In this post we generate a PowerPoint report from a Shiny app.

**PowerPoint Report**

These are the slides from the PowerPoint report in our Shiny app:

**Creating a Basic PowerPoint with R**

We use the officer R package to create the PowerPoint. officer connects R to Microsoft Word and PowerPoint (similar to how openxlsx connects R to Excel). We use the flextable package to format our PowerPoint tables.

To begin creating the PowerPoint, we assign a blank PowerPoint to the ppt\_report variable:

library(officer)

ppt\_report <- read\_pptx()

We can now create a PowerPoint file from our ppt\_report object by using the generic print() function and setting target = "" as below:

print(ppt\_report, target = "my\_ppt\_file.pptx")

# run `?print.rpptx` for more info on the above function

The PowerPoint we created above is completely empty, so it’s not terribly interesting. Let’s add a basic title page:

library(dplyr)

ppt\_report <- ppt\_report %>%

add\_slide(

layout = "Title Slide",

master = "Office Theme"

) %>%

ph\_with\_text(

type = "ctrTitle",

str = c(

"Example Client Name",

"Workers' Compensation Claims Report"

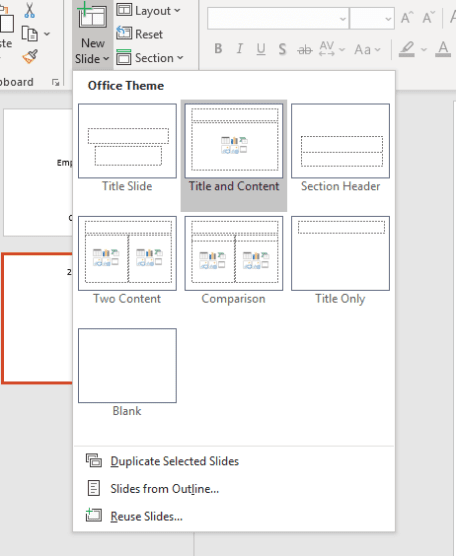
)

)

print(ppt\_report, target = "my\_ppt\_file.pptx")

The new version of our PowerPoint now has a title page.

You can specify the slide theme with the master argument to add\_slide() (“Office Theme” in our above example). All the layouts available with PowerPoint can be set with the layout argument. e.g. you can make slides with all the following layouts:



**Using Flextables**

officer can add tables, but it does not have many table formatting options. For more customized tables we use the flextable package.

Check out the code below to see the full formatting of the PowerPoint in our Shiny app:

|  |
| --- |
| output$generate\_ppt\_report <- downloadHandler( |
|  |  |
|  | filename = function() { |
|  | paste0("claims-report-as-of-", input$val\_date, ".pptx") |
|  | }, |
|  |  |
|  | content = function(file) { |
|  | removeModal() |
|  |  |
|  | eval\_ <- input$val\_date |
|  |  |
|  | lr\_current <- val\_tbl() |
|  | lr\_prior <- loss\_run(eval\_ - years(1)) |
|  |  |
|  | table1 <- lr\_current %>% |
|  | mutate( |
|  | year = as.character(lubridate::year(accident\_date)), |
|  | n\_open = ifelse(status == "Open", 1, 0) |
|  | ) %>% |
|  | group\_by(year) %>% |
|  | summarize( |
|  | paid = sum(paid), |
|  | case = sum(case), |
|  | reported = sum(reported), |
|  | n\_open = sum(n\_open), |
|  | n = n() |
|  | ) %>% |
|  | ungroup() %>% |
|  | summaryrow::blank\_row() %>% |
|  | summaryrow::totals\_row( |
|  | cols = 2:6, |
|  | label\_col = 1 |
|  | ) |
|  |  |
|  | footers = as.character(as.vector(table1[10,])) |
|  | for (i in 2:6) { |
|  | footers[[i]] <- format(as.numeric(footers[[i]]), big.mark = ",") |
|  | } |
|  |  |
|  | table1 <- table1[1:8, ] |
|  |  |
|  | names(table1) <- c("Accident Year", "Paid", "Case", "Reported1", "Open", "Reported2") |
|  |  |
|  | table1 <- flextable(table1) %>% |
|  | set\_header\_labels(Reported1 = "Reported", Reported2 = "Reported") %>% |
|  | add\_header\_row( |
|  | top = TRUE, |
|  | values = c( |
|  | "", |
|  | rep("Loss & ALAE", 3), |
|  | rep("Number of Claims", 2) |
|  | ) |
|  | ) %>% |
|  | merge\_h(part = "header") %>% |
|  | add\_footer\_row(values = footers, colwidths = rep(1, 6)) %>% |
|  | width(width = 1.25) %>% |
|  | height\_all(height = 0.5) %>% |
|  | colformat\_num(col\_keys = c("Paid", "Case", "Reported1", "Open"), |
|  | big.mark = ",", digits = 0) %>% |
|  | theme\_booktabs() %>% |
|  | align(align = "center", part = "header") |
|  |  |
|  |  |
|  | #table 2 |
|  |  |
|  | out <- lr\_current %>% |
|  | select(claim\_num, accident\_date, claimant, state, status, paid, reported) |
|  |  |
|  | lr\_prior\_out <- lr\_prior %>% |
|  | select(claim\_num, paid, reported) |
|  |  |
|  | table2 <- out %>% |
|  | left\_join(lr\_prior\_out, by = "claim\_num") %>% |
|  | mutate( |
|  | paid\_change = paid.x - paid.y, |
|  | reported\_change = reported.x - reported.y |
|  | ) %>% |
|  | filter(paid\_change >= 100000) %>% |
|  | arrange(desc(paid\_change)) %>% |
|  | summaryrow::blank\_row() %>% |
|  | summaryrow::totals\_row( |
|  | cols = 10:11, |
|  | label\_col = 1 |
|  | ) |
|  |  |
|  | table2 <- table2[1:4,] |
|  |  |
|  | names(table2) <- c( |
|  | "Claim Number", |
|  | "Accident Date", |
|  | "Claimant", |
|  | "State", |
|  | "Status", |
|  | "Paid1", |
|  | "Reported1", |
|  | "Paid2", |
|  | "Reported2", |
|  | "Paid3", |
|  | "Reported3" |
|  | ) |
|  |  |
|  | table2 <- flextable(table2) %>% |
|  | set\_header\_labels(Paid1 = "Paid", Paid2 = "Paid", Paid3 = "Paid", |
|  | Reported1 = "Reported", Reported2 = "Reported", Reported3 = "Reported") %>% |
|  | add\_header\_row( |
|  | top = TRUE, |
|  | values = c( |
|  | rep("", 5), |
|  | rep(paste0("As of ", format(input$val\_date, "%B %d, %Y")), 2), |
|  | rep(paste0("As of ", format(input$val\_date - years(1), "%B %d, %Y")), 2), |
|  | rep("Change", 2) |
|  | ) |
|  | ) %>% |
|  | merge\_h(part = "header") %>% |
|  | width(width = 0.8) %>% |
|  | height\_all(height = 0.75) %>% |
|  | colformat\_num(col\_keys = c("Paid1", "Reported1", "Paid2", |
|  | "Reported2", "Paid3", "Reported3"), |
|  | big.mark = ",", digits = 0) %>% |
|  | theme\_booktabs() %>% |
|  | align(align = "center", part = "header") |
|  |  |
|  | ppt\_report <- read\_pptx() %>% |
|  | add\_slide(layout = "Title Slide", master = "Office Theme") %>% |
|  | ph\_with\_text( |
|  | type = "ctrTitle", |
|  | str = c( |
|  | "Example Client Name", |
|  | "Workers' Compensation Claims Report") |
|  | ) %>% |
|  | ph\_with\_text( |
|  | type = "subTitle", |
|  | str = c( |
|  | paste0("Data Evaluated as of ", format(input$val\_date, "%B %d, %Y")), |
|  | paste0("Report Generated on ", format(Sys.Date(), "%B %d, %Y")) |
|  | ) |
|  | ) %>% |
|  | ph\_with\_img\_at( |
|  | src = "server/04-report-srv/images/tychobra\_logo\_blue\_co\_name.png", |
|  | height = 1.5, |
|  | width = 6, |
|  | left = 2, |
|  | top = 0 |
|  | ) %>% |
|  | add\_slide(layout = "Blank", master = "Office Theme") %>% |
|  | ph\_with\_text( |
|  | type = "ftr", |
|  | str = c( |
|  | "Exhibit 1", |
|  | "Summary of Loss & ALAE", |
|  | paste0("Evaluated as of ", format(input$val\_date, "%B %d, %Y")) |
|  | ) |
|  | ) %>% |
|  | ph\_with\_flextable\_at( |
|  | value = table1, |
|  | left = 1.3, |
|  | top = 0.6 |
|  | ) %>% |
|  | add\_slide(layout = "Title and Content", master = "Office Theme") %>% |
|  | ph\_with\_text( |
|  | type = "ftr", |
|  | str = c( |
|  | "Exhibit 2", |
|  | "Claims with charge in paid >= 100,000", |
|  | paste0("Evaluated as of ", format(input$val\_date, "%B %d, %Y")) |
|  | ) |
|  | ) %>% |
|  | ph\_with\_flextable\_at( |
|  | value = table2, |
|  | left = 0.55, |
|  | top = 1 |
|  | ) |
|  |  |
|  |  |
|  | print(ppt\_report, target = file) |
|  | } |
|  | ) |

**Demo App**

Below is a complete but very simple Shiny app that can generate a PowerPoint report. Simply run the following directly in your R console to run the app yourself!

library(shiny)

library(officer)

library(flextable)

library(dplyr)

my\_table <- data.frame(

Name = letters[1:4],

Age = seq(20, 26, 2),

Occupation = LETTERS[15:18],

Income = c(50000, 20000, 30000, 45000)

)

ui <- fluidRow(

column(

width = 12,

align = "center",

tableOutput("data"),

br(),

downloadButton("download\_powerpoint", "Download Data to PowerPoint")

)

)

server <- function(input, output) {

output$data <- renderTable({

my\_table

})

output$download\_powerpoint <- downloadHandler(

filename = function() {

"employee\_data.pptx"

},

content = function(file) {

flextable\_prep <- flextable(my\_table) %>%

colformat\_num(col\_keys = c("Age", "Income"), digits = 0) %>%

width(width = 1.25) %>%

height\_all(height = 0.35) %>%

theme\_zebra() %>%

align(align = "center", part = "all")

example\_pp <- read\_pptx() %>%

add\_slide(layout = "Title Slide", master = "Office Theme") %>%

ph\_with\_text(

type = "ctrTitle",

str = "Employee Data"

) %>%

ph\_with(

location = ph\_location\_type(type = "subTitle"),

value = "Company 2019 Report"

) %>%

add\_slide(layout = "Title and Content", master = "Office Theme") %>%

ph\_with\_text(

type = "title",

str = "2019 Data"

) %>%

ph\_with\_flextable\_at(

value = flextable\_prep,

left = 2.5,

top = 2

)

print(example\_pp, target = file)

}

)

}

shinyApp(ui, server)

Which will download this basic PowerPoint:

