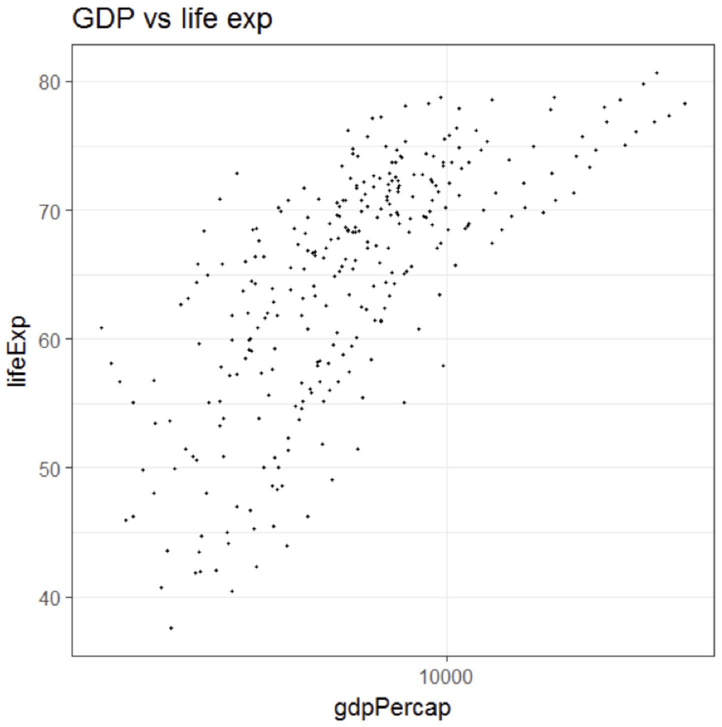
Make the perfect plot using Shiny

CASE STUDIES: BUILDING WEB APPLICATIONS WITH SHINY IN R

Dean Attali
Shiny Consultant

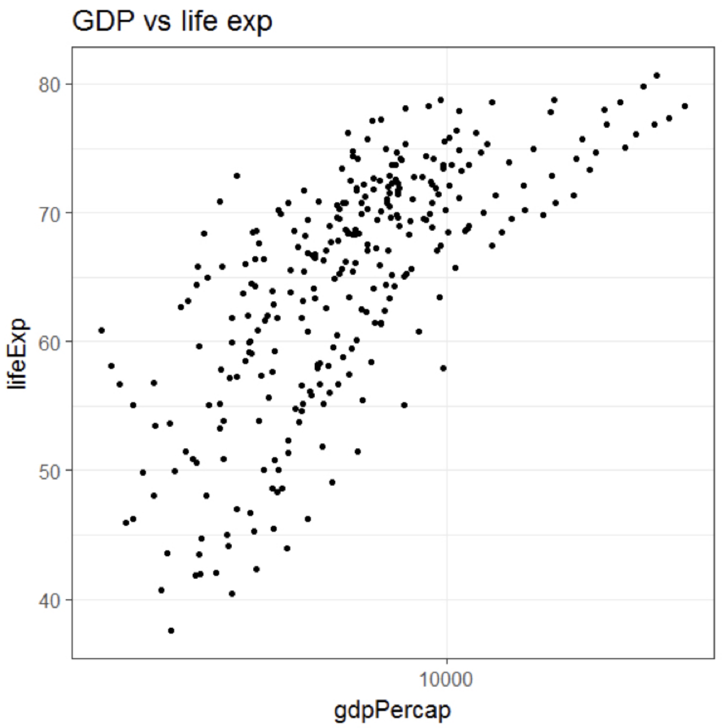
Re-plotting using R code

```r
make_figure(
  data   = data1,
  size   = 1,
  colour = "black",
  title  = "GDP vs life exp"
)
```



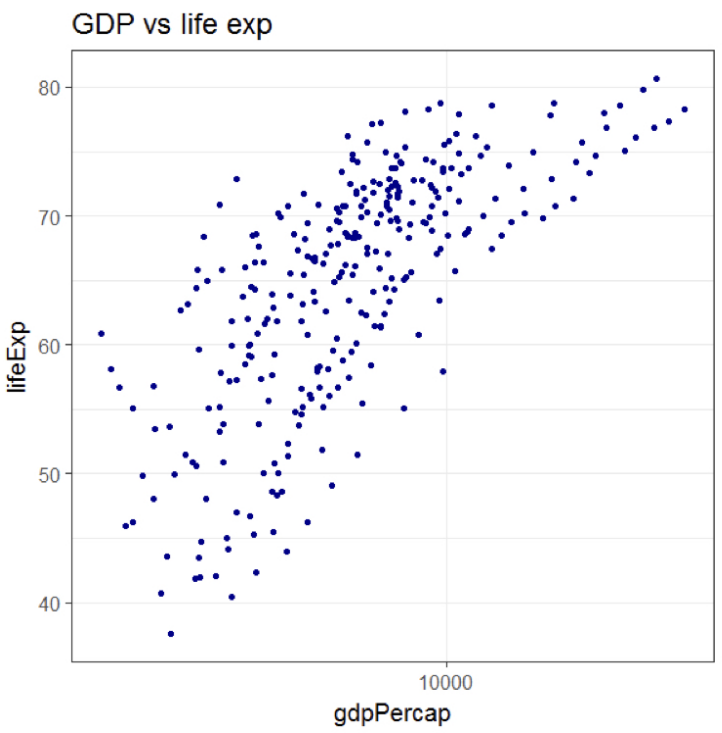
Re-plotting using R code

```
make_figure(
  data   = data1,
  size   = 2,
  colour = "black",
  title  = "GDP vs life exp"
)
```



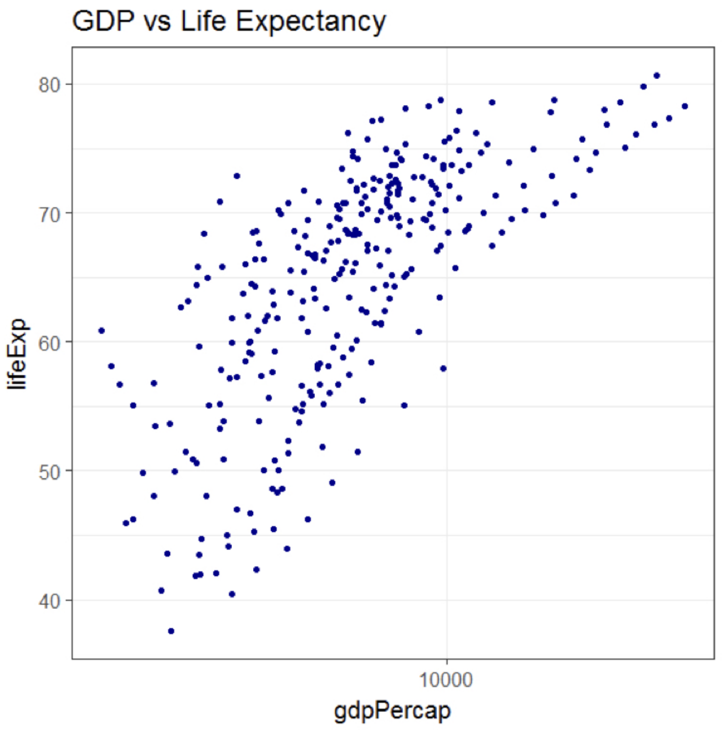
Re-plotting using R code

```
make_figure(
  data   = data1,
  size   = 2,
  colour = "darkblue",
  title  = "GDP vs life exp"
)
```



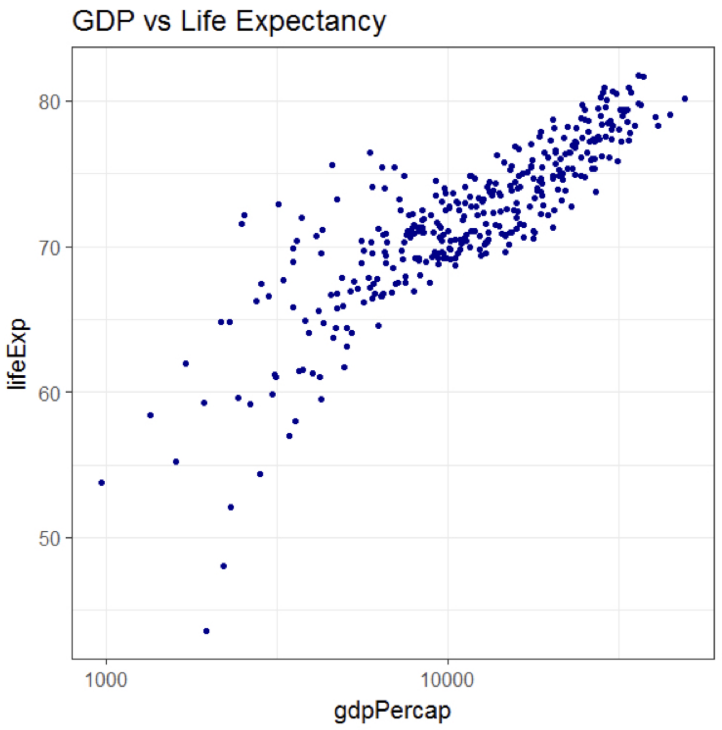
Re-plotting using R code

```
make_figure(
  data   = data1,
  size   = 2,
  colour = "darkblue",
  title  = "GDP vs Life Expectancy"
)
```

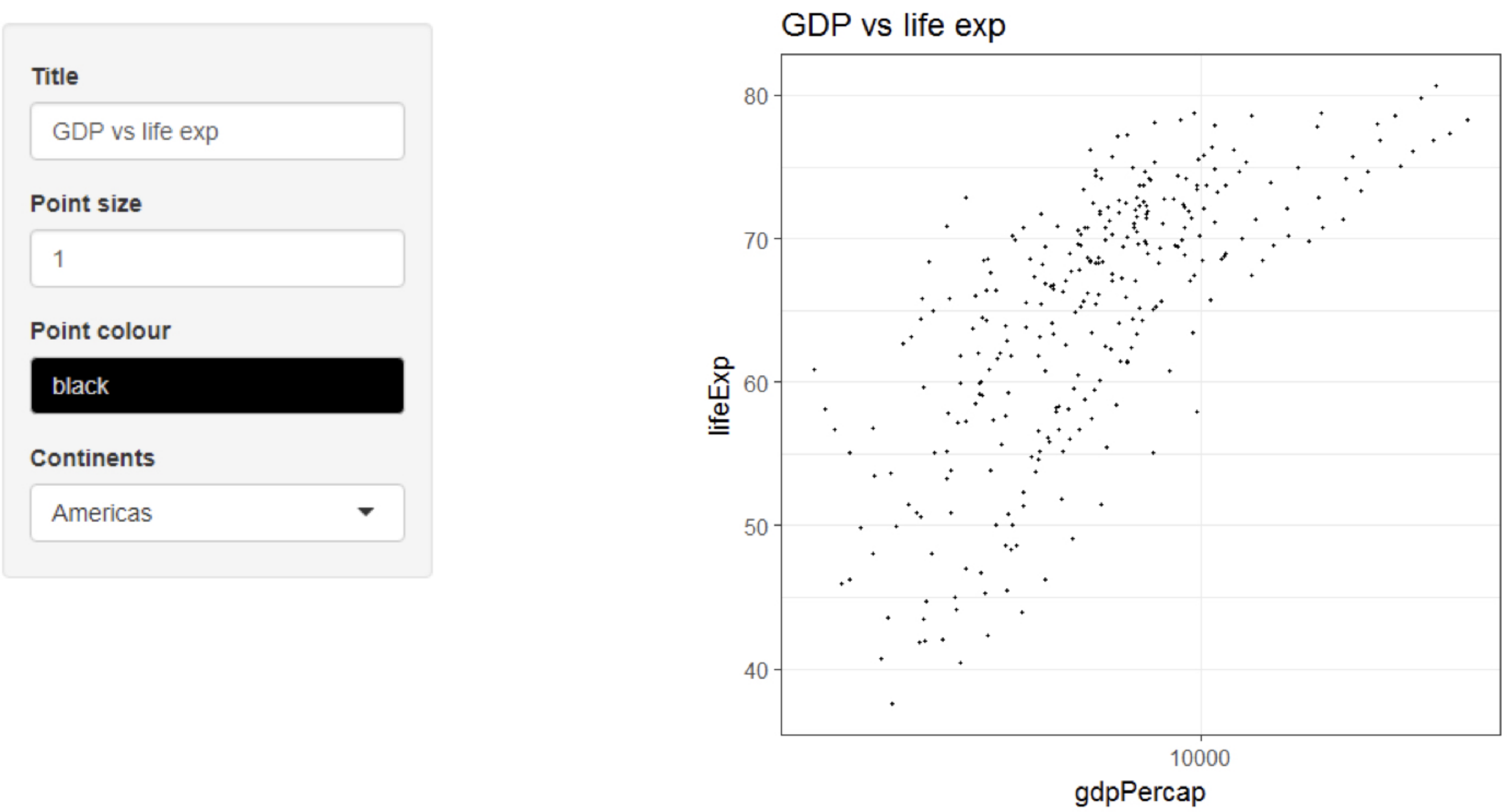


Re-plotting using R code

```
make_figure(
  data   = data2,
  size   = 2,
  colour = "darkblue",
  title  = "GDP vs Life Expectancy"
)
```



Re-plotting using Shiny



Title

GDP vs life exp

Point size

1

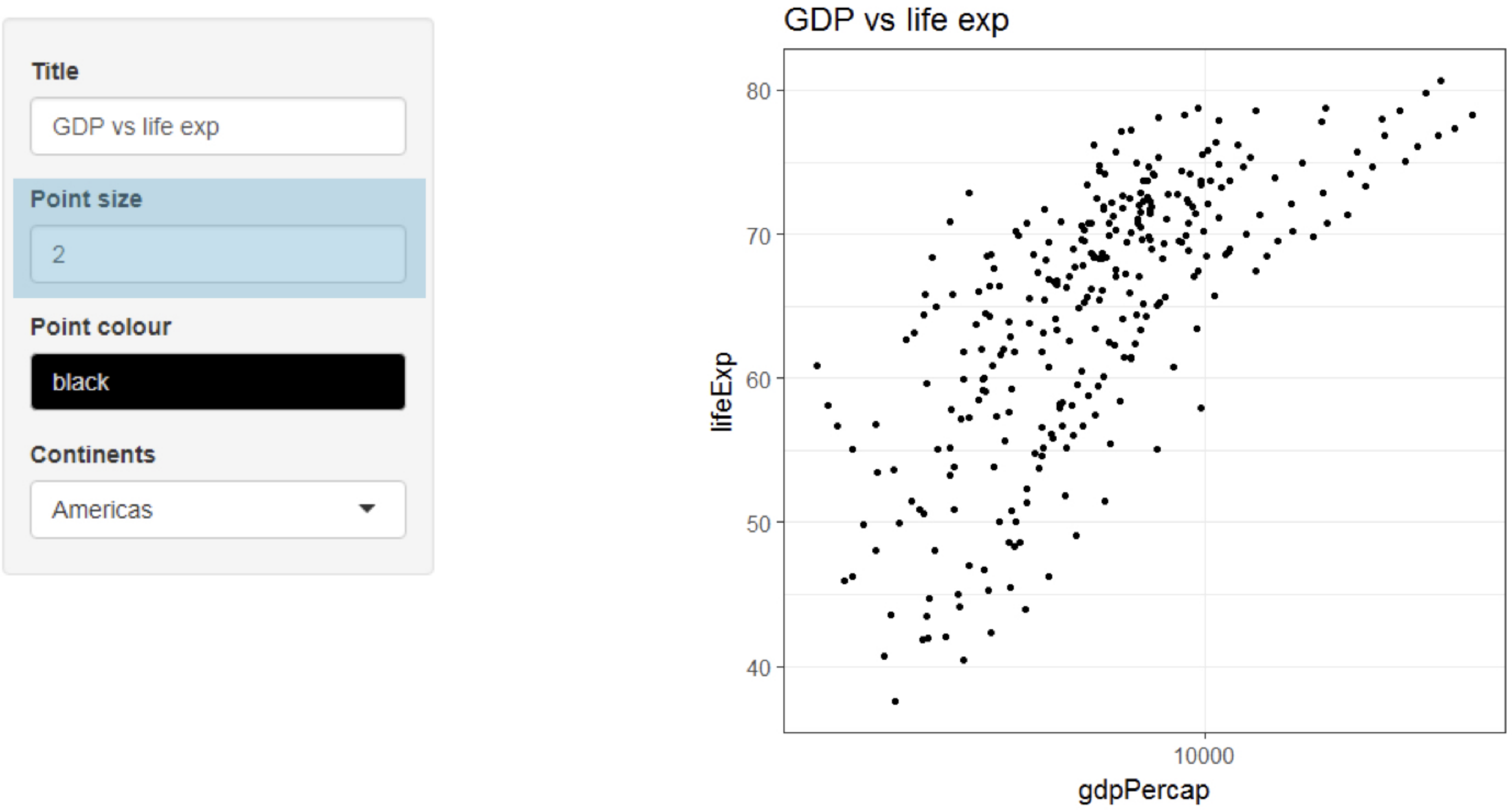
Point colour

black

Continents

Americas

Re-plotting using Shiny



Title

GDP vs life exp

Point size

2

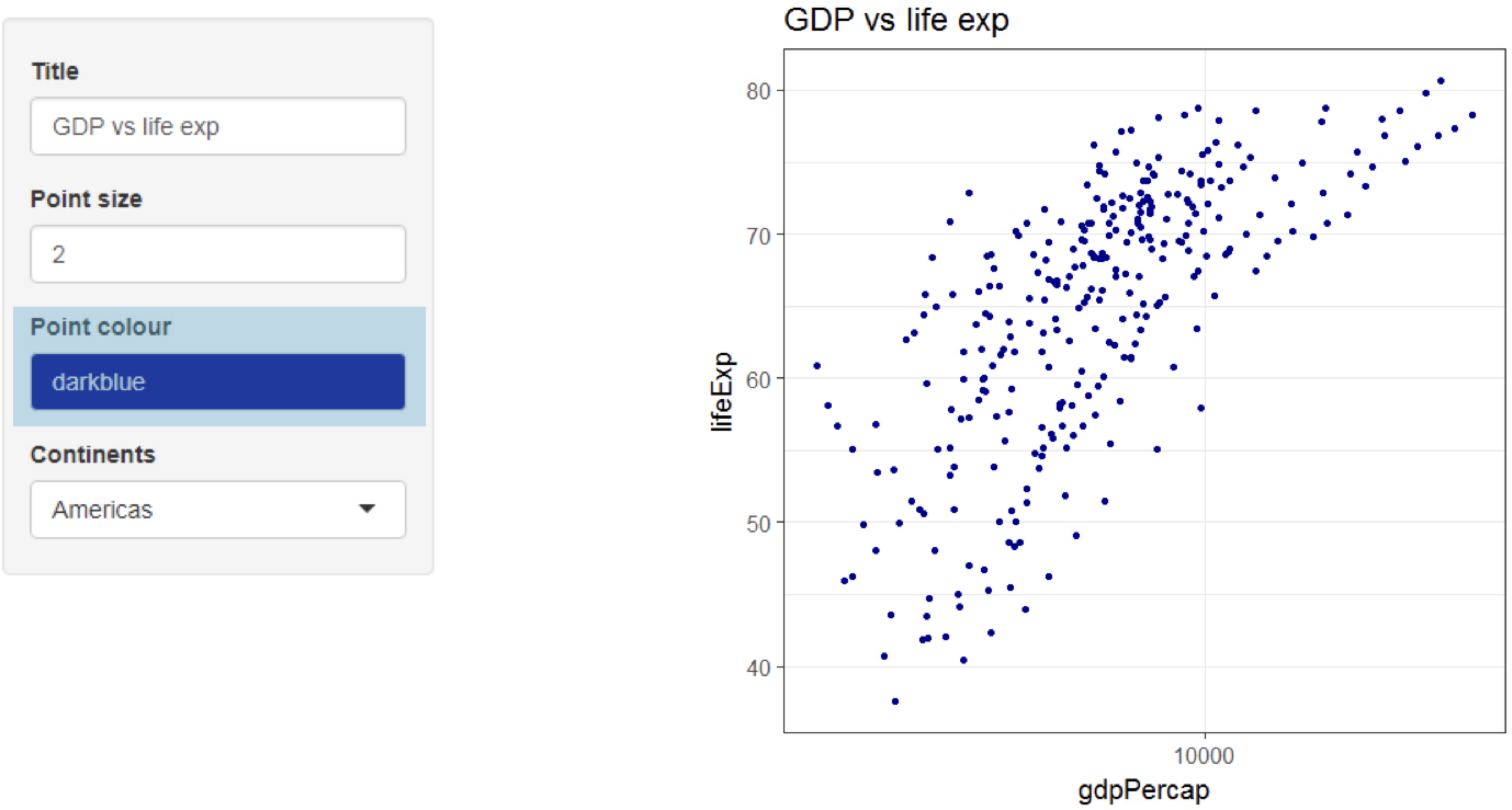
Point colour

black

Continents

Americas

Re-plotting using Shiny



Title

GDP vs life exp

Point size

2

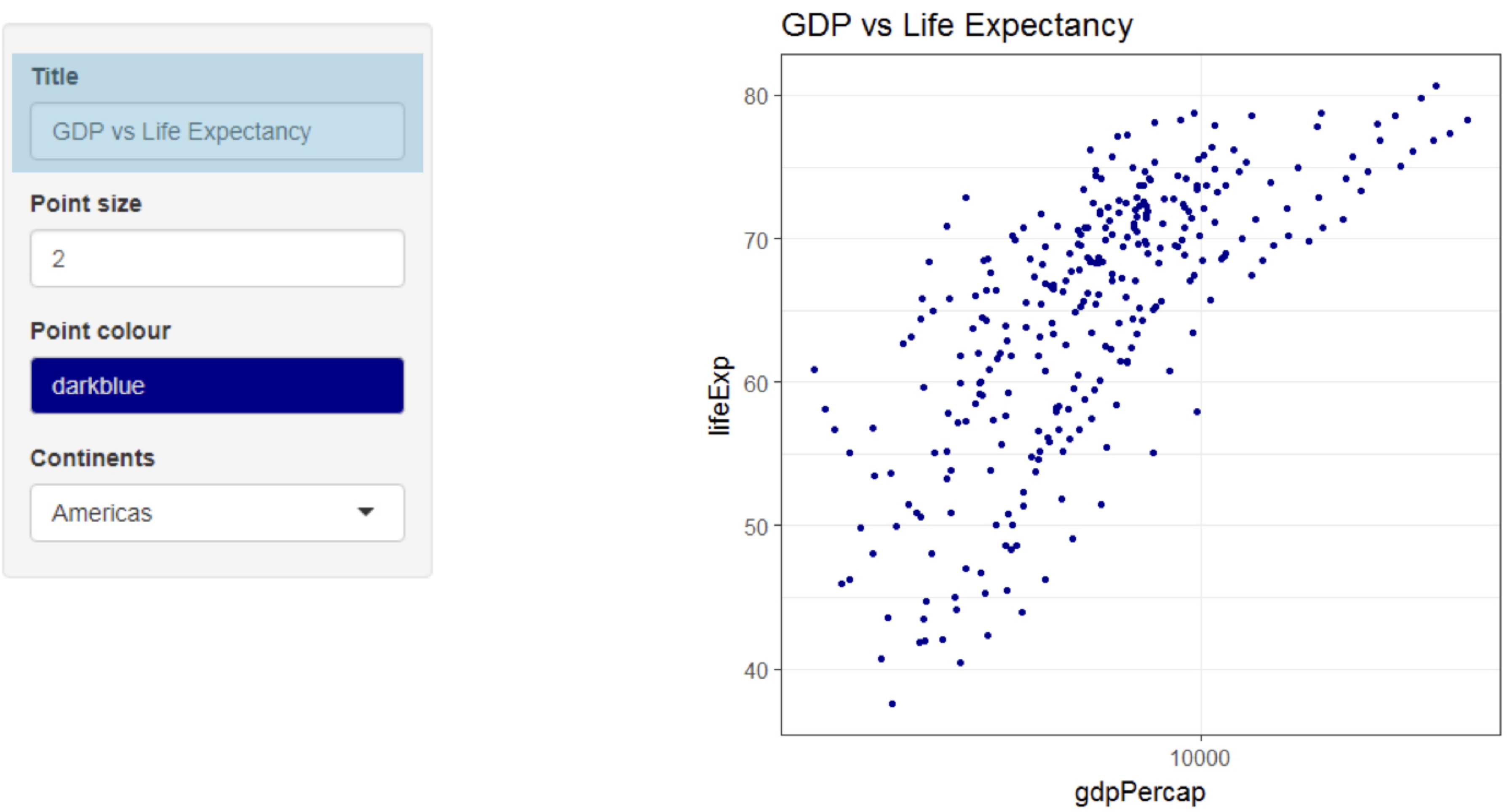
Point colour

darkblue

Continents

Americas

Re-plotting using Shiny



Title

GDP vs Life Expectancy

Point size

2

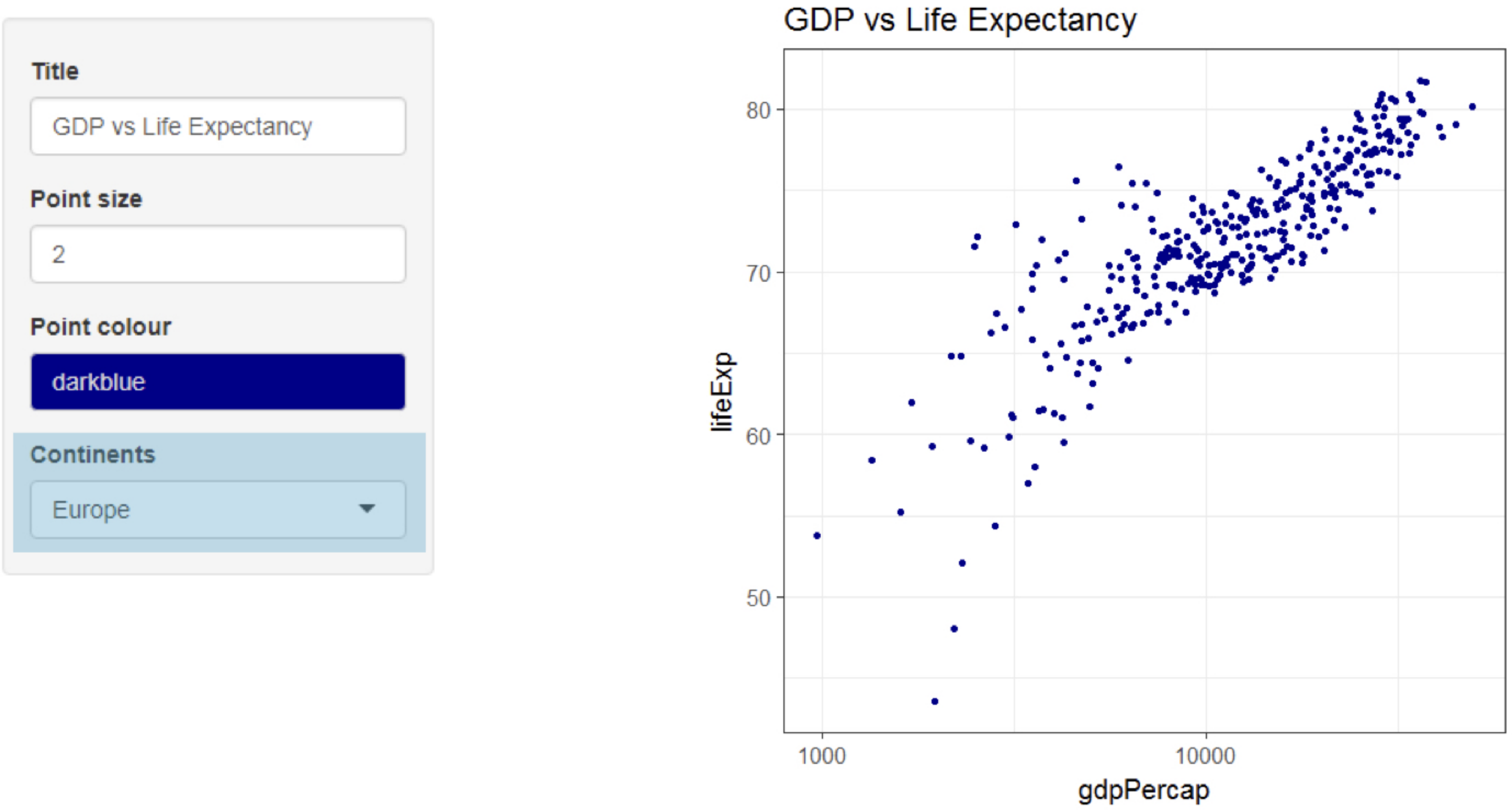
Point colour

darkblue

Continents

Americas

Re-plotting using Shiny



Title: GDP vs Life Expectancy

Point size: 2

Point colour: darkblue

Continents: Europe

Gapminder dataset

country	continent	year	lifeExp	pop	gdpPercap
Netherlands	Europe	2002	78.530	16122830	33724.758
Turkey	Europe	1987	63.108	52881328	5089.044
Oman	Asia	1987	67.734	1593882	18115.223
Jamaica	Americas	1982	71.210	2298309	6068.051
Algeria	Africa	1967	51.407	12760499	3246.992

Gapminder dataset

country	continent	year	lifeExp	pop	gdpPercap
Netherlands	Europe	2002	78.530	16122830	33724.758
Turkey	**Europe**	**1987**	**63.108**	**52881328**	**5089.044**
Oman	Asia	1987	67.734	1593882	18115.223
Jamaica	Americas	1982	71.210	2298309	6068.051
Algeria	Africa	1967	51.407	12760499	3246.992

Gapminder dataset

country	continent	year	lifeExp	pop	gdpPercap
Netherlands	Europe	2002	78.530	16122830	33724.758
Turkey	Europe	1987	**63.108**	52881328	5089.044
Oman	Asia	1987	67.734	1593882	18115.223
Jamaica	Americas	1982	71.210	2298309	6068.051
Algeria	Africa	1967	51.407	12760499	3246.992

Gapminder dataset

country	continent	year	lifeExp	pop	gdpPercap
Netherlands	Europe	2002	78.530	16122830	33724.758
Turkey	Europe	1987	63.108	**52881328**	5089.044
Oman	Asia	1987	67.734	1593882	18115.223
Jamaica	Americas	1982	71.210	2298309	6068.051
Algeria	Africa	1967	51.407	12760499	3246.992

Gapminder dataset

country	continent	year	lifeExp	pop	gdpPercap
Netherlands	Europe	2002	78.530	16122830	33724.758
Turkey	Europe	1987	63.108	52881328	**5089.044**
Oman	Asia	1987	67.734	1593882	18115.223
Jamaica	Americas	1982	71.210	2298309	6068.051
Algeria	Africa	1967	51.407	12760499	3246.992

Gapminder dataset

country	continent	year	lifeExp	pop	gdpPercap
Netherlands	Europe	2002	78.530	16122830	33724.758
Turkey	Europe	1987	63.108	52881328	5089.044
Oman	Asia	1987	67.734	1593882	18115.223
Jamaica	Americas	1982	71.210	2298309	6068.051
Algeria	Africa	1967	51.407	12760499	3246.992

Gapminder package

```
library(gapminder)
min(gapminder$pop)
```

```
60011
```

```
max(gapminder$pop)
```

```
1318683096
```

Gapminder package

```r
subset(gapminder, country == "Canada" & year < 1965)
```

```
    country continent year lifeExp      pop gdpPercap
241  Canada  Americas 1952   68.75 14785584  11367.16
242  Canada  Americas 1957   69.96 17010154  12489.95
243  Canada  Americas 1962   71.30 18985849  13462.49
```

```r
subset(gapminder, country == "Canada" & year == 1962)$lifeExp
```

```
71.3
```

Let's practice!

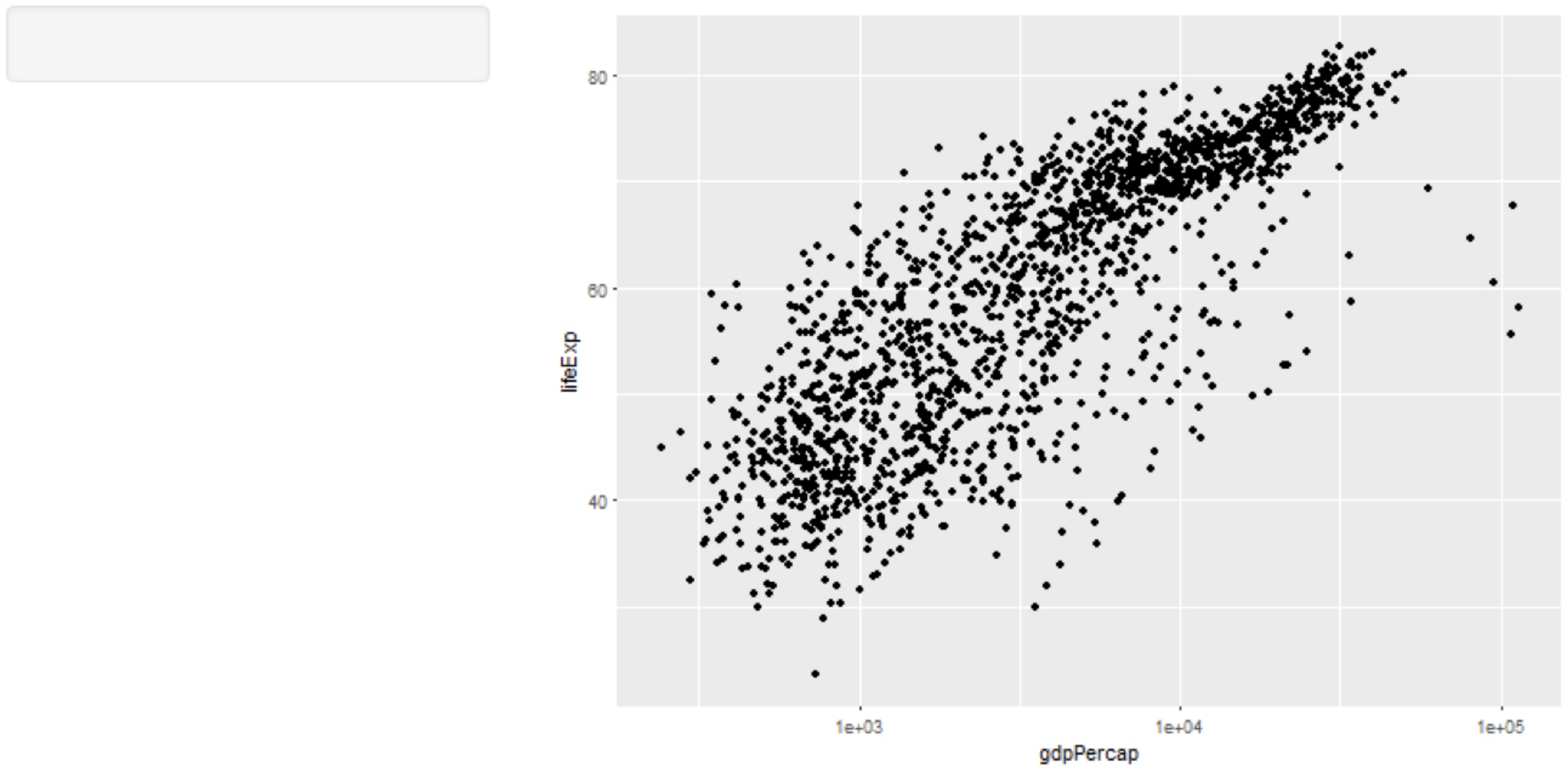
CASE STUDIES: BUILDING WEB APPLICATIONS WITH SHINY IN R

Adding simple inputs to modify a plot

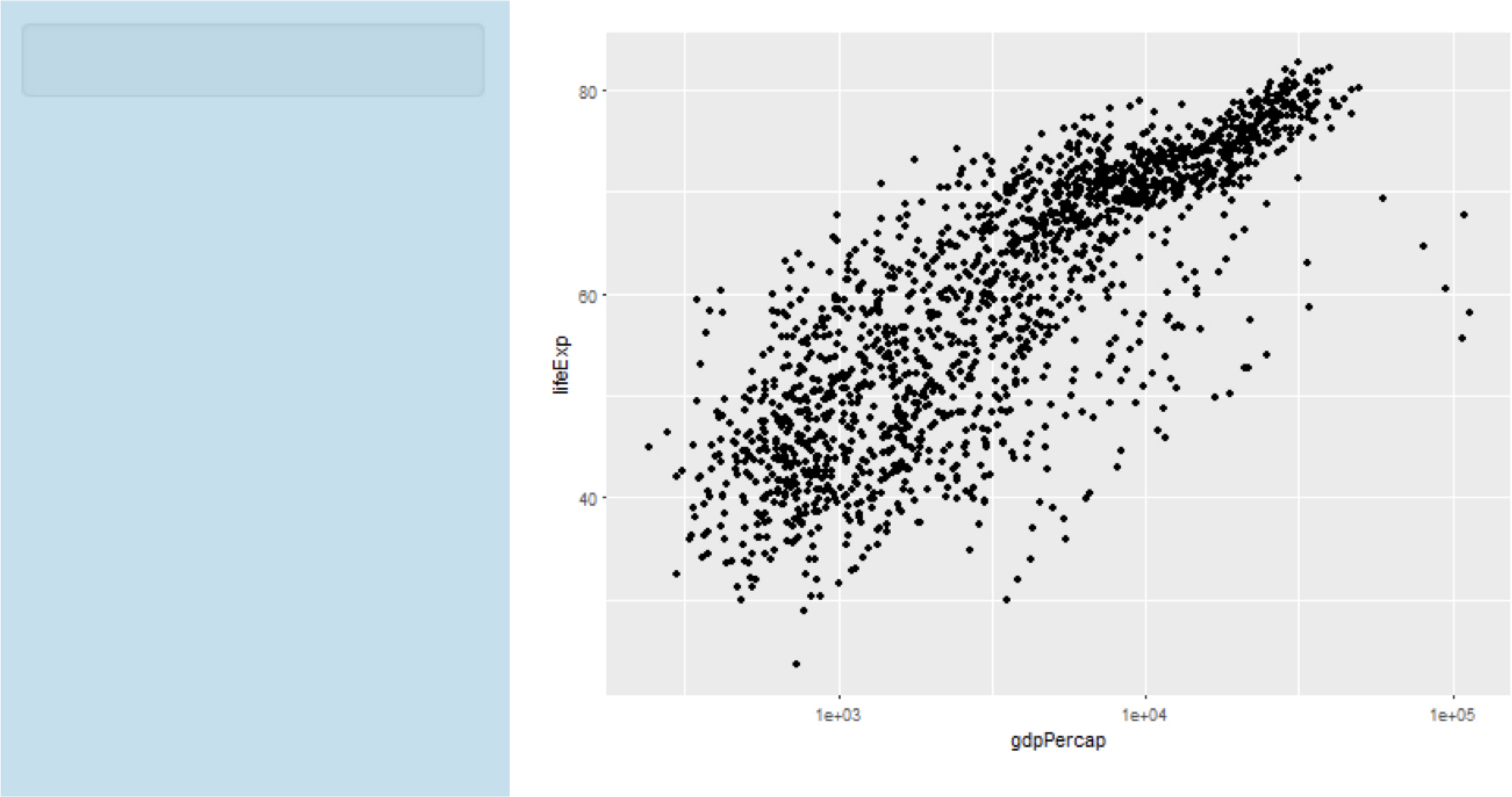
CASE STUDIES: BUILDING WEB APPLICATIONS WITH SHINY IN R

Dean Attali
Shiny Consultant

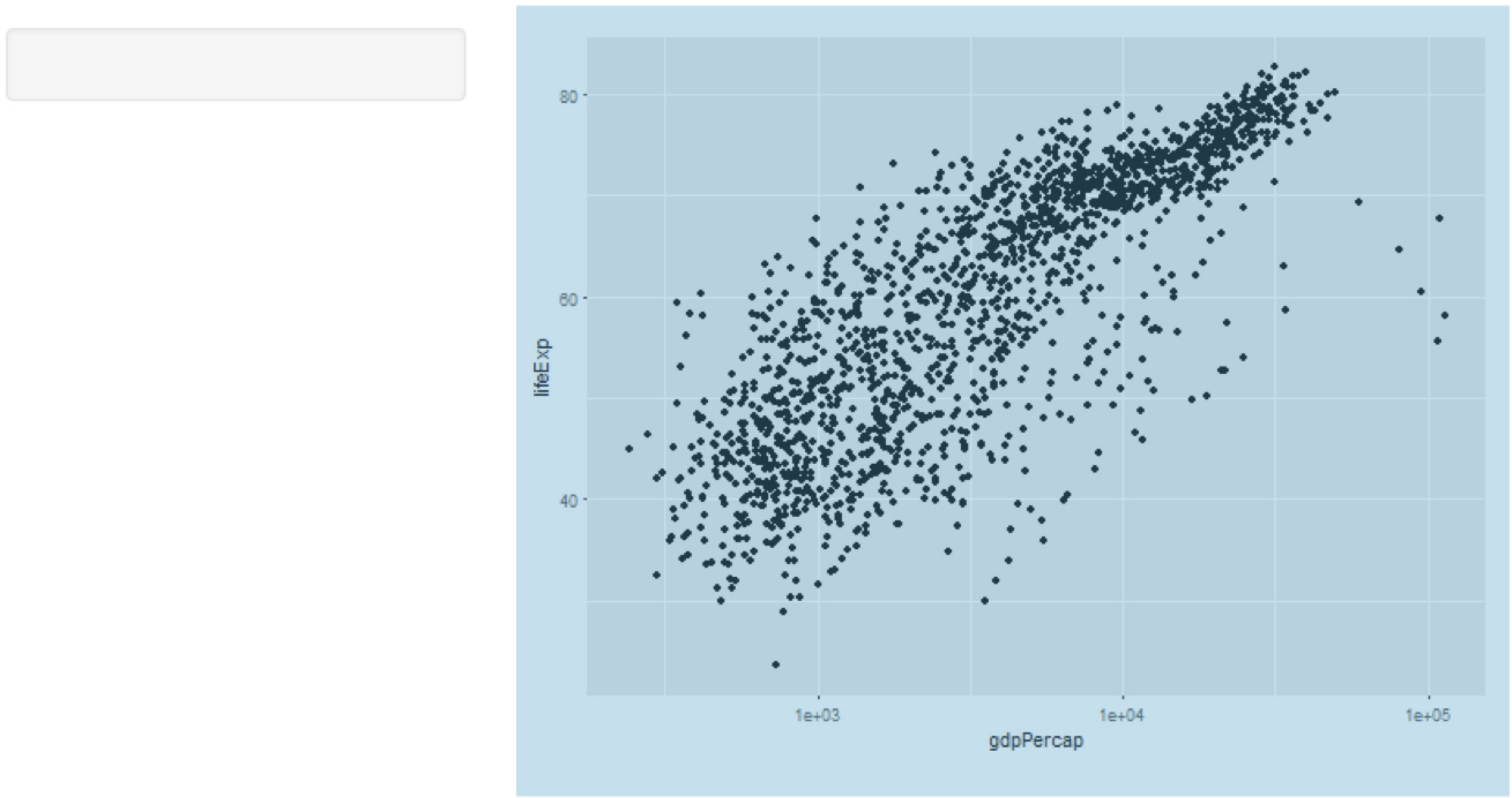
Gapminder plot app



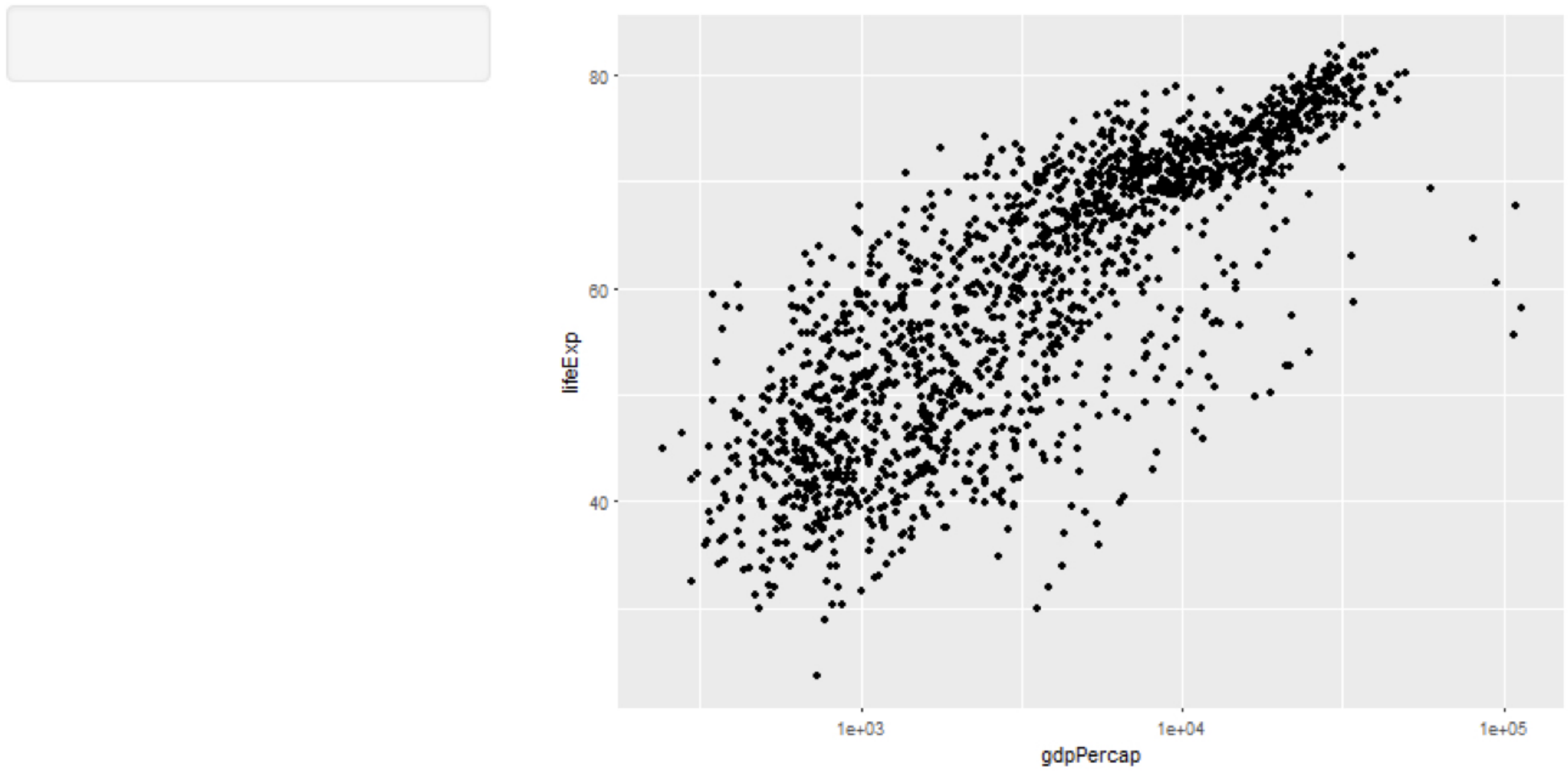
Gapminder plot app



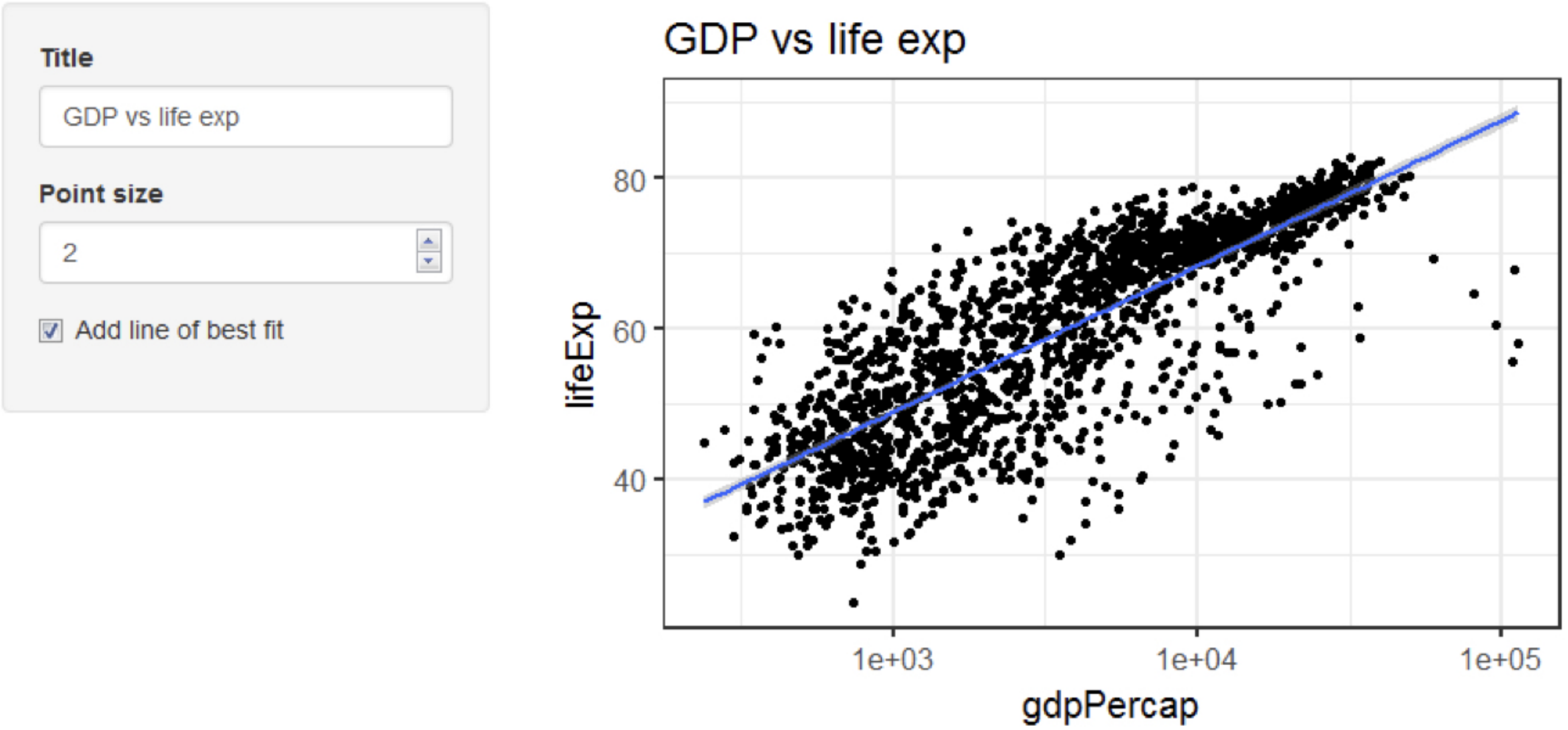
Gapminder plot app



Gapminder plot app



Gapminder plot app



Title

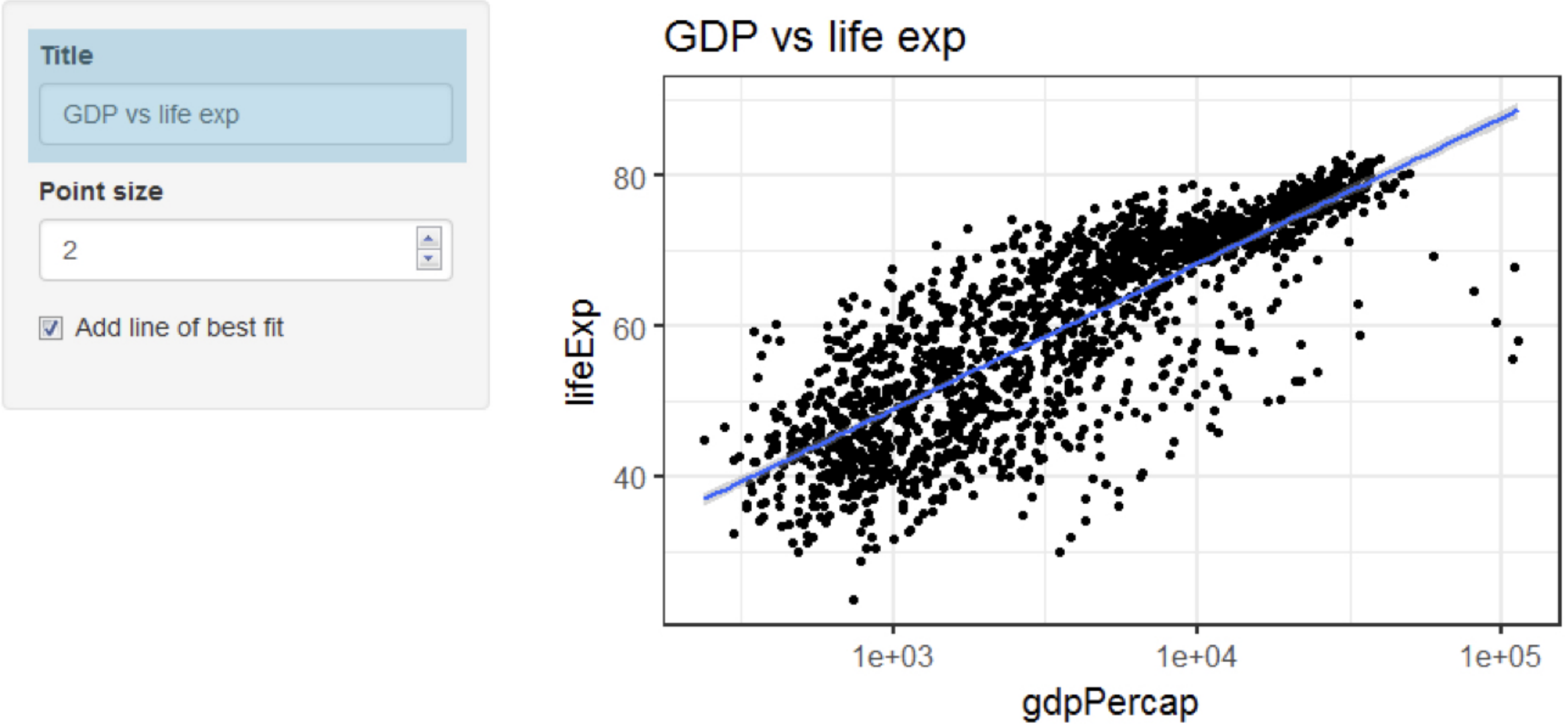
GDP vs life exp

Point size

2

☑ Add line of best fit

Gapminder plot app



Title

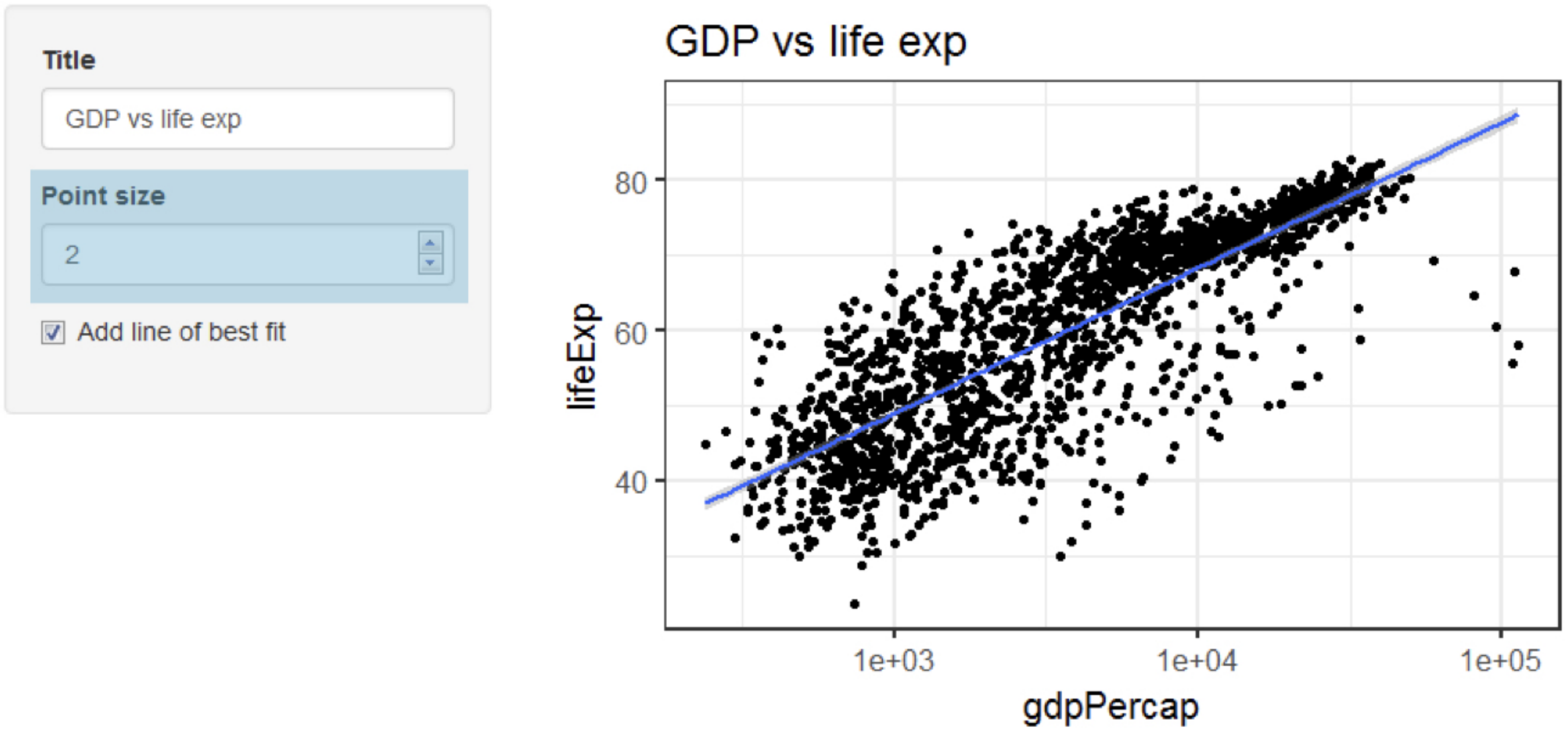
GDP vs life exp

Point size

2

☑ Add line of best fit

Gapminder plot app



Title

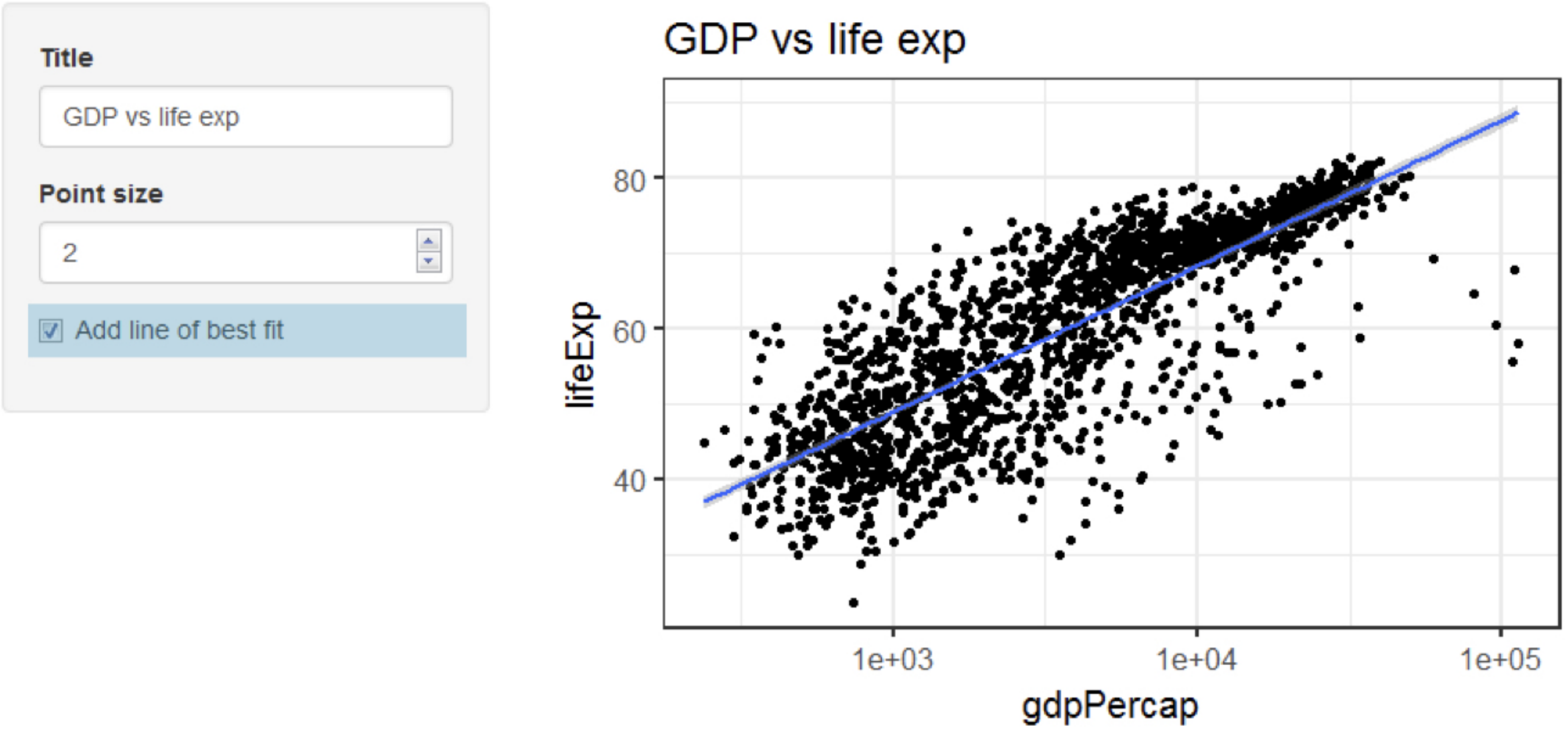
GDP vs life exp

Point size

2

☑ Add line of best fit

Gapminder plot app



Title

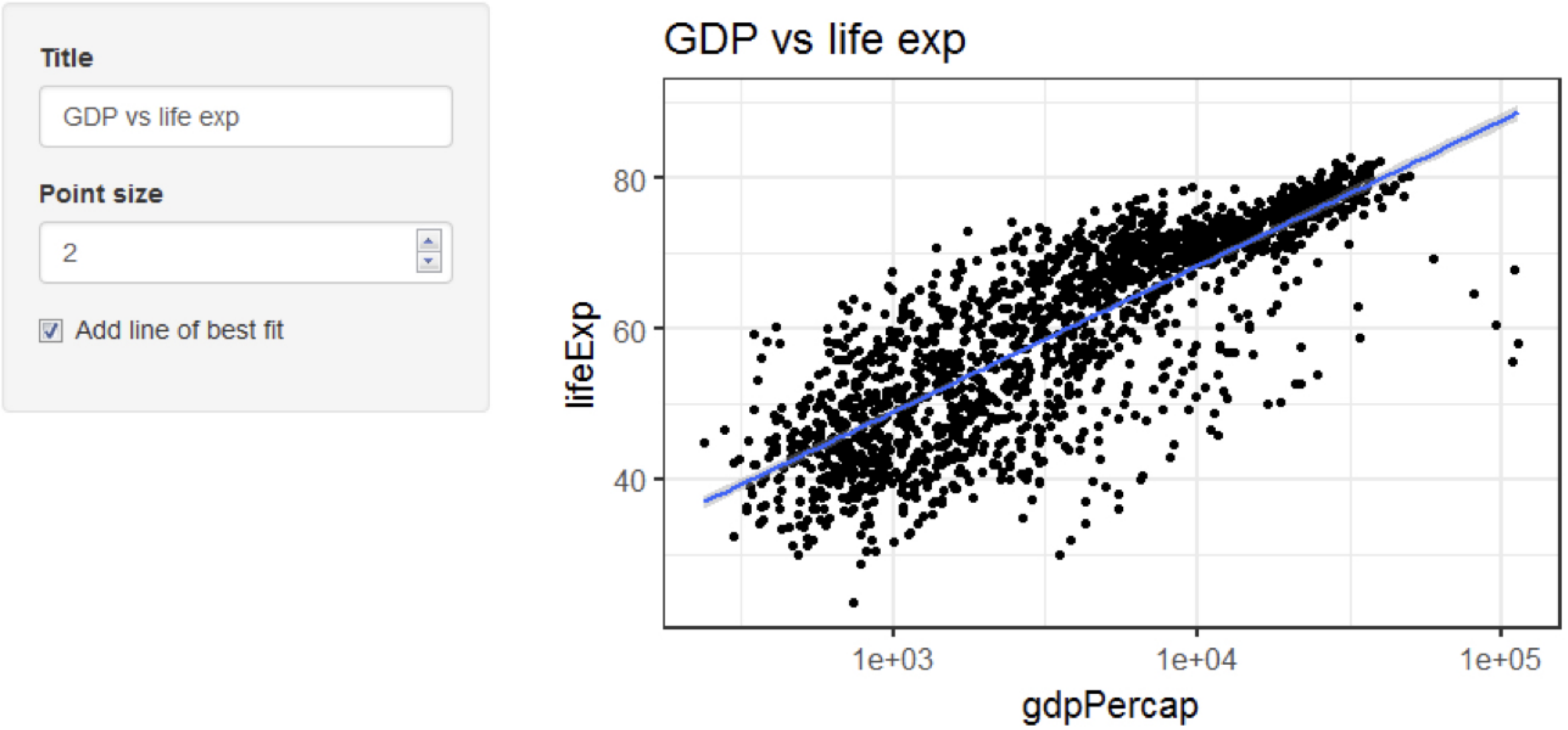
GDP vs life exp

Point size

2

☑ Add line of best fit

Gapminder plot app



Title

GDP vs life exp

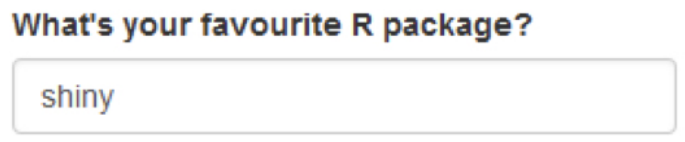
Point size

2

☑ Add line of best fit

Text inputs

```r
ui <- fluidPage(
    textInput(inputId = "package",
              label = "What's your favourite R package?",
              value = "shiny")
)
server <- function(input, output) {}
shinyApp(ui, server)
```



What's your favourite R package?

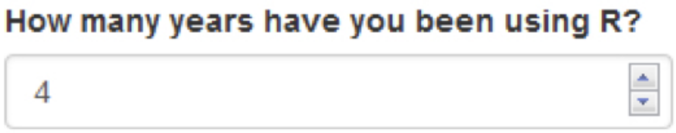
shiny

```r
str(input$package)
```

```
chr "shiny"
```

Numeric inputs

```
numericInput("years", "How many years have you been using R?",
             value = 4, min = 0, max = 25)
```



How many years have you been using R?

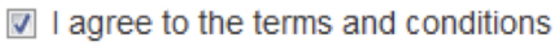
4

```
str(input$years)
```

```
int 4
```

Checkbox inputs

```r
checkboxInput("agree", "I agree to the terms and conditions",
              value = TRUE)
```



☑ I agree to the terms and conditions

```r
str(input$agree)
```

```
logi TRUE
```

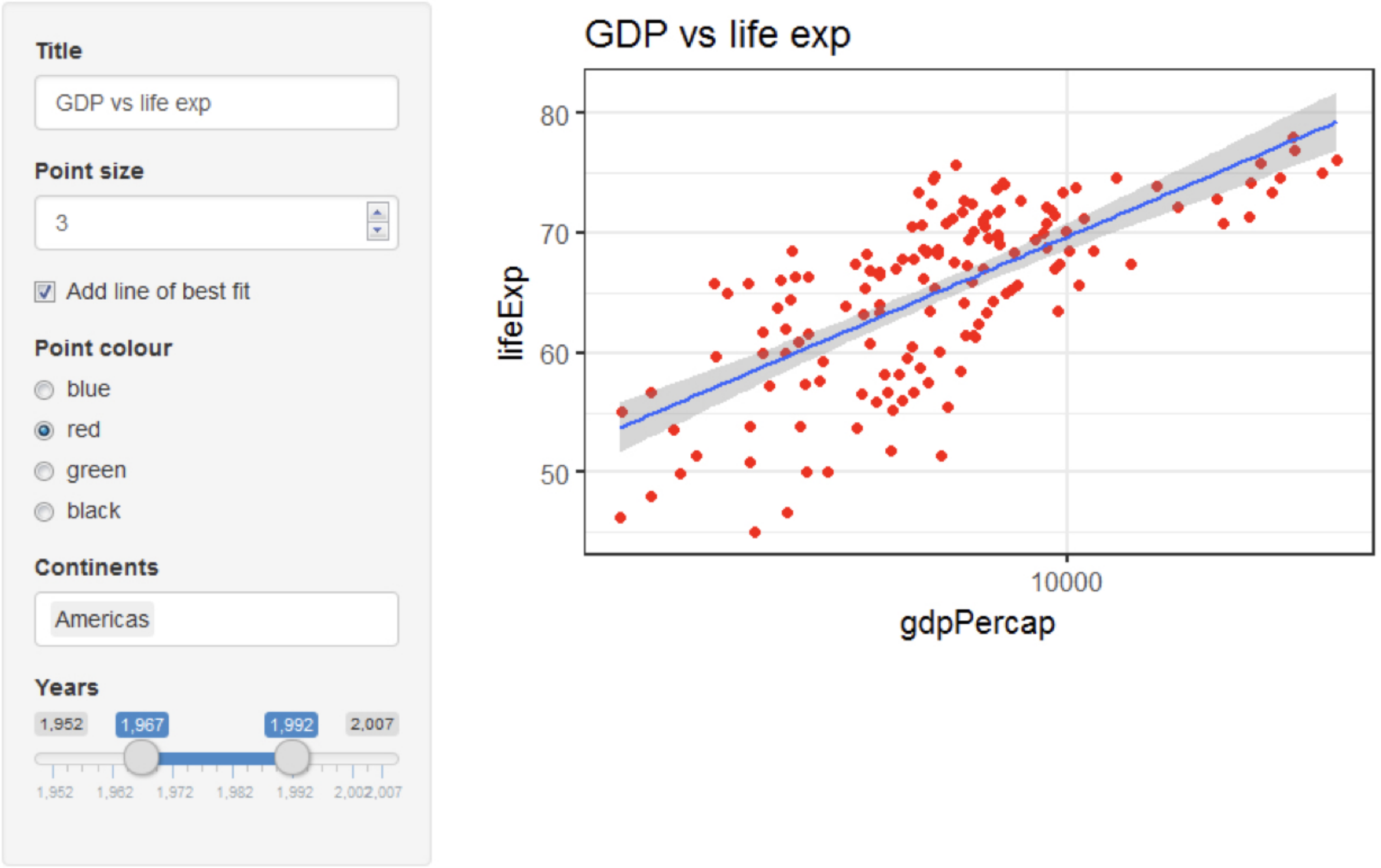
Let's practice!

More input types

CASE STUDIES: BUILDING WEB APPLICATIONS WITH SHINY IN R

Dean Attali
Shiny Consultant

Gapminder plot app



Title

GDP vs life exp

Point size

3

☑ Add line of best fit

Point colour

○ blue

● red

○ green

○ black

Continents

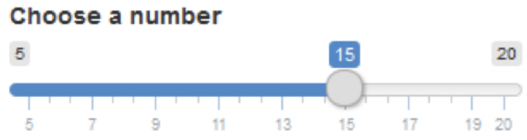
Americas

Years

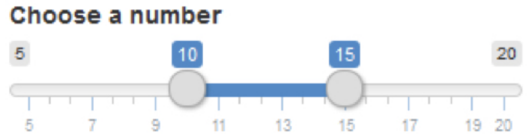
1,952 1,967 1,992 2,007

Slider inputs

```
sliderInput("slider", "Choose a number",
            value = 15, min = 5, max = 20)
```



```
sliderInput("slider2", "Choose a number",
            value = c(10, 15), min = 5, max = 20)
```

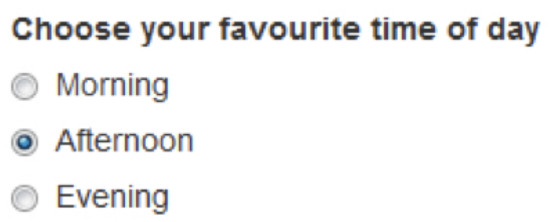


```
str(input$slider2)
```

```
num [1:2] 10 15
```

Radio buttons

```
radioButtons("radio", "Choose your favorite time of day",
             choices = c("Morning", "Afternoon", "Evening"),
             selected = "Afternoon")
```



Choose your favourite time of day

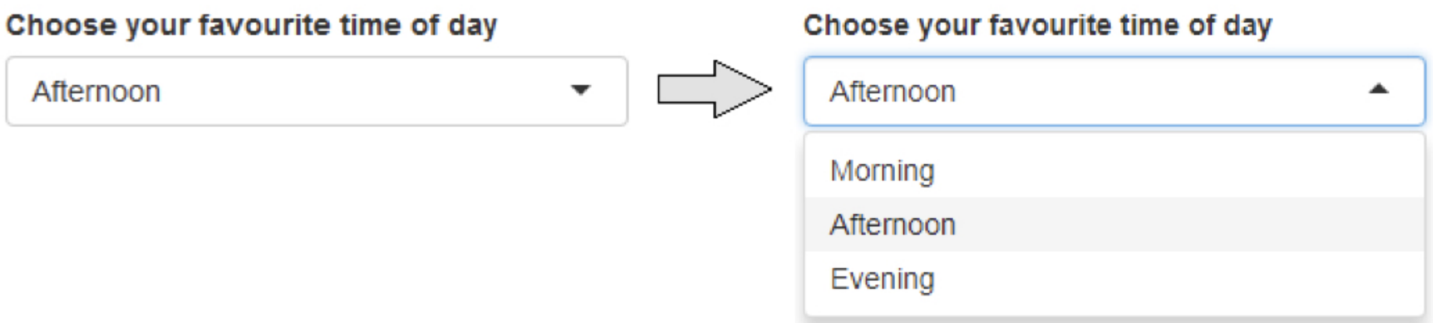
○ Morning

● Afternoon

○ Evening

Select inputs (dropdowns)

```
selectInput("select", "Choose your favorite time of day",
            choices = c("Morning", "Afternoon", "Evening"),
            selected = "Afternoon")
```



Choose your favourite time of day

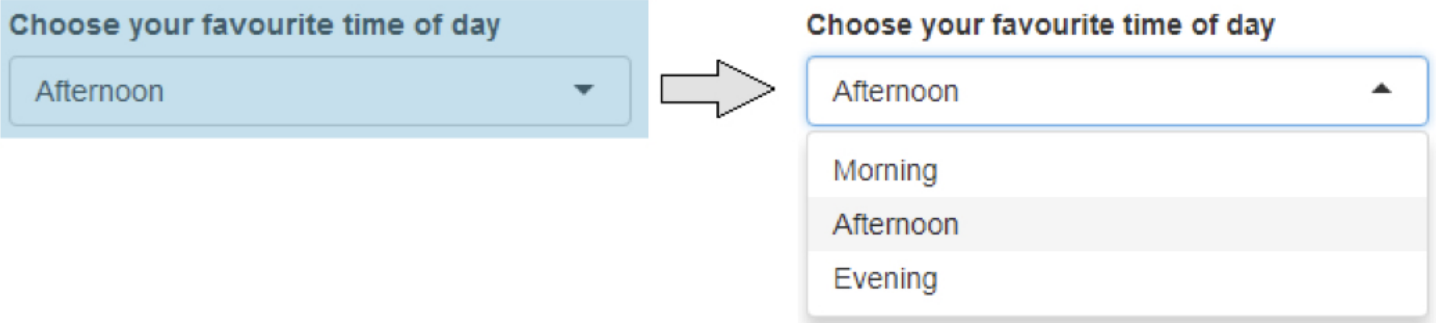
Afternoon

Choose your favourite time of day

Afternoon

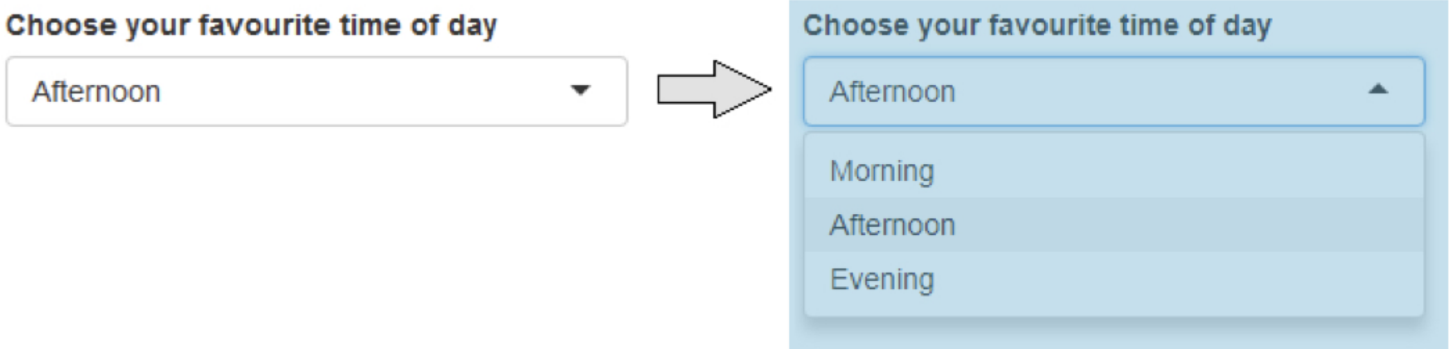
Select inputs (dropdowns)

```r
selectInput("select", "Choose your favorite time of day",
            choices = c("Morning", "Afternoon", "Evening"),
            selected = "Afternoon")
```


Choose your favourite time of day
Afternoon
Choose your favourite time of day
Afternoon
Morning
Afternoon
Evening

Select inputs (dropdowns)

```
selectInput("select", "Choose your favorite time of day",
            choices = c("Morning", "Afternoon", "Evening"),
            selected = "Afternoon")
```



Choose your favourite time of day

Afternoon

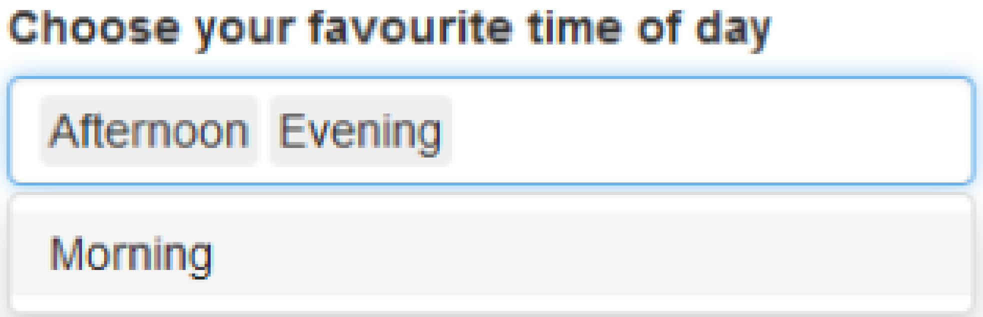
Choose your favourite time of day

Afternoon

Select inputs (dropdowns)

- Allow multiple selections

```
selectInput("select", "Choose your favorite time of day",
            choices = c("Morning", "Afternoon", "Evening"),
            selected = c("Afternoon", "Evening"),
            multiple = TRUE)
```



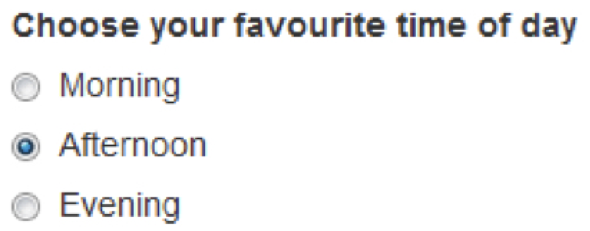
Choose your favourite time of day

Afternoon Evening

Morning

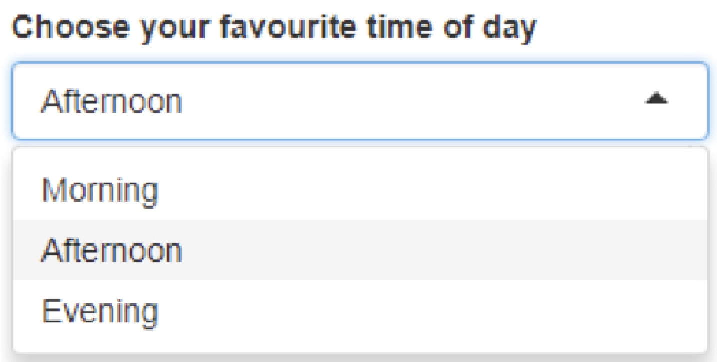
Radio buttons vs select inputs

- Radio buttons
  - Few options
  - All options are visible
  - Exactly one option selected
- Select inputs
  - Few or many options
  - Harder to see all options
  - Multiple options can be selected



Choose your favourite time of day

○ Morning
● Afternoon
○ Evening



Choose your favourite time of day

Afternoon ▲

Morning
Afternoon
Evening

Let's practice!

CASE STUDIES: BUILDING WEB APPLICATIONS WITH SHINY IN R

Advanced features to improve your plot

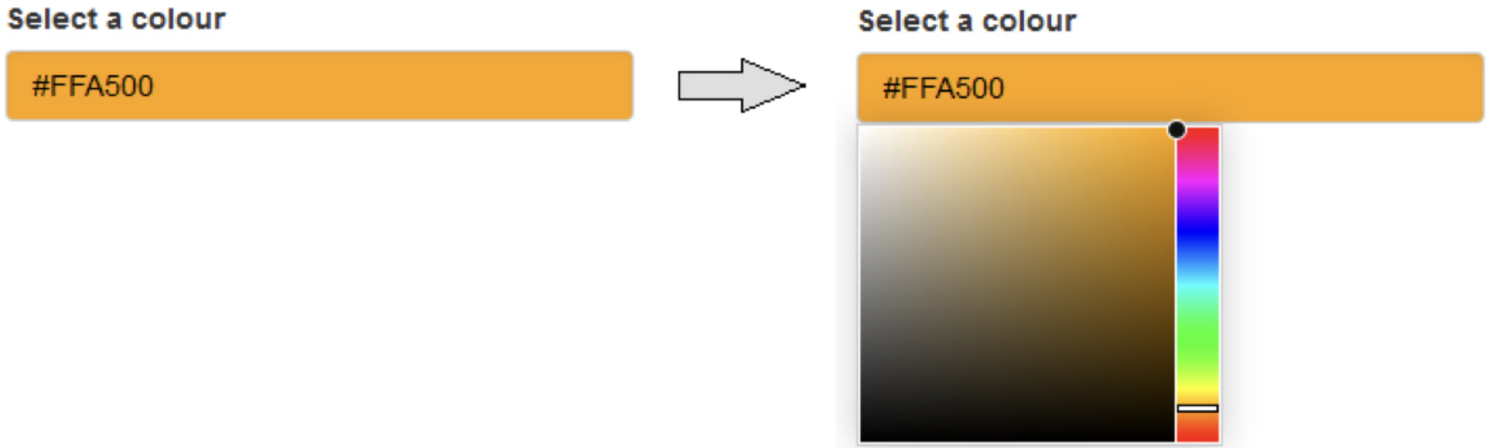
CASE STUDIES: BUILDING WEB APPLICATIONS WITH SHINY IN R

Dean Attali
Shiny Consultant

datacamp

Colour input

```
library(colourpicker)
colourInput("col", "Select a colour", value = "orange")
```



Select a colour

#FFA500

Select a colour

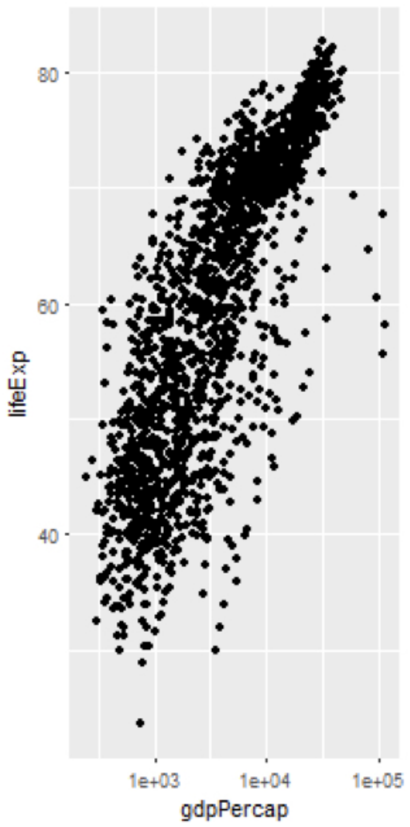
#FFA500

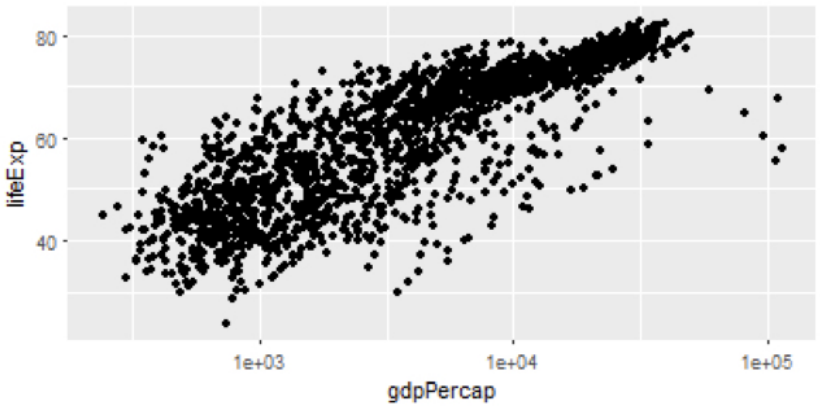
Outputs can have arguments

```
plotOutput(outputId, width = "100%", height = "400px",
           click = NULL, dblclick = NULL, hover = NULL,
           hoverDelay = NULL, hoverDelayType = NULL,
           brush = NULL, clickId = NULL, hoverId = NULL,
           inline = FALSE)
```

Plot output arguments

```
plotOutput("plot1", width = 200, height = 400)
plotOutput("plot2", width = 400, height = 200)
```

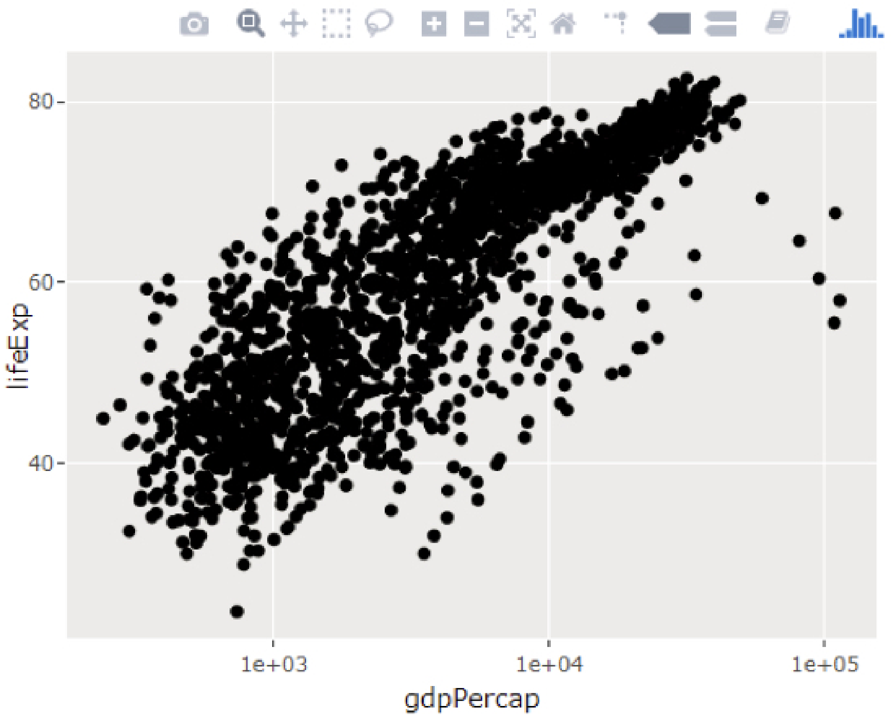




Interactive plots with plotly

- Many packages for interactive plots
- `plotly` is popular choice
- `ggplotly()` :
  - `ggplot2` plot ⇒ interactive

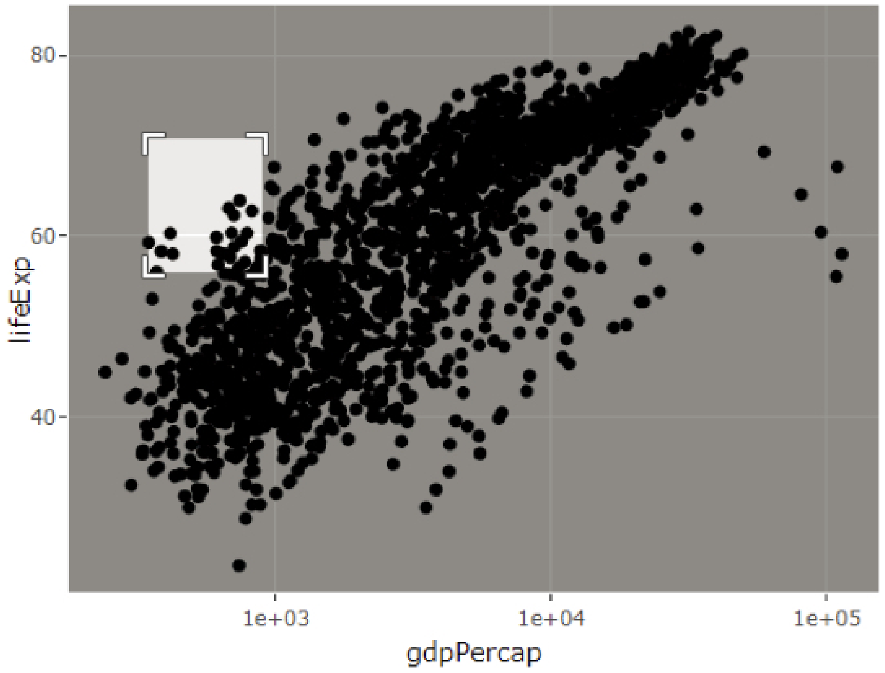
```
p <- ggplot(gapminder,
            aes(gdpPercap, lifeExp)) +
     geom_point() +
     scale_x_log10()
library(plotly)
ggplotly(p)
```



Interactive plots with plotly

- Many packages for interactive plots
- `plotly` is popular choice
- `ggplotly()` :
    - `ggplot2` plot ⇒ interactive

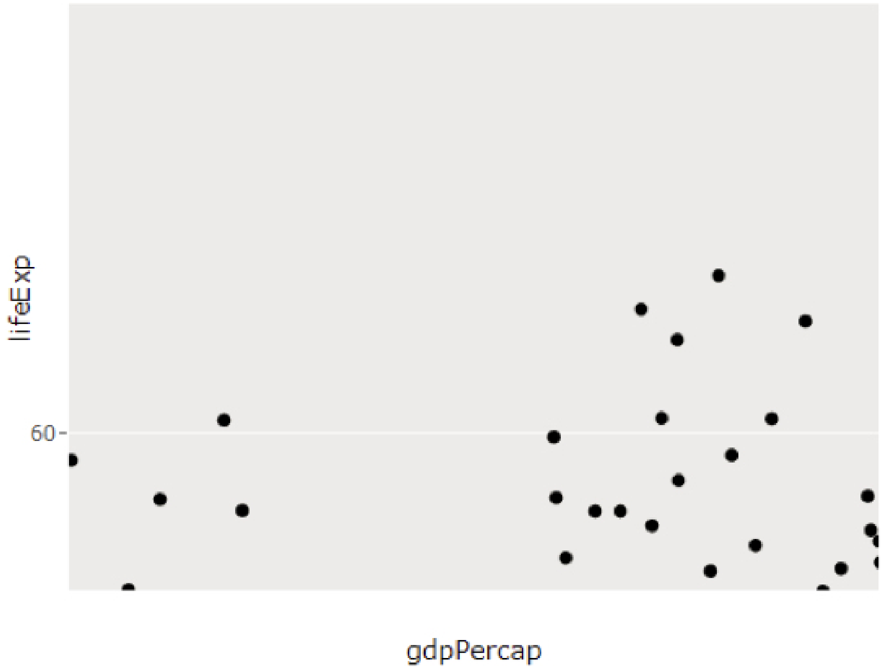
```
p <- ggplot(gapminder,
            aes(gdpPercap, lifeExp)) +
     geom_point() +
     scale_x_log10()
library(plotly)
ggplotly(p)
```



Interactive plots with plotly

- Many packages for interactive plots
- `plotly` is popular choice
- `ggplotly()` :
    - `ggplot2` plot ⇒ interactive

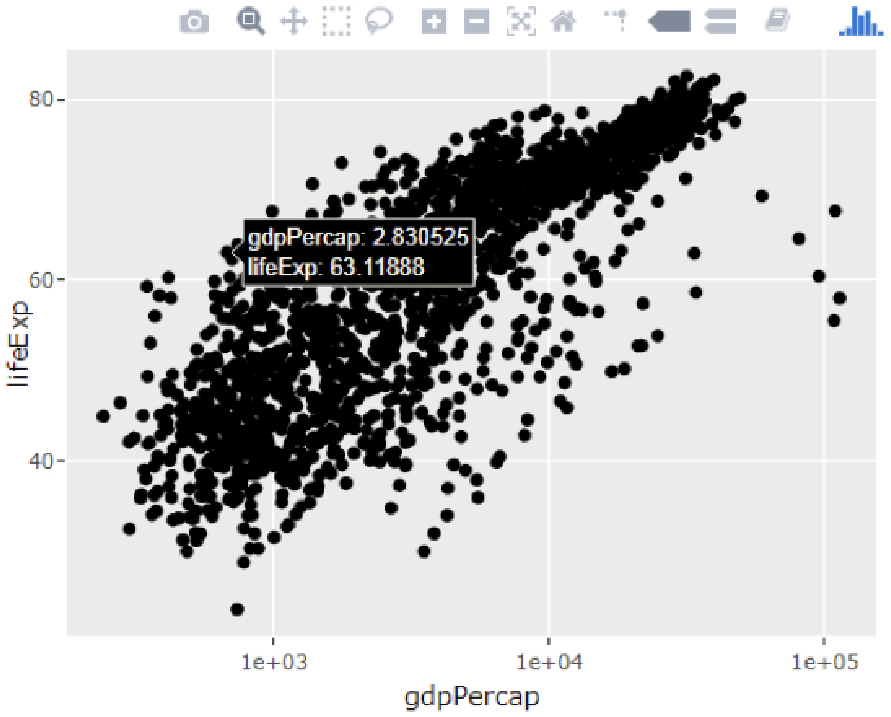
```
p <- ggplot(gapminder,
            aes(gdpPercap, lifeExp)) +
     geom_point() +
     scale_x_log10()
library(plotly)
ggplotly(p)
```



Interactive plots with plotly

- Many packages for interactive plots
- `plotly` is popular choice
- `ggplotly()` :
  - `ggplot2` plot ⇒ interactive

```
p <- ggplot(gapminder,
            aes(gdpPercap, lifeExp)) +
     geom_point() +
     scale_x_log10()
library(plotly)
ggplotly(p)
```



Plotly in Shiny

- Incorrect

```
plotOutput("plot")
```

```
renderPlot(ggplotly(p))
```

- Correct

```
plotlyOutput("plot")
```

```
renderPlotly(ggplotly(p))
```

Let's practice!

CASE STUDIES: BUILDING WEB APPLICATIONS WITH SHINY IN R