

Let's make a shiny dashboard

Aurora & Lene

2019-02-04

What is shiny?

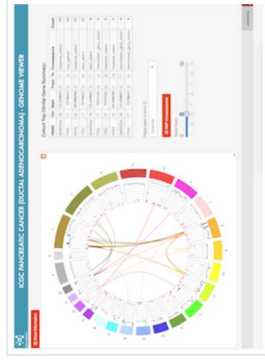
Shiny from  Studio

Get Started [Gallery](#) Articles Reference Deploy Help [Contribute](#) 

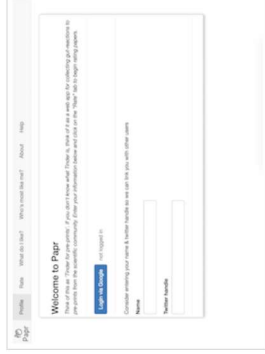
Gallery

Shiny User Showcase

The Shiny User Showcase contains an inspiring set of sophisticated apps developed and contributed by Shiny users.



Genome browser



Papr



Lego Set Database Explorer



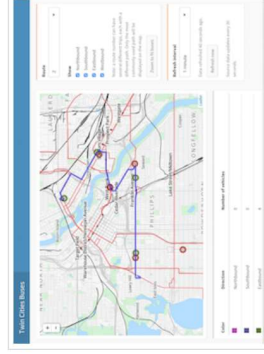
See more

Interactive visualizations

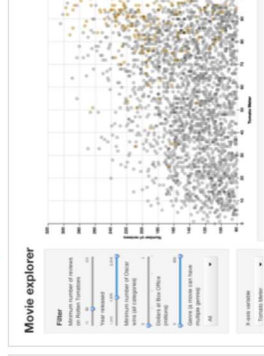
Shiny is designed for fully interactive visualization, using JavaScript libraries like d3, Leaflet, and Google Charts.



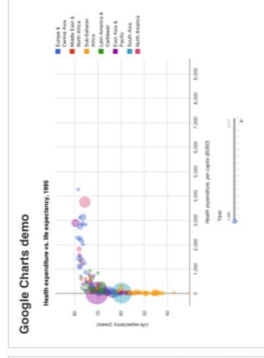
SuperZip example



Bus dashboard



Movie explorer



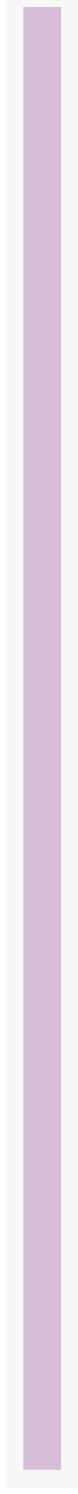
Google Charts

<https://shiny.rstudio.com>

Plan for today:

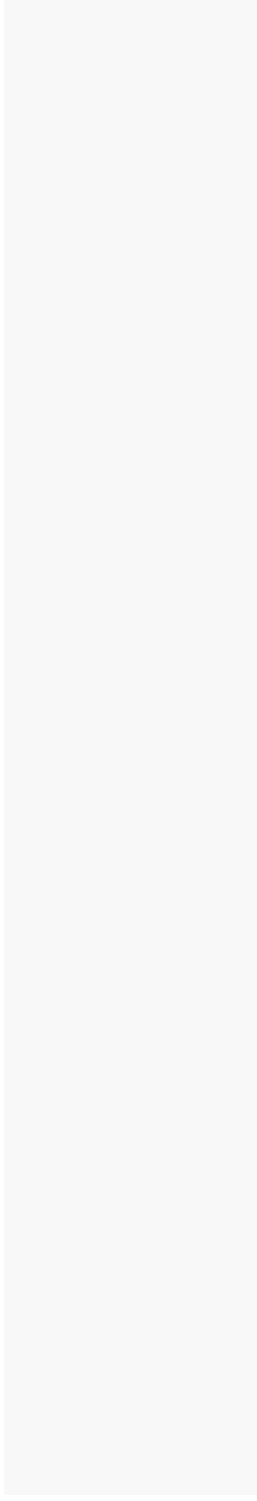
- Step-by-step creation of a shiny dashboard
- A little bit on how shiny works
- Some nice extra features

Code instructions:



Setup

We need these packages:



File structure

We will use the upper file structure:

R > Rladies > workshop_dashboard				
Navn	Status	Endringsdato	Type	
data	✓	04.01.2019 19.59	Filmappe	
www	✓	29.01.2019 13.49	Filmappe	
app	✓	31.01.2019 15.07	R-fil	

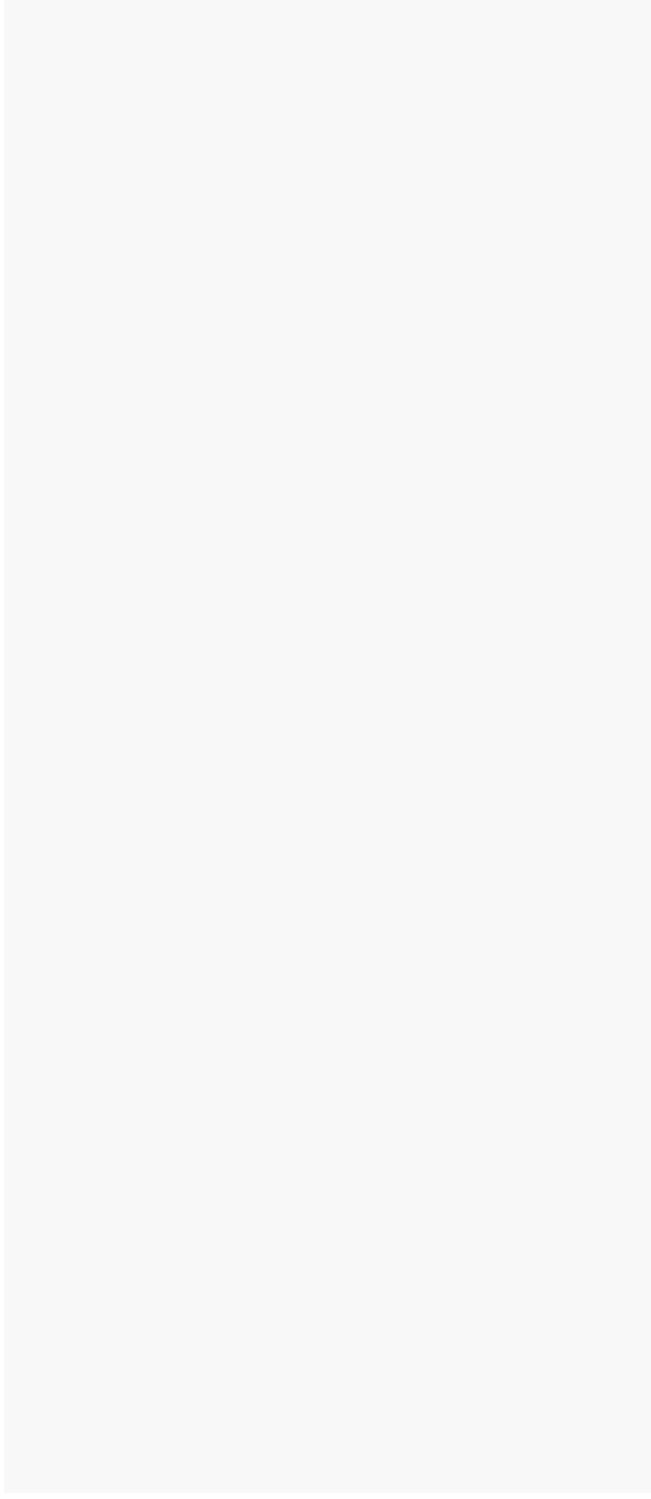
Navn	Endringsdato	Type
data	31.01.2019 19.56	Filmappe
www	31.01.2019 19.55	Filmappe
server	31.01.2019 19.57	R-fil
ui	31.01.2019 19.56	R-fil

Starter code

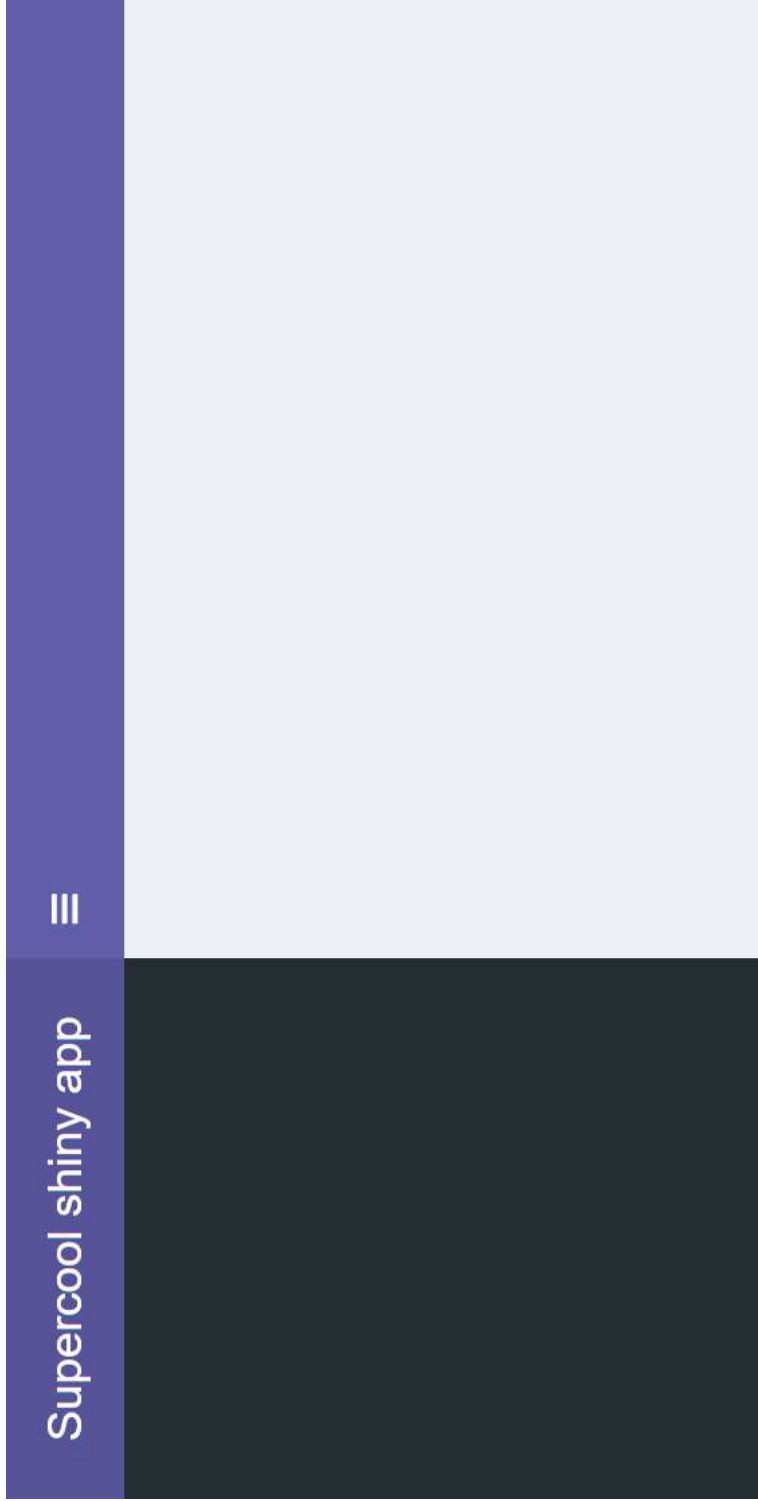
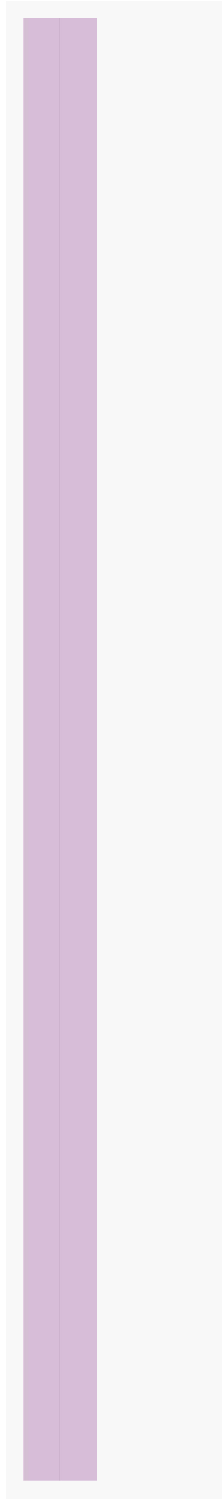
- https://rstudio.github.io/shinydashboard/get_started.html or
- https://github.com/rladies/meetup-presentations_oslo -> shiny-workshop/starter_kit

Starter code

- https://rstudio.github.io/shinydashboard/get_started.html or
- https://github.com/rladies/meetup-presentations_oslo -> shiny-workshop/starter_kit



Title and color



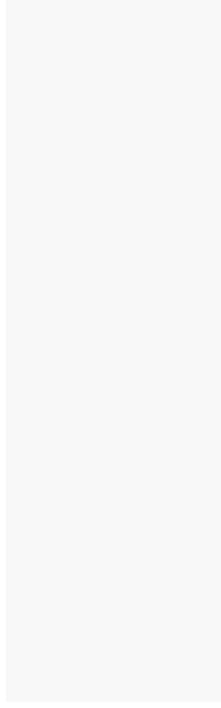
Create tabs and menu



Icons: <https://shiny.rstudio.com/reference/shiny/0.14/icon.html>

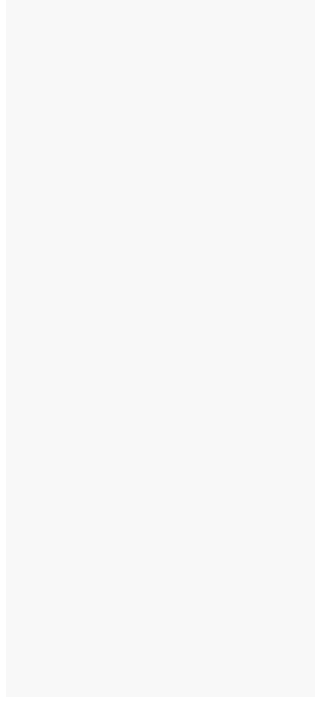
Reactivity = magic?

Connect ui and server



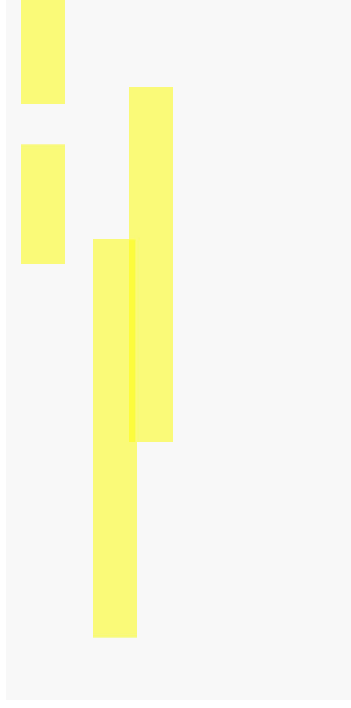
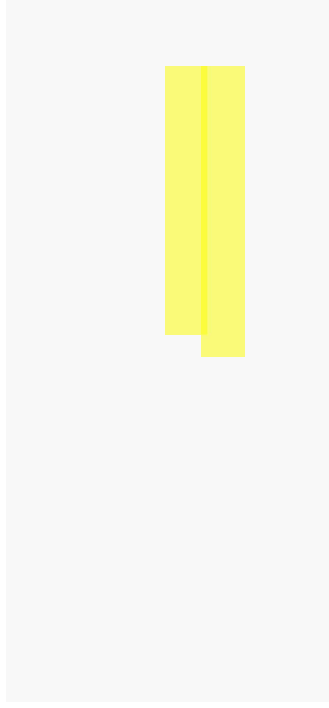
ui: “render” HTML

- Use widgets to create input
- Show output such as text, plots..

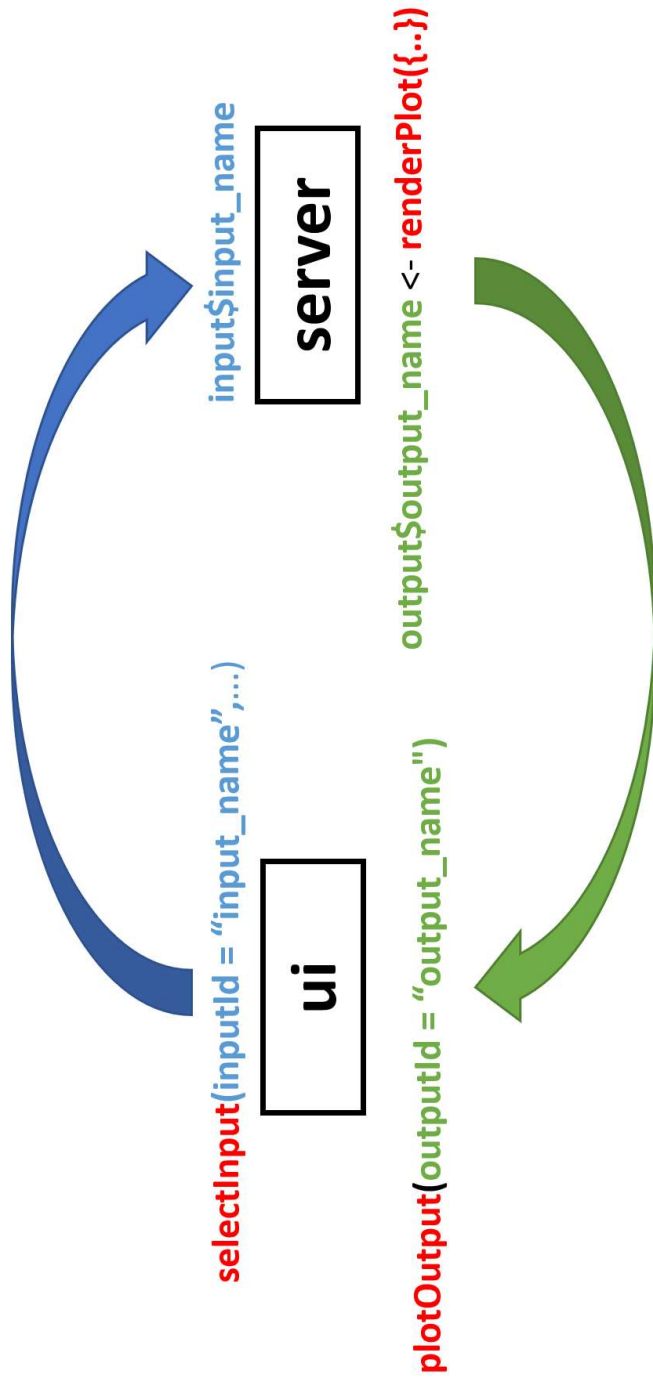


server: run R code

Connect ui and server



Connect ui and server



<https://shiny.rstudio.com/articles/reactivity-overview.html>

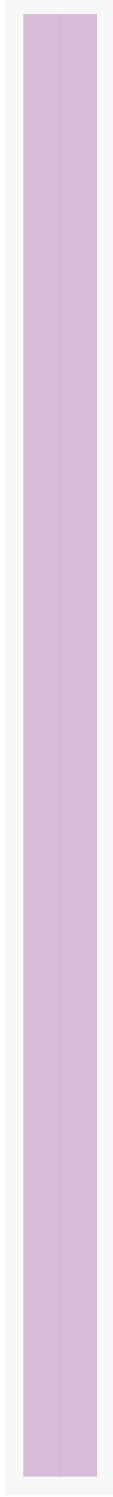
Gapminder data

country	continent	year	lifeExp	pop	gdpPercap
Afghanistan	Asia	1952	28.801	8425333	779.4453
Afghanistan	Asia	1957	30.332	9240934	820.8530
Afghanistan	Asia	1962	31.997	10267083	853.1007
Afghanistan	Asia	1967	34.020	11537966	836.1971
Afghanistan	Asia	1972	36.088	13079460	739.9811
Afghanistan	Asia	1977	38.438	14880372	786.1134

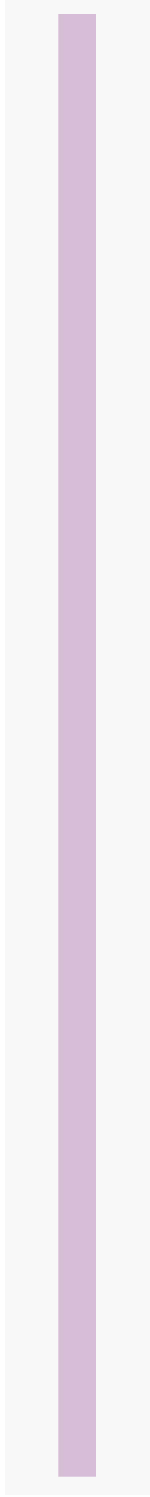
First output: Data table



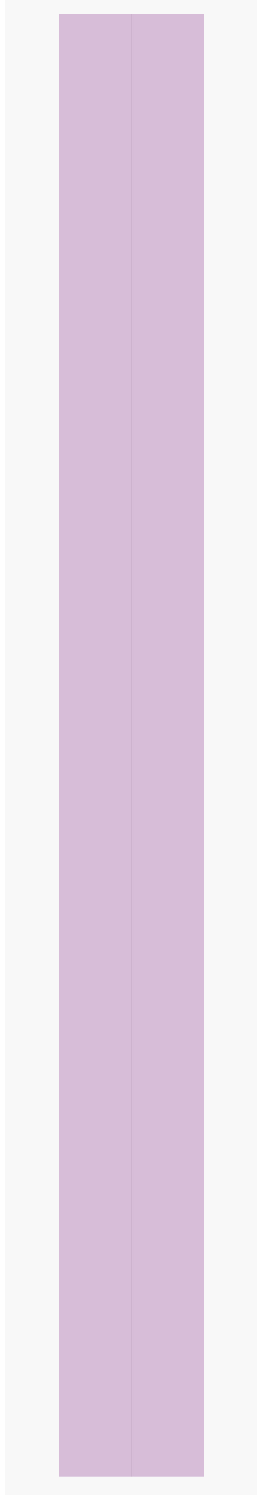
First output: Data table



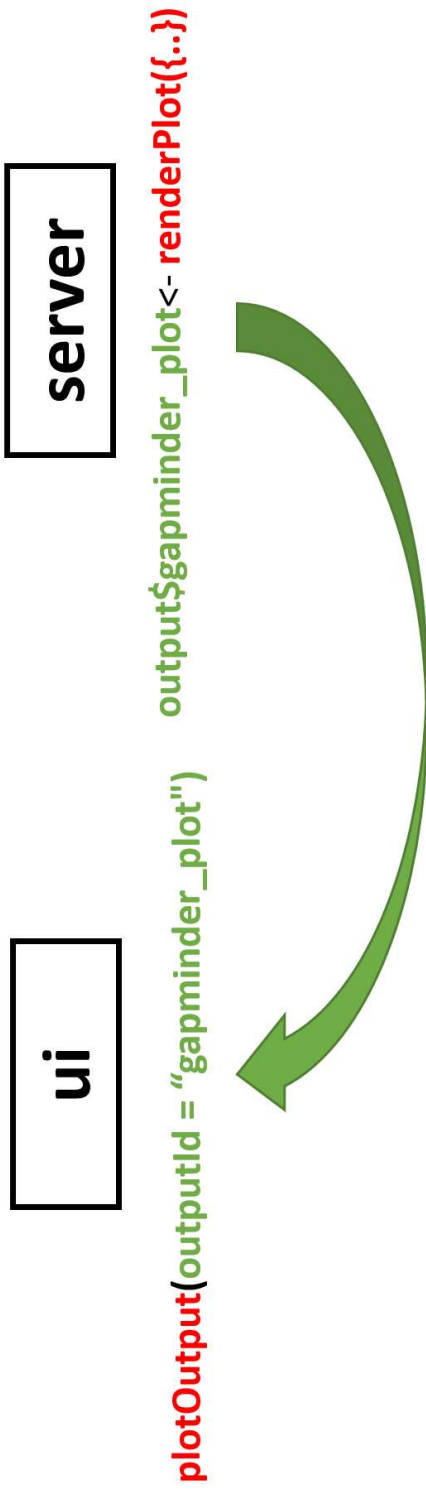
Add the output object under the tab in the ui



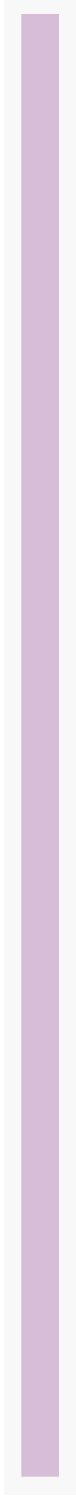
Create the table on the server side



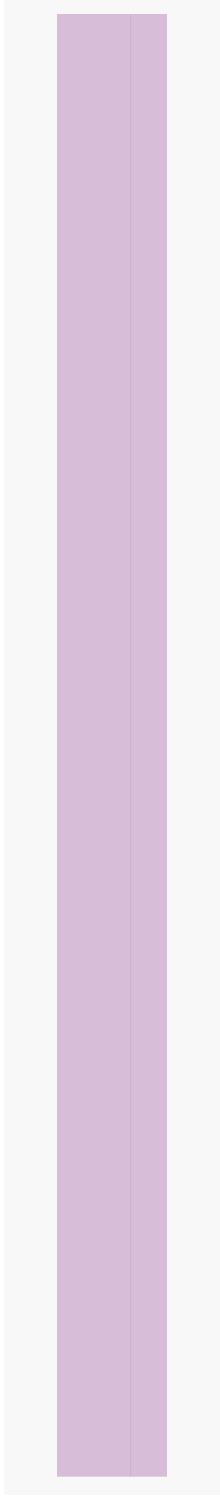
Plots: renderPlot and plotOutput



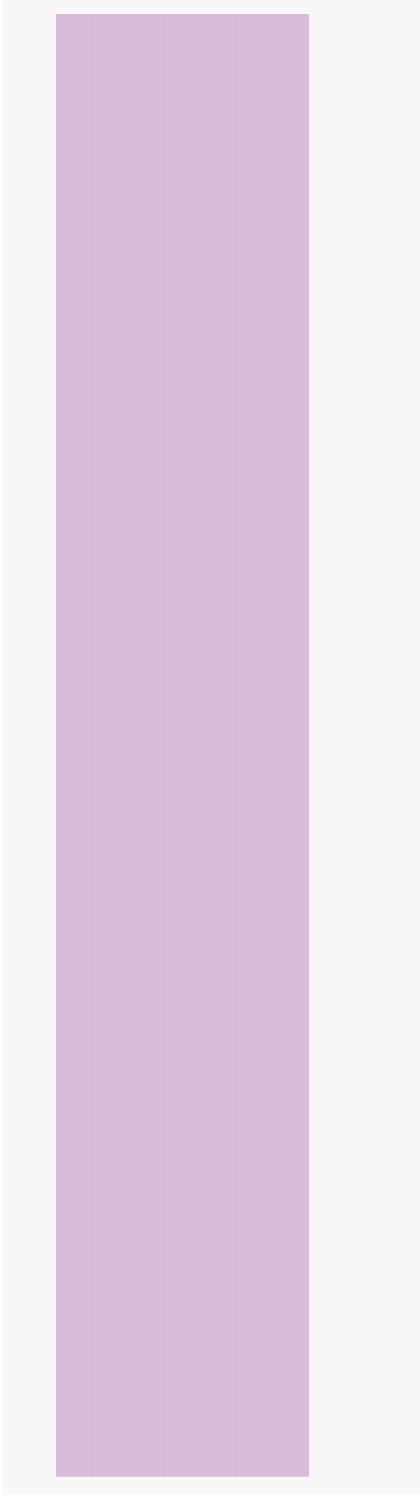
Plots: renderPlot and plotOutput



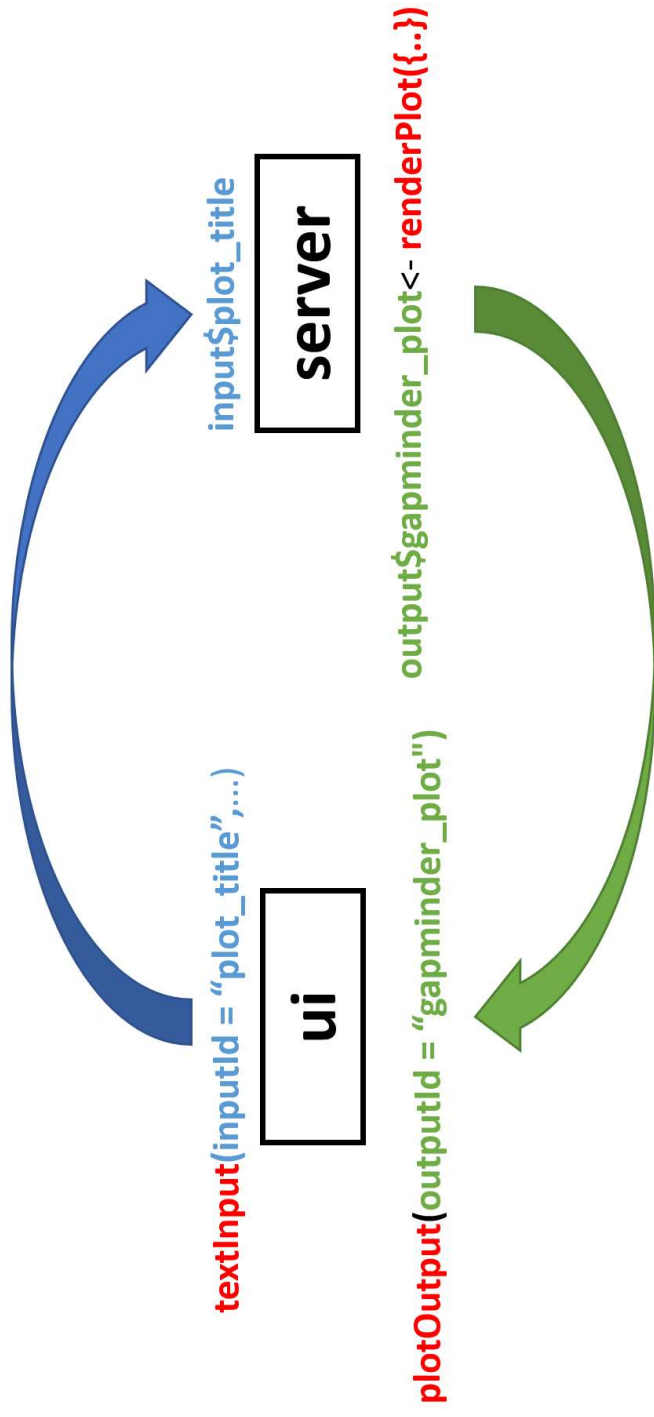
Create a row and a box and a plotOutput in the UI



Generate the plot on the server side



Use widgets to interact with the plot



Input widgets:

Numeric input

1

Current Value:

[1] 1

See Code

Radio buttons

☒ Choice 1
☐ Choice 2
☐ Choice 3

Current Values:

[1] "1"

See Code

Select box

Choice 1 ▾

Current Value:

[1] "1"

See Code

Slider

0 10 20 30 40 50 60 70 80 90 100

50

Current Value:

[1] 50

See Code

Slider range

0 10 20 30 40 50 60 70 80 90 100

25 75

Current Values:

[1] 25 75

See Code

Text input

Enter text...

Current Value:

[1] "Enter text..."

See Code

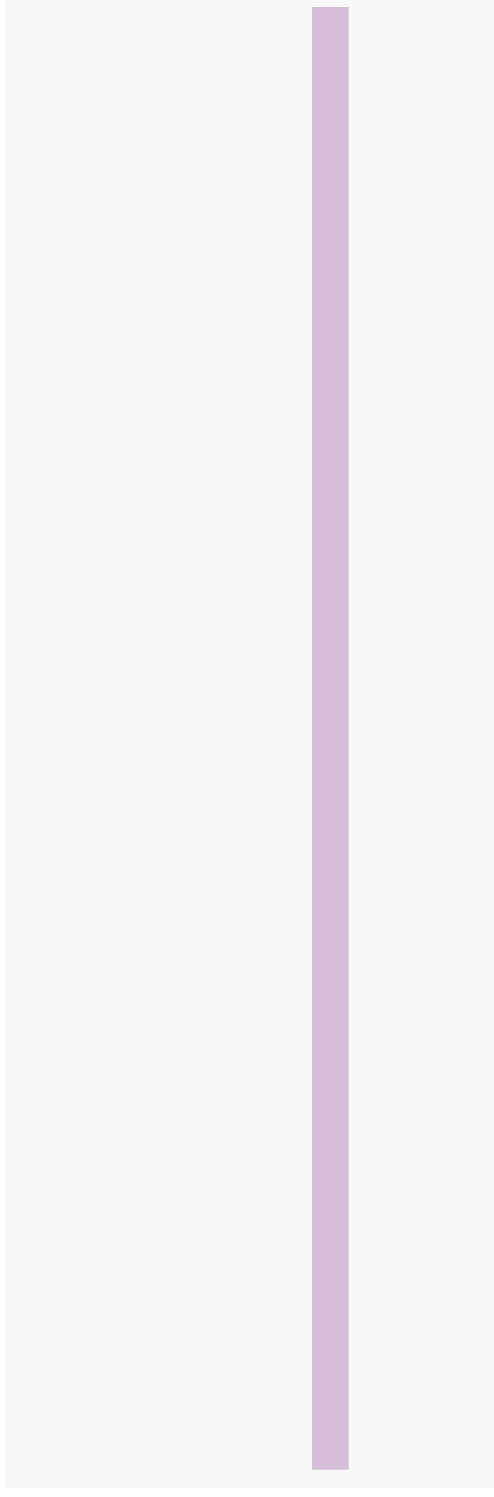
<https://shiny.rstudio.com/gallery/widget-gallery.html>

First widget: TextInput

Add another box for the widgets

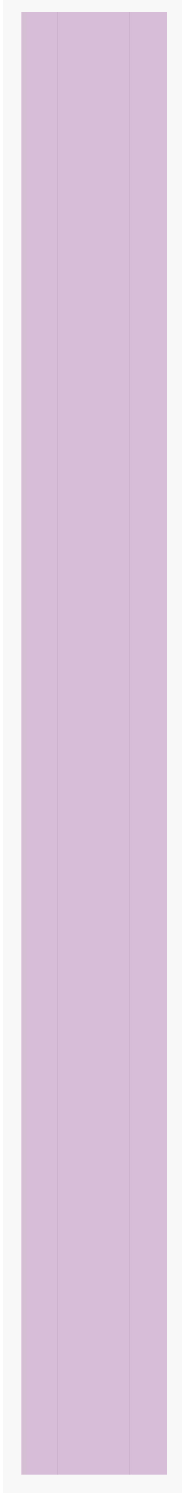
First widget: TextInput

Connect the widget to the plot on the server side

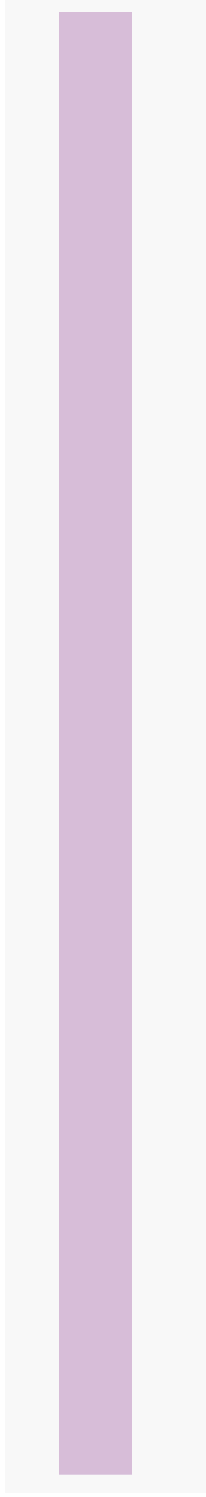


Filter the data using a sliderInput

Add the widget in the same box

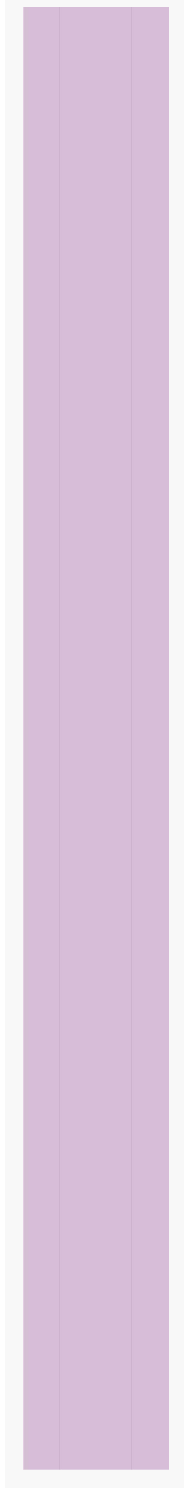


Add a filter inside renderPlot on the server side

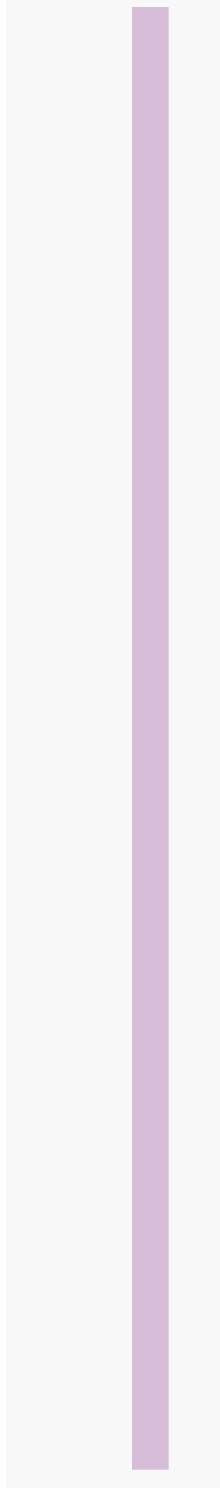


Use selectInput widget to pick continents

Add another widget in the same box



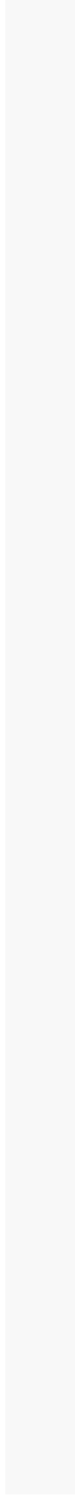
And connect it on the server



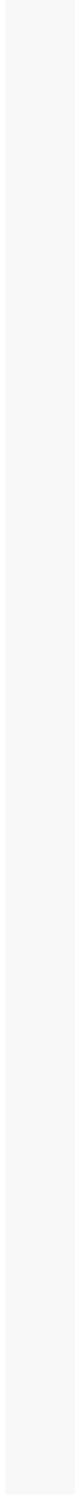
Exercise: create a numeric input widget to set the point size

- `inputId = "point_size"`
- `label = "Point size"`
- `value = 2`

ui inside box:

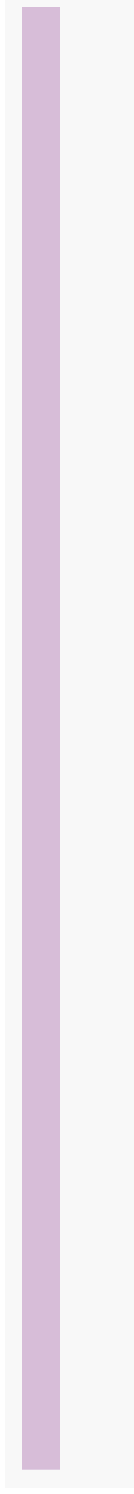


server inside renderPlot:

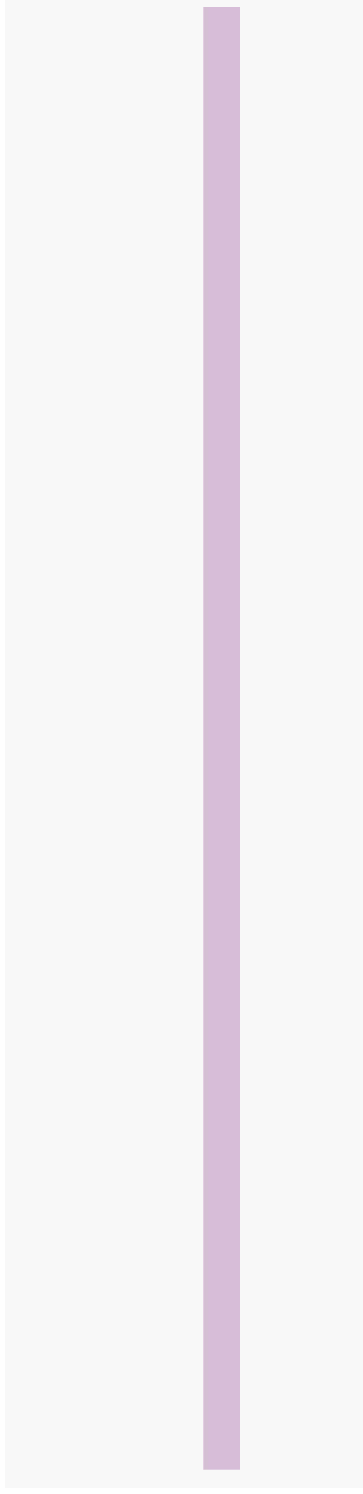


Exercise: create a numeric input widget to set the point size

The widget goes in the ui



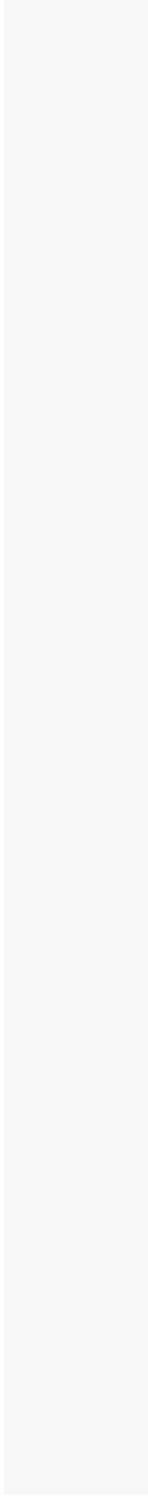
The input `$point_size` goes into `renderPlot`



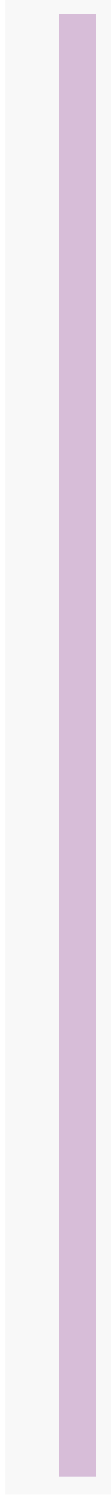
Make it pretty using CSS

Cascading Style Sheets - the "makeup" of your app

Create a new file `custom.css` under `www`:

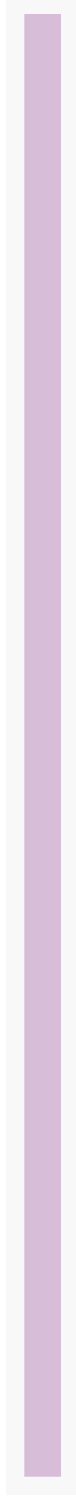


Include it in your app under `dashboardBody`



Advanced css: download the Rladies stylesheet we made for you from

https://github.com/rladies/meetup-presentations_oslo/tree/master/shiny-workshop/workshop_dashboard/www



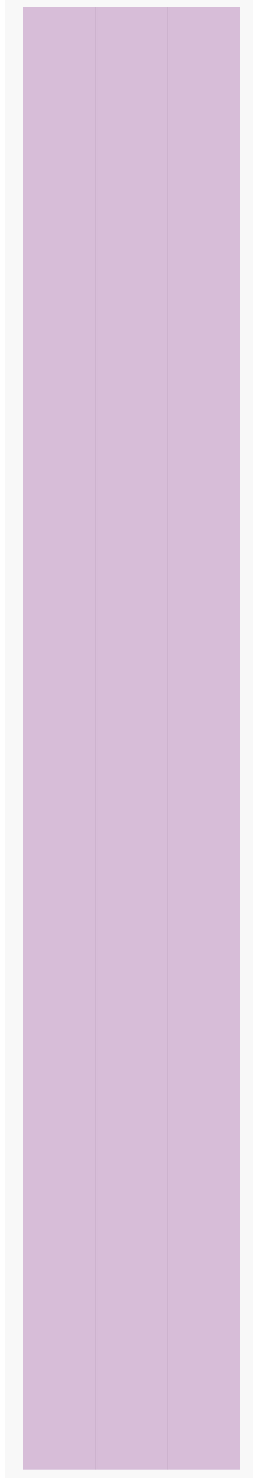
Rladies logo

Download the logo from

https://github.com/rladies/meetup-presentations_oslo/tree/master/shiny-workshop/workshop_dashboard/www

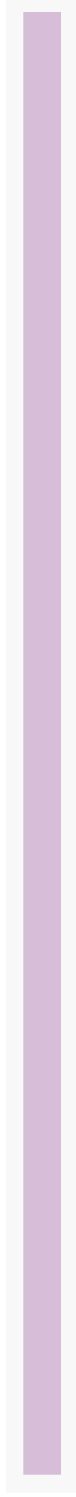
into your app's www folder and add code.

After dashboard title:

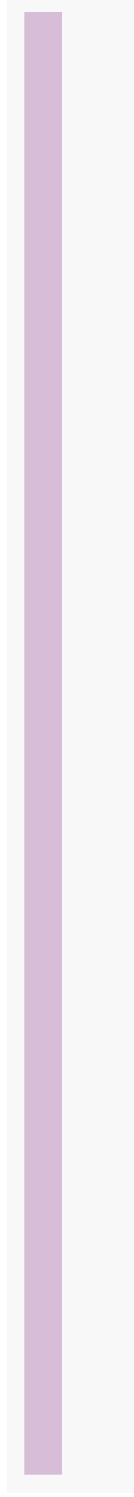


Use plotly for animation

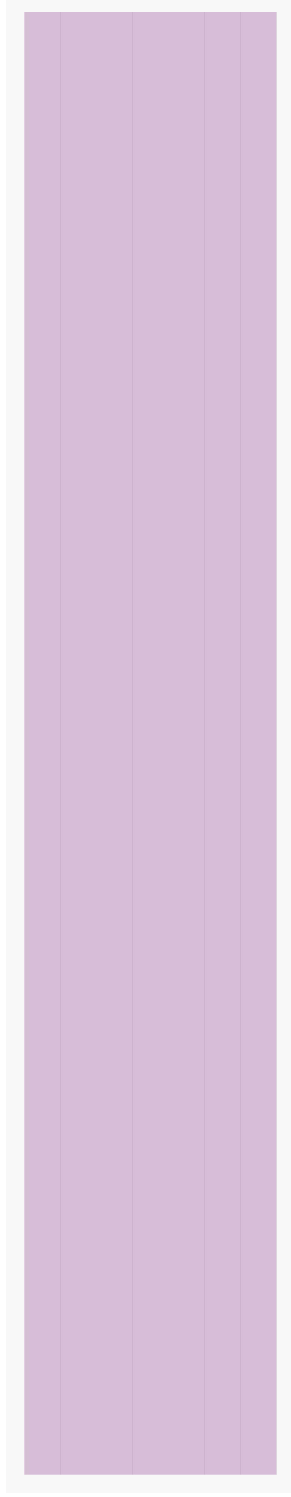
Include plotly



Create a new menu item

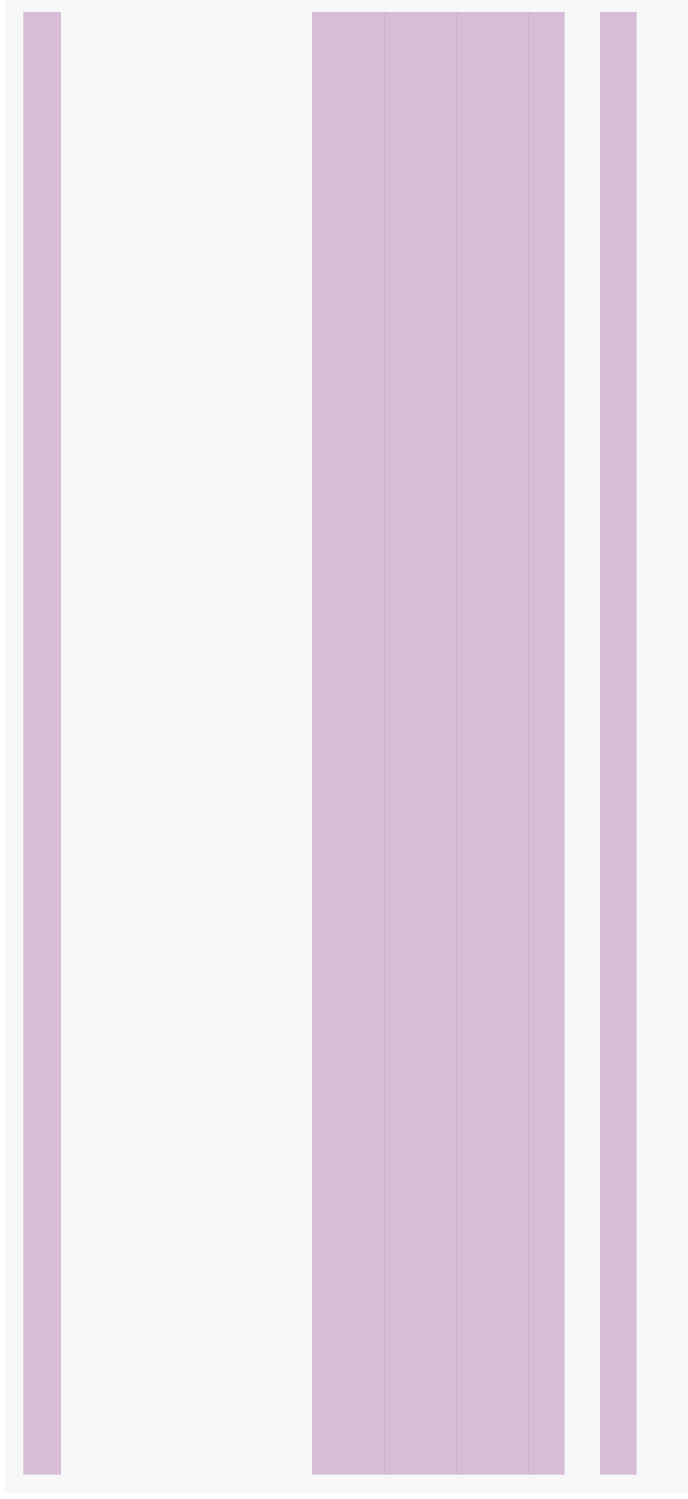


Create a new tab



Use plotly for animation

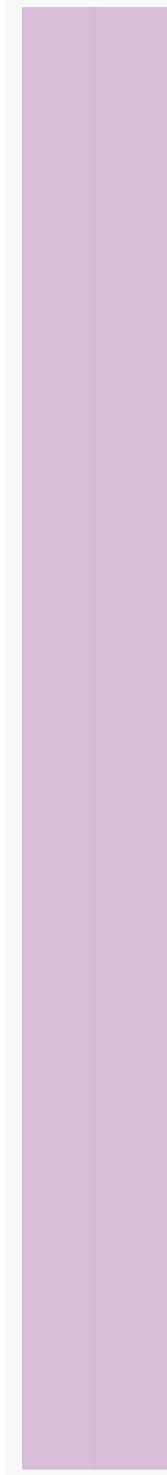
Create the plot:



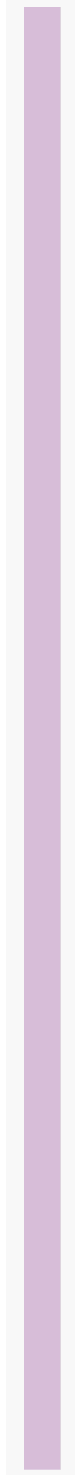
Some cool extras

Actionbuttons can do lot's of things for example change tabs

Add button under the widgets in the plot tab

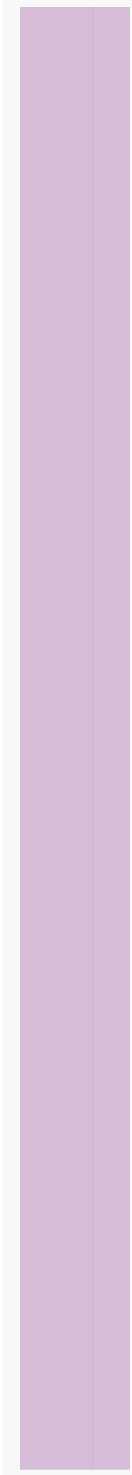


Special: To change tabs we need to name the menu

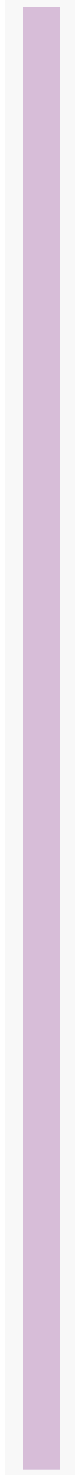


Actionbuttons can do lot's of things for example change tabs

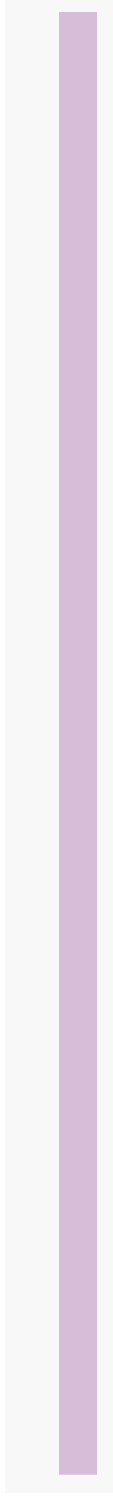
Inside server: use **observeEvent** to listen to the button



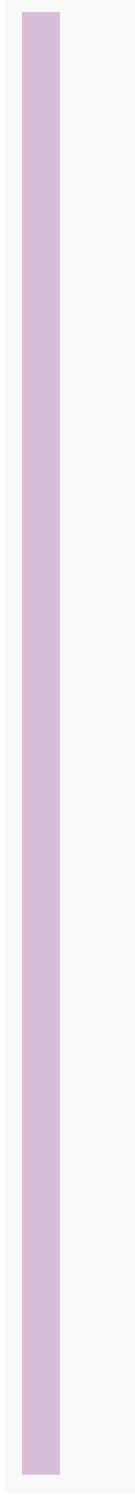
Special: add **session** to the server arguments



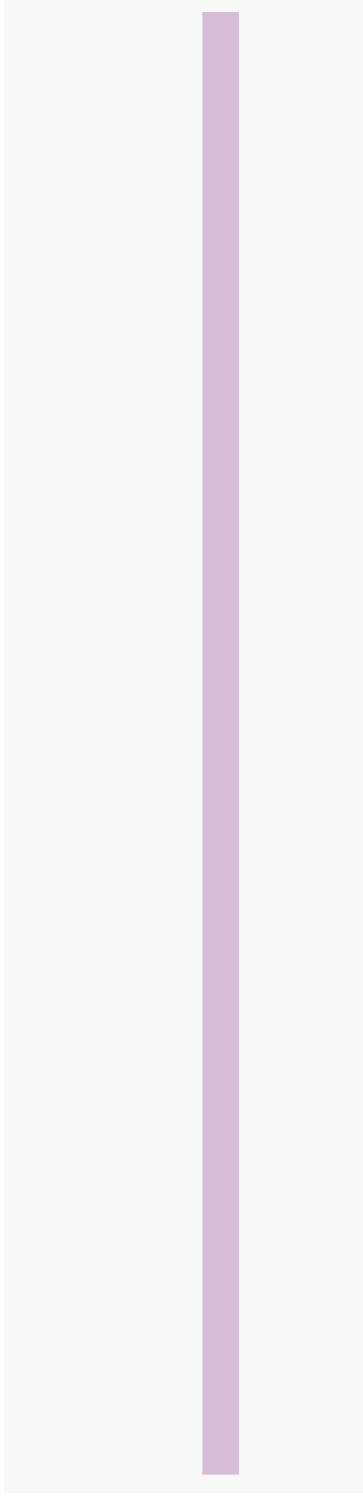
ColourInput



ui inside box:



server inside renderPlot:



What next - Shiny in production

- Control reactivity: next topic to learn
- Responsivity:
 - Done for you
 - App scales size on any gadget (short demo)
- Scaling to many users:
 - Shinyloadtest - testing synthetic load of many users
 - Profvis - what part of code is slow
 - Keynote talk by Joe Cheng @ rstudioconf:2019:
<https://resources.rstudio.com/rstudio-conf-2019/shiny-in-production-principles-practices-and-tools-joe-cheng>
- Deployment:
 - shiny server: <https://www.rstudio.com/products/shiny/shiny-server/>
 - shinyapps.io: <https://shiny.rstudio.com/articles/shinyapps.html>
 - RStudioConnect: <https://www.rstudio.com/products/connect/>

Next meetup: 18 March!

Bayesian methods for rank and preference data - from recommendation systems to cancer genomics - <https://www.meetup.com/rladies-oslo/events/256566088/>

See also Oslo useR! on Wednesday: <https://www.meetup.com/Oslo-useR-Group/events/256805098/>