ! Labeling Rules

- Tables: `#| label: tbl-name`
- Figures: `#| label: fig-name`
- Sections: `{#sec-name}`

- Equations: Reference by tag number

## 7 Advanced Formatting

### 7.1 Callout Blocks

Create attention-grabbing callouts:

```
::: {.callout-note}
This is a note about something important.
:::

::: {.callout-tip}
Pro Tip
This is a helpful tip for readers.
:::

::: {.callout-warning}
Be careful with this approach!
:::

::: {.callout-important}
This is critical information.
:::

::: {.callout-caution}
Proceed with caution.
:::
```

### 7.2 Column Layouts

Create side-by-side content:

```
::: {.columns}
::: {.column width="50%"}
Advantages:
- Clear interpretation
- Easy to implement
- Widely understood
```

```

:::

::: {.column width="50%"}
Limitations:
- Assumes linearity
- Sensitive to outliers
- May miss interactions
:::
:::

```

## 7.3 Tabbed Content

Organize content in tabs:

```

::: {.panel-tabset}

Data Exploration

Your exploratory analysis here...

Model Results

Your regression results here...

Diagnostics

Your diagnostic plots here...

:::

```

## 7.4 Custom Styling

Apply custom CSS styling:

```

::: {style="background-color: #f0f0f0; padding: 15px; border-radius: 5px;}
This content has a gray background with padding.
:::

This is [highlighted text]{style="background-color: yellow;"}.
```

## 8 Complete Example: Regression Analysis

Here's a complete example putting it all together:

```

title: "Regression with Interaction Variables"
author: "Zahid Asghar"
date: today
format:
 html:
 toc: true
 number-sections: true
 code-fold: true
 theme: cosmo

Introduction

In this analysis, I examine salary differences between faculty members
by gender and discipline. The data contains information about
`r nrow(Salaries)` faculty members.

::: {.callout-note}
Objective
Explain interaction variables in regression using only binary
independent variables.
:::

Data Description

```{r}
#| label: tbl-desc
#| tbl-cap: "Summary Statistics by Gender"

Salaries %>%
  group_by(sex) %>%
  summarise(avg_salary = mean(salary)) %>%
  gt() %>%
  fmt_currency(columns = avg_salary, decimals = 1)
```

@tbl-desc shows that male faculty earn more on average.
```

## # Regression Analysis

We estimate the following model:

$$\text{salary}_i = \beta_0 + \beta_1 \text{sex}_i + \epsilon_i \tag{1}$$

where  $i = 1, 2, \dots, n$  and  $\epsilon_i$  is the error term.

```
```{r}
#| label: tbl-reg1
#| tbl-cap: "Basic Regression Results"

model1 <- lm(salary ~ sex, data = Salaries)
modelsummary(model1, stars = TRUE)
```
```

The coefficient  $\beta_1 = 14,088$  in @tbl-reg1 indicates male faculty earn \$14,088 more than female faculty on average.

## ## Adding Discipline

```
```{r}
#| label: tbl-reg2
#| tbl-cap: "Regression with Discipline"

model2 <- lm(salary ~ sex + discipline, data = Salaries)
modelsummary(model2, stars = TRUE)
```
```

## # Interaction Effects

```
```{r}
#| label: tbl-reg3
#| tbl-cap: "Regression with Interaction Term"

model3 <- lm(salary ~ sex * discipline, data = Salaries)
modelsummary(model3, stars = TRUE)
```
```

@tbl-reg3 shows the interaction effect. The coefficient on the

```

interaction term is $-14,109$, indicating the gender gap is
smaller in Discipline B.

Conclusion

::: {.callout-important}
Interaction terms are essential when relationships vary across
subgroups. @tbl-reg3 demonstrates how the effect of gender on
salary differs by discipline.
:::

```

## 9 Tips for Academic Writing

### 9.1 Do's

1. **Use descriptive labels:** `tbl-salary-gender` not `tbl-1`
2. **Add clear captions:** Explain what the table/figure shows
3. **Reference everything:** Use `@tbl-` and `@fig-` consistently
4. **Hide setup code:** Use `#| echo: false` for data loading
5. **Format numbers:** Use `fmt_currency()`, `fmt_number()` etc.
6. **Add section numbers:** Set `number-sections: true`
7. **Include TOC:** Help readers navigate long documents

### 9.2 Don'ts

1. Don't use vague labels like `tbl-1`, `fig-plot`
2. Don't show package loading messages
3. Don't forget to caption tables and figures
4. Don't use raw R output - format with `gt` or `modelsummary`
5. Don't hardcode numbers - use inline code
6. Don't mix different table styles in one document

## 10 Workflow Tips

### 10.1 Iterative Rendering

Render frequently to catch errors early:

- **Keyboard shortcut:** `Ctrl/Cmd + Shift + K`

- Command line: `quarto render document.qmd`
- Render to specific format: `quarto render document.qmd --to pdf`

## 10.2 Code Organization

Structure your code logically:

```
Setup chunk - load packages and data
Exploratory analysis - tables and visualizations
Model fitting - regression models
Results presentation - formatted tables
Diagnostics - check model assumptions
```

## 10.3 Reproducibility Checklist

Load all required packages at the top Set random seed if using random processes Use relative file paths, not absolute Comment your code clearly Include session info at the end Test rendering all output formats

# 11 Output Formats

## 11.1 HTML (Interactive)

Best for sharing online:

```
format:
 html:
 toc: true
 code-fold: true
 code-tools: true
 embed-resources: true
 theme: cosmo
```

## 11.2 PDF (Print)

Best for submission:

```
format:
 pdf:
 toc: true
 number-sections: true
 geometry: margin=1in
 colorlinks: true
```

### 11.3 Word (Collaboration)

Best for tracked changes:

```
format:
 docx:
 toc: true
 number-sections: true
 reference-doc: custom-template.docx
```

## 12 Common Issues and Solutions

### 12.1 Tables Not Appearing

**Problem:** Table code runs but doesn't show in output

**Solution:** Make sure you're using `gt()` or other table function, not just printing

```
Wrong - just prints
Salaries %>% group_by(sex) %>% summarise(mean(salary))

Right - creates formatted table
Salaries %>% group_by(sex) %>% summarise(mean(salary)) %>% gt()
```

### 12.2 Cross-References Not Working

**Problem:** `@tbl-name` shows as plain text

**Solution:** Ensure label starts with correct prefix and caption is included

```
#| label: tbl-summary # Correct prefix
#| tbl-cap: "My table" # Caption required
```

## 12.3 Math Not Rendering

**Problem:** LaTeX equations show as plain text

**Solution:** Check for proper syntax and escaping

```
Wrong
$$
salary = \beta_0 + \beta_1 * sex
$$

Right
$$
salary = \beta_0 + \beta_1 \times sex
$$
```

## 12.4 Code Chunks Producing Errors

**Problem:** Document won't render due to code errors

**Solution:** Test code chunks individually

```
#| error: true # Show error but continue rendering
#| eval: false # Don't run problematic code
```

## 13 Practice Exercise

Create a complete document with:

1. YAML header with HTML and PDF output
2. Numbered sections with table of contents
3. Introduction with objectives
4. Data description with a summary table
5. At least one regression table
6. One mathematical equation (numbered)
7. Cross-references to tables
8. One callout block
9. Conclusion section

**Bonus challenges:**

- Add a scatter plot with caption

- Create a tabbed section with multiple models
- Use inline R code to report key statistics
- Format your regression table with `gt`

## 14 Additional Resources

### 14.1 Documentation

- [Quarto Guide](#)
- [Quarto Gallery](#)
- [GT Package](#)
- [Modelsummary Package](#)

### 14.2 Cheat Sheets

- [Quarto Reference](#)
- [Markdown Quick Reference](#)

### 14.3 Community

- [Posit Community Forum](#)
- [Stack Overflow - Quarto Tag](#)
- [GitHub Discussions](#)

## 15 Quick Reference

### 15.1 Essential YAML Options

```

title: "Title"
author: "Author"
date: today
format:
 html:
 toc: true # Table of contents
 toc-depth: 3 # Depth of TOC
 number-sections: true # Number sections
 code-fold: true # Collapsible code
```

```
code-tools: true # Code viewing tools
theme: cosmo # Visual theme
embed-resources: true # Standalone file

```

## 15.2 Code Chunk Options

```
#| label: chunk-name # Unique identifier
#| echo: true # Show code
#| eval: true # Run code
#| warning: false # Hide warnings
#| message: false # Hide messages
#| fig-cap: "Caption" # Figure caption
#| tbl-cap: "Caption" # Table caption
```

## 15.3 Cross-References

```
@tbl-name # Reference table
@fig-name # Reference figure
@sec-name # Reference section
@eq-name # Reference equation
```

## 15.4 Math Symbols

```
$_alpha, _beta, _gamma$ # Greek letters
$_x_i$ # Subscript
$_x^2$ # Superscript
$_frac{a}{b}$ # Fraction
$_sum_{i=1}^n$ # Sum
$_bar{x}$ # Bar
$_hat{y}$ # Hat
```

### 💡 Final Tip

The best way to learn Quarto is by doing. Start with simple documents and gradually add complexity. Don't be afraid to experiment!

## 16 Appendix: Complete Working Example

Here's the complete code for the salary analysis document:

```

title: "Faculty Salary Analysis"
subtitle: "Gender and Discipline Effects"
author: "Your Name"
date: today
format:
 html:
 toc: true
 number-sections: true
 code-fold: true
 theme: cosmo

```{r}
#| label: setup
#| message: false

library(tidyverse)
library(gt)
library(modelsummary)
library(carData)
data(Salaries)
```

Introduction

This analysis examines salary differences among `r nrow(Salaries)`
faculty members by gender and discipline.

Descriptive Statistics

```{r}
```

```

#| label: tbl-summary
#| tbl-cap: "Average Salary by Gender"

Salaries %>%
  group_by(sex) %>%
  summarise(avg_salary = mean(salary)) %>%
  gt() %>%
  fmt_currency(columns = avg_salary)
```

Regression Analysis

Our model is:

$$\text{salary}_i = \beta_0 + \beta_1 \text{sex}_i + \beta_2 \text{discipline}_i + \beta_3 (\text{sex}_i \times \text{discipline}_i) + \epsilon_i$$

$$\{r\}$$

#| label: tbl-models
#| tbl-cap: "Regression Results"

m1 <- lm(salary ~ sex, data = Salaries)
m2 <- lm(salary ~ sex + discipline, data = Salaries)
m3 <- lm(salary ~ sex * discipline, data = Salaries)

modelsummary(
 list("(1)" = m1, "(2)" = m2, "(3)" = m3),
 stars = TRUE
)
```

# Conclusion

@tbl-models reveals significant gender differences in faculty salaries, with important interactions by discipline.

```

This completes your tutorial on Quarto document writing!