Workshop Series:
Reusable Research Data Made Shiny

Ontario Dairy Research Centre | Online
February 21st - 24th, 2023

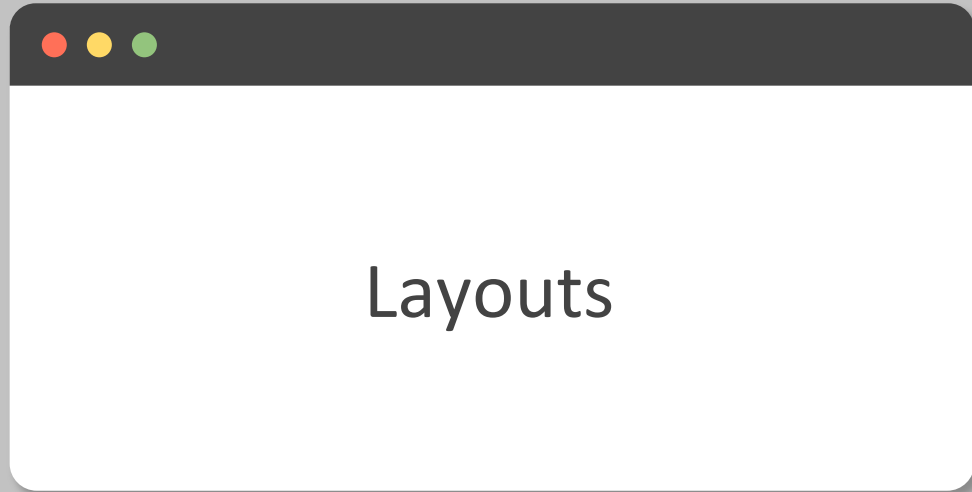UNIVERSITY of GUELPH
IMPROVE LIFE.
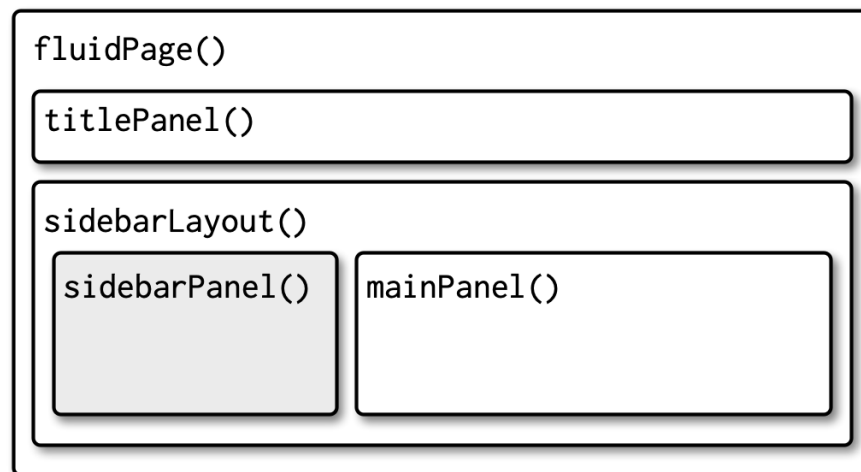
Compute Ontario

ADC

Layouts

Single-page Layouts

Your Shiny App can have one or multiple pages. So far we have only built pages with a single-page layout. Let's take a closer look at the sidebar layout:

```
fluidPage(
    titlePanel(),
    sidebarLayout(
            sidebarPanel(),
            mainPanel()
    )
)
```
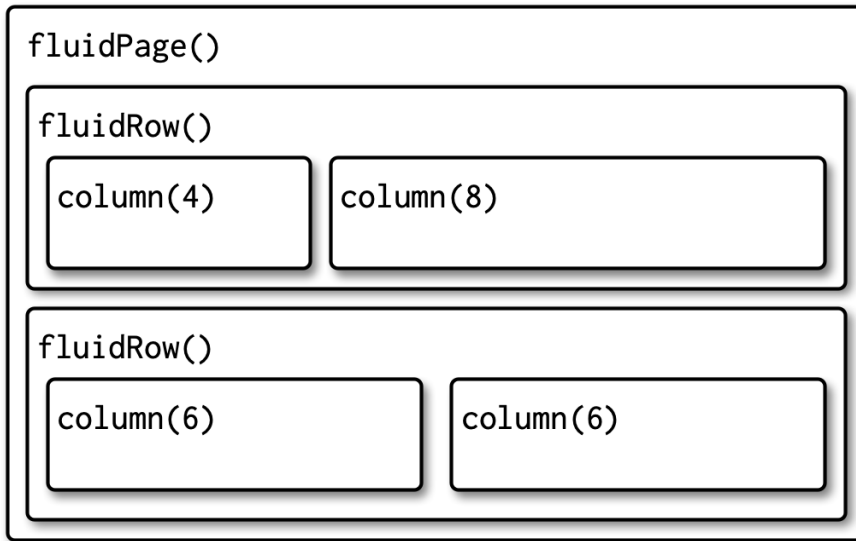
Single-page Layouts

But this is your app - you can build your own layout with the specified number of rows and columns.

```
fluidPage(
    fluidRow(
        column(4,
            …
            ),
        column(8,
            …
            )
        ),
    fluidRow(
        column(6,
            …
            ),
        column(6,
            …
            )
        )
    )
```



```
fluidPage()

    fluidRow()

        column(4)    column(8)


    fluidRow()

        column(6)          column(6)
```

You can have up to 12 columns per row and not all of them need to have content – they can be used as spacers

Your turn!

Modify the **ui** for the sample app in the folder **single-page-app** by recreating the sidebar layout using fluidRow() and column() instead.

Hint: Check the sidebarLayout documentation on the "help" tab to see its default width (number of columns) per panel.

Multi-page Layouts

As your app grows in complexity, it might become impossible to fit everything on a single page. You can use tabPanel() to create the illusion of multiple pages. It's still just one page, but it's broken down into multiple tabs that are only visible one at a time.

Let's take a look at our sample app inside **multi-page-app**

Your turn!

Convert the single-page app we used in the previous session to a multi-page app. You can find it inside the folder: my-previous-app. It should look like this at the end:

Multi-page Layouts

Because tabs are displayed horizontally, there's a fundamental limit to how many tabs you can use, particularly if they have long titles. navlistPanel() provide an alternative layout that let you use more tabs with longer titles.

Multi-page Layouts

Another option is to use navbarPage() and navbarMenu() to create a horizontal navigation bar at the top of the page with additional subpanels.

Page title	panel 1	panel 2	panel 3	subpanels ▾

panel 4a

panel 4b

panel 4c

one

Multi-page Layouts

Another option is to build a dashboard with the **dashboardPage()** function from the shinydashboard package.

Your turn!

Play around with our previous app. Pick one of the layouts we saw today and refactor the code to fit one of those.

Themes

It's no secret that our apps could use some styling! We can use the shinythemes package to apply some themes on our app and make it look nicer!

Add themeSelector() anywhere inside your ui to create a live theme selector to load with your app.

Once you picked one, you can comment this line and add your theme name directly, for instance the lux theme would be:

theme = shinytheme("lux")

Go further: look into the bslib package and how you can control everything on your page, from color, to font, to font size, etc.