

Module화를 통한 shiny app 개발

문건웅

가톨릭대학교 성빈센트병원

Shiny Module

Keon-Woong Moon

2018/10/26

강의의 진행

이번 강의는 live coding으로 진행되므로 및 RStudio가 설치된 노트북이 필요하다

패키지 설치

이번 강의를 위해 다음 패키지의 설치가 필요하다.

- `install.packages(c("ggplotAssist"))`
- `devtools::install_github("cardiomoon/rrtable")`
- `devtools::install_github("cardiomoon/editData")`

예제 소스 파일

이번 강의에 사용되는 예제 소스 파일들은 다음 github에서 다운로드 받을수 있다.

<https://github.com/cardiomoon/shinyLecture2>

- `tower_of_hanoi.R` # 하노이의 탑
- `/inst/` 폴더의 모든 파일들

Scoping Rule(1)

```
x <- 1
y <- 2

sum=function(x,y){
  x<-x+y
  x
}
sum(x,y)
x
```

sum(x,y) 및 x의 출력은?

Scoping Rule(1-1)

```
x <- 1
y <- 2

sum=function(x,y){
  x<-x+y
  x
}
sum(x,y)
```

```
[1] 3
```

```
x
```

```
[1] 1
```

Scoping Rule(2)

```
x <- 1
y <- 2

sum=function(x,y){
  x<<-x+y
  x
}
sum(x,y)
x
```

sum(x,y) 및 x의 출력은?

Scoping Rule(2-1)

```
x <- 1
y <- 2

sum=function(x,y){
  x<<-x+y
  x
}
sum(x,y)
```

```
[1] 1
```

```
x
```

```
[1] 3
```

Scoping Rule(2-2)

```
x <- 1
y <- 2

sum=function(a,b){
  x<-a+b
  x
}
sum(x,y)
x
```


Scoping Rule(2-3)

```
x <- 1
y <- 2

sum=function(a,b){
  x<-a+b
  x
}
sum(x,y)
```

```
[1] 3
```

```
x
```

```
[1] 3
```

Scoping rules for Shiny Apps(1)

```
library(shiny)

### objects(1)

ui=fluidPage(
  verbatimTextOutput("text")
)
server=function(input,output,session){
  ### objects(2)

  output$text=renderPrint({
    ### objects(3)
    ...
  })
}
shinyApp(ui,server)
```

Scoping rules for Shiny Apps(2)

```
library(shiny)

### objects(1) ; are shared across all sessions in the same R process

ui=fluidPage(
  verbatimTextOutput("text")
)
server=function(input,output,session){
  ### objects(2) ; Objects here are defined in each session

  output$text=renderPrint({
    ### objects(3) ; are defined each time this function is called
    ...
  })
}
shinyApp(ui,server)
```

Exercise

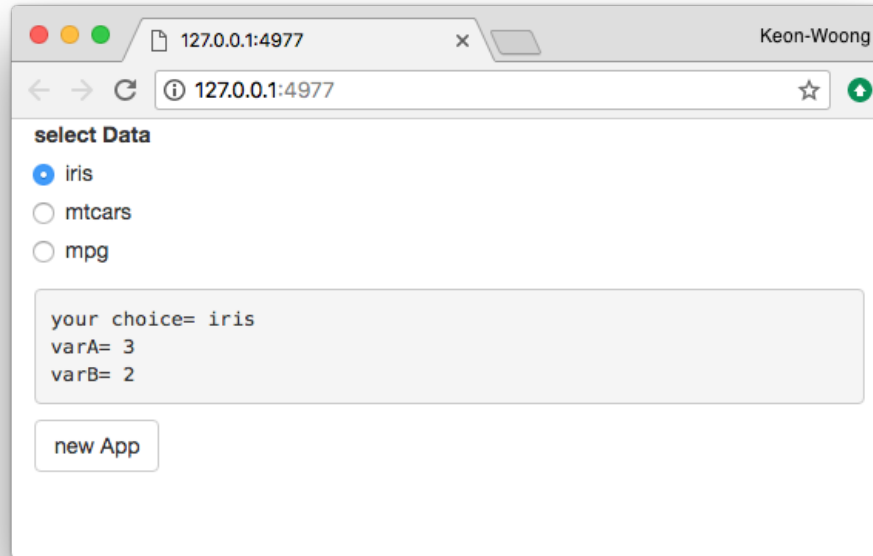
```
varA <- 1
varB <- 1

ui=fluidPage(
  radioButtons("select","select Data",choices=c("iris","mtcars","mpg"),
  verbatimTextOutput("text"),
  actionButton("newApp","new App")
)
server=function(input,output,session){

  varA <- varA + 1
  varB <<- varB + 1

  output$text=renderPrint({
    cat("your choice=",input$select,"\n")
    varA <<- varA+1
    cat("varA=",varA,"\n")
    cat("varB=",varB,"\n")
  })
}
```

Exercise : Scoping Rule



```
shiny::runGitHub('shinyLecture2', 'cardiomoon', subdir='inst/app21')
```

Modularizing Shiny App Code

- shiny app의 규모가 점점 커짐에 따라 namespace 문제가 발생
- shiny app의 input과 output의 ID는 global namespace를 share하고 있다.
- server function에서 input과 output을 만들 때 ID가 서로 충돌하면 안됨
- 이를 해결하기 위해 shiny app의 일부를 shiny module로 제작

모듈화의 장점

- 재사용이 쉽다.
- 재귀호출이 가능하다
- 유지보수가 쉽다
- 협업이 가능하다.

shiny module 의 제작 : editData 패키지의 예

- UI 제작 :

```
editableDTUI(id)
```

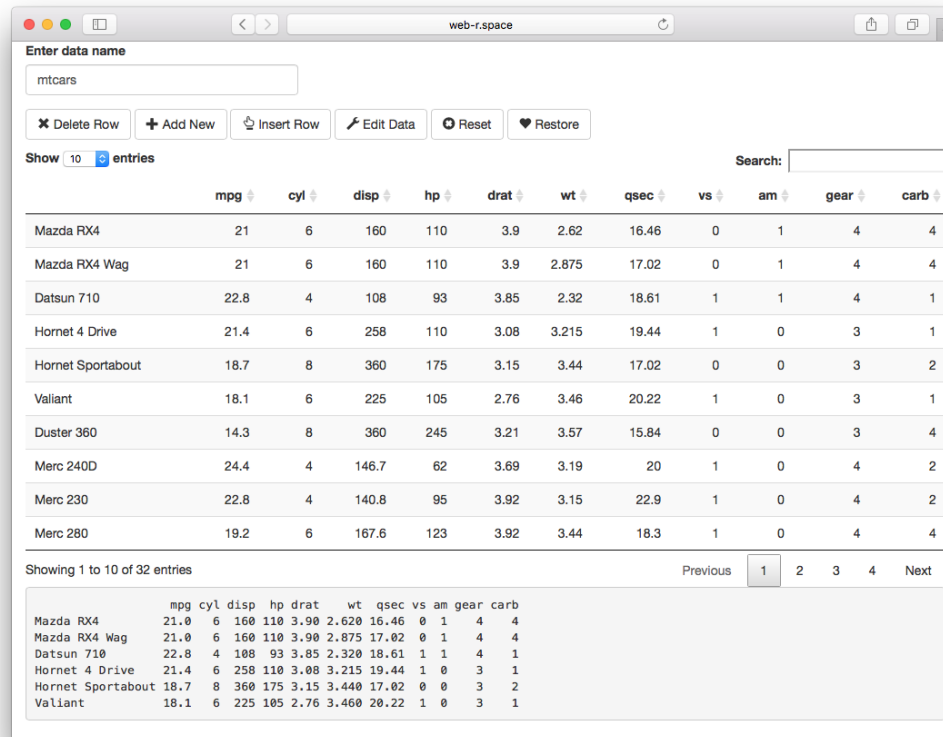
- Server function의 제작

```
editableDT(input, output, session, dataname = reactive(""),  
  data = reactive(NULL), inputwidth = reactive(100), mode=reactive(1)
```


editableDTUI

```
editableDTUI <- function(id){  
  
  ns=NS(id)  
  fluidPage(  
    fluidRow(  
      actionButton(ns("delRow"), "Delete Row", icon=icon("remove", 1), lib="font-awesome"),  
      actionButton(ns("addRow"), "Add New", icon=icon("plus", lib="font-awesome"), lib="font-awesome"),  
      actionButton(ns("insertRow"), "Insert Row", icon=icon("hand-up", lib="font-awesome"), lib="font-awesome"),  
      actionButton(ns("editData"), "Edit Data", icon=icon("wrench", lib="font-awesome"), lib="font-awesome"),  
      actionButton(ns("newCol"), "New Col", icon=icon("plus-sign", lib="font-awesome"), lib="font-awesome"),  
      actionButton(ns("removeCol"), "Remove Col", icon=icon("trash", lib="font-awesome"), lib="font-awesome"),  
      radioButtons3(ns("selection"), "Data Selection", choices=c("select", "none", "all"),  
                    inline=TRUE, labelwidth=130, align="center"),  
      p(""),  
      DT::dataTableOutput(ns("origTable")),  
      conditionalPanel(condition="true==false",  
                        numericInput(ns("width2"), "width2", value=100),  
                        textInput(ns("result"), "result", value=""),  
                        numericInput(ns("no"), "no", value=1))  
    )  
  )  
}
```

Shiny Module 을 이용한 앱 개발



The screenshot shows a web browser window with the URL 'web-r.space'. The application interface includes a text input field for 'Enter data name' containing 'mtcars'. Below this are several action buttons: 'Delete Row', 'Add New', 'Insert Row', 'Edit Data', 'Reset', and 'Restore'. A 'Show' dropdown is set to '10 entries', and a search bar is present. The main content is a table with 11 columns: 'mpg', 'cyl', 'disp', 'hp', 'drat', 'wt', 'qsec', 'vs', 'am', 'gear', and 'carb'. The table lists 10 car models. At the bottom, there is a pagination control showing 'Showing 1 to 10 of 32 entries' and a set of page numbers (1, 2, 3, 4) with 'Previous' and 'Next' links.

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
Mazda RX4	21	6	160	110	3.9	2.62	16.46	0	1	4	4
Mazda RX4 Wag	21	6	160	110	3.9	2.875	17.02	0	1	4	4
Datsun 710	22.8	4	108	93	3.85	2.32	18.61	1	1	4	1
Hornet 4 Drive	21.4	6	258	110	3.08	3.215	19.44	1	0	3	1
Hornet Sportabout	18.7	8	360	175	3.15	3.44	17.02	0	0	3	2
Valiant	18.1	6	225	105	2.76	3.46	20.22	1	0	3	1
Duster 360	14.3	8	360	245	3.21	3.57	15.84	0	0	3	4
Merc 240D	24.4	4	146.7	62	3.69	3.19	20	1	0	4	2
Merc 230	22.8	4	140.8	95	3.92	3.15	22.9	1	0	4	2
Merc 280	19.2	6	167.6	123	3.92	3.44	18.3	1	0	4	4

```
shiny::runGitHub('shinyLecture2', 'cardiomoon', subdir='inst/app23')
```

소스파일

```
library(shiny)
library(editData)

ui <- fluidPage(
  textInput("mydata", "Enter data name", value="mtcars"),
  editableDTUI("table1"),
  verbatimTextOutput("test")
)
server <- function(input, output) {
  df=callModule(editableDT,"table1",dataname=reactive(input$mydata))

  output$test=renderPrint({
    head(df())
  })
}
shinyApp(ui, server)
```

Shiny Module의 장점 : 재사용

- multiple editData() function

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
Mazda RX4	21	6	160	110	3.9	2.62	16.46	0	1	4	4
Mazda RX4 Wag	21	6	160	110	3.9	2.875	17.02	0	1	4	4
Datsun 710	22.8	4	108	93	3.85	2.32	18.61	1	1	4	1
Hornet 4 Drive	21.4	6	258	110	3.08	3.215	19.44	1	0	3	1
Hornet Sportabout	18.7	8	360	175	3.15	3.44	17.02	0	0	3	2
Valiant	18.1	6	225	105	2.76	3.46	20.22	1	0	3	1
Duster 360	14.3	8	360	245	3.21	3.57	15.84	0	0	3	4
Merc 240D	24.4	4	146.7	62	3.69	3.19	20	1	0	4	2
Merc 230	22.8	4	140.8	95	3.92	3.15	22.9	1	0	4	2

```
shiny::runApp(system.file('multipleData', package='editData'))
```

샤이니 모듈의 시작 : uiOutput

지난 시간 만들었던 다중회귀분석 앱을 한국어/영어를 지원하는 앱으로 바꾸어 본다.

Multiple Regression Analysis x Keon-Woong

127.0.0.1:4977

다중회귀분석

select language
 english 한국어

데이터선택
 mtcars
 iris
 acs
 radial

반응변수
mpg

설명변수(들)

다중회귀분석

데이터 테이블 보기

Show 10 entries Search:

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear
Mazda FX4	21	6	160	110	3.9	2.62	16.46	0	1	4
Mazda FX4 Wag	21	6	160	110	3.9	2.875	17.02	0	1	4
Datsun 710	22.8	4	108	93	3.85	2.32	18.61	1	1	4
Hornet 4 Drive	21.4	6	258	110	3.08	3.215	19.44	1	0	3
Hornet Sportabout	18.7	8	360	175	3.15	3.44	17.02	0	0	3
Valliant	18.1	6	225	105	2.76	3.46	20.22	1	0	3
Duster 360	14.3	8	360	245	3.21	3.57	15.84	0	0	3
Merc 240D	24.4	4	146.7	62	3.69	3.19	20	1	0	4
Merc 230	22.8	4	140.8	95	3.92	3.15	22.9	1	0	4
Merc 280	19.2	6	167.6	123	3.92	3.44	18.3	1	0	4

Showing 1 to 10 of 32 entries Previous 1 2 3 4 Next

```
shiny::runGitHub('shinyLecture2', 'cardiomoon', subdir='inst/app20')
```

uiOutput을 샤이니 모듈로

바로 전에 만들었던 다중회귀분석 앱의 uiOutput을 샤이니 모듈로 바꾸어 본다.

shiny module을 사용한 다중회귀분석

언어선택
 영어 한국어

데이터선택
 mtcars
 iris
 acs
 radial

반응변수
mpg

설명변수
cyl disp

다중회귀분석

데이터테이블보기

```
Call:  
lm(formula = mpg ~ cyl + disp, data = mtcars)
```

Residuals:
Min 1Q Median 3Q Max
-4.4213 -2.1722 -0.6362 1.1899 7.0516

Coefficients:
Estimate Std. Error t value Pr(>|t|)
(Intercept) 34.66099 2.54700 13.609 4.02e-14 ***
cyl -1.58728 0.71184 -2.230 0.0337 *
disp -0.02058 0.01026 -2.007 0.0542 .

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.055 on 29 degrees of freedom
Multiple R-squared: 0.7596, Adjusted R-squared: 0.743
F-statistic: 45.81 on 2 and 29 DF, p-value: 1.058e-09

재귀호출 : Recursive Call

예제 1: 계승(factorial)의 계산

$$n! = \prod_{k=1}^n k = n \times (n-1) \times (n-2) \times \dots \times 3 \times 2 \times 1$$

Factorial by iteration

```
factorial=function(n){  
  result=1  
  for(i in n:1) result=result*i  
  result  
}  
  
factorial(5)
```

[1] 120

Factorial by recursion

```
factorial2=function(n){  
  ifelse(n==1,1,n*factorial2(n-1))  
}  
  
factorial2(5)
```

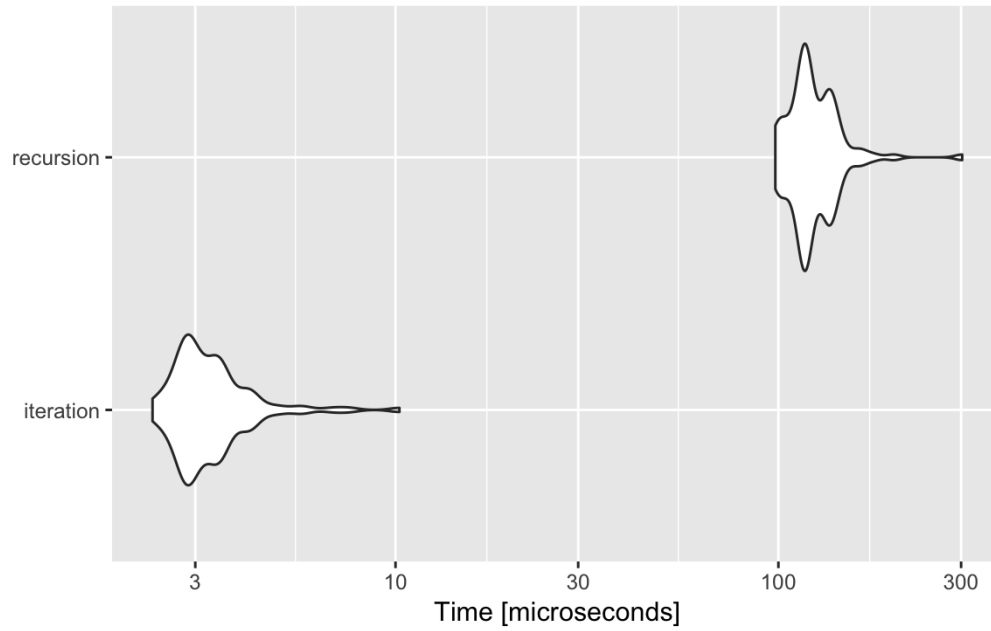
[1] 120

```
library(microbenchmark)
n=50
res=microbenchmark(
  iteration=factorial(n),
  recursion=factorial2(n)
)
res
```

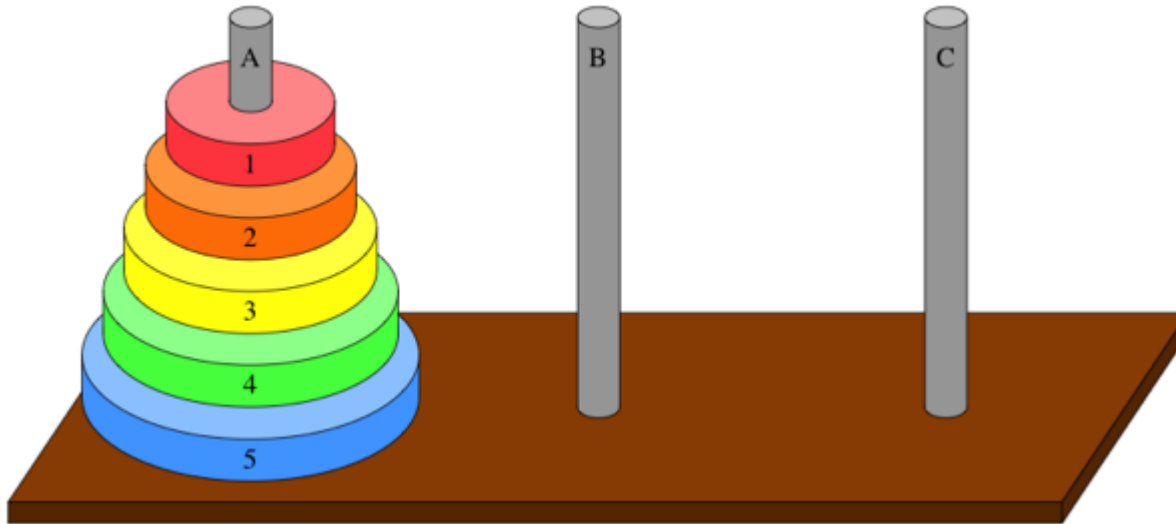
Unit: microseconds

expr	min	lq	mean	median	uq	max	neval	cld
iteration	2.320	2.8245	3.39874	3.0955	3.532	10.238	100	a
recursion	98.177	115.2305	125.28092	118.1075	135.634	302.015	100	b

```
ggplot2::autoplot(res)
```



예제 2: 하노이의 탑



하노이의 탑 https://en.wikipedia.org/wiki/Tower_of_Hanoi

R을 이용한 하노이의 탑 문제 해결

https://github.com/cardiomoon/shinyLecture2/blob/master/tower_of_hanoi.R

tower_of_hanoi.R

```
tower_of_hanoi <- function(n = 7) {  
  ...  
  move.hanoi <- function(k, from, to, via) {  
    if (k > 1) {  
      move.hanoi(k - 1, from, via, to)  
      move.hanoi(1, from, to, via)  
      move.hanoi(k - 1, via, to, from)  
    }  
    else {  
      cat("Move ", tower[[from]][1], " from ", LETTERS[from],  
          " to ", LETTERS[to], "\n")  
      tower[[to]] <- c(tower[[from]][1], tower[[to]])  
      tower[[from]] <- tower[[from]][-1]  
      draw.hanoi()  
      Sys.sleep(0.5)  
    }  
  }  
  draw.hanoi()  
  move.hanoi(n, 1, 2, 3)  
  par(mfrow = c(1, 1))  
}
```

예제 3: 피보나치 수



레오나르도 피보나치(1175-1250)

n번째 달의 토끼의 수 계산

- 첫 달에는 새로 태어난 토끼 한 쌍만이 존재한다.
- 두 달 이상이 된 토끼는 번식 가능하다.
- 번식 가능한 토끼 한 쌍은 매달 새끼 한 쌍을 낳는다.
- 토끼는 죽지 않는다.

n번째 달의 토끼의 수 계산

- 첫달: 1
- 둘째달: 1
- 세째달: 2
- 네째달: 3

- 다섯번째 달: 5

- n번째 달 토끼는 a쌍, n+1번째 달 토끼(새로태어난 토끼 포함)의 수를 b쌍이라고 하면 n+2번째 달의 토끼는 a+b쌍의 토끼가 된다

피보나치의 수를 재귀호출로 풀어보면

```
fibonacci=function(n){  
  ifelse(n==0,0,  
         ifelse(n==1,1,fibonacci(n-1)+fibonacci(n-2)))  
}  
  
sapply(1:20,fibonacci)
```

```
[1] 1 1 2 3 5 8 13 21 34 55 89 144 233 377  
[15] 610 987 1597 2584 4181 6765
```

재귀호출의 문제점

```
fibonacci=function(n){  
  cat("+")  
  ifelse(n==0,0,  
         ifelse(n==1,1,fibonacci(n-1)+fibonacci(n-2)))  
}  
  
fibonacci(10)
```

+++++

[1] 55

재귀호출 횟수?

```
fibonacci=function(n){  
  count<-count+1  
  ifelse(n==0,0,  
         ifelse(n==1,1,fibonacci(n-1)+fibonacci(n-2)))  
}  
  
n<-20  
result<-resultcount<-vector("integer",n)  
for(i in 1:n){  
  count<-0  
  result[i]=fibonacci(i)  
  resultcount[i]=count  
}  
df=data.frame(result=result,count=resultcount)
```

```
z=ztable::ztable(df,type="html",digits=0,caption="Recursion")
z
```

Recursion		
	result	count
1	1	1
2	1	3
3	2	5
4	3	9
5	5	15
6	8	25
7	13	41
8	21	67
9	34	109
10	55	177
11	89	287
12	144	465
13	233	753
14	377	1219
15	610	1973
16	987	3193
17	1597	5167
18	2584	8361
19	4181	13529
20	6765	21891

꼬리 재귀

```
fibonacci2=function(n){
    fibonacciSub(n,0,1)
}

fibonacciSub=function(n,t0,t1){
    ifelse(n==0,t0,
           fibonacciSub(n-1,t1,t0+t1))
}

sapply(1:20,fibonacci2)
```

```
[1] 1 1 2 3 5 8 13 21 34 55 89 144 233 377
[15] 610 987 1597 2584 4181 6765
```

재귀호출 횟수?

```
fibonacciSub=function(n,t0,t1){
  count<-count+1
  ifelse(n==0,t0,
         fibonacciSub(n-1,t1,t0+t1))
}

n<-20
result<-resultcount<-vector("integer",n)
for(i in 1:n){
  count<-0
  result[i]=fibonacci2(i)
  resultcount[i]=count
}
df1=data.frame(result=result,count=resultcount)
```

```
z1=ztable::ztable(df1,type="html",digits=0,caption="Tail Recursion")
z1
```

Tail Recursion		
	result	count
1	1	2
2	1	3
3	2	4
4	3	5
5	5	6
6	8	7
7	13	8
8	21	9
9	34	10
10	55	11
11	89	12
12	144	13
13	233	14
14	377	15
15	610	16
16	987	17
17	1597	18
18	2584	19
19	4181	20
20	6765	21

```
ztable::parallelTables(c(0.5,0.5),listTables=list(z,z1),type="html")
```

Recursion		
	result	count
1	1	1
2	1	3
3	2	5
4	3	9
5	5	15
6	8	25
7	13	41
8	21	67
9	34	109
10	55	177
11	89	287
12	144	465
13	233	753
14	377	1219
15	610	1973
16	987	3193
17	1597	5167
18	2584	8361
19	4181	13529
20	6765	21891

Tail Recursion		
	result	count
1	1	2
2	1	3
3	2	4
4	3	5
5	5	6
6	8	7
7	13	8
8	21	9
9	34	10
10	55	11
11	89	12
12	144	13
13	233	14
14	377	15
15	610	16
16	987	17
17	1597	18
18	2584	19
19	4181	20
20	6765	21

반복문

```
fibonacci=function(n){
  before=0
  current=1
  ifelse(n==0,0,
        ifelse(n==1,1,
              {
                for(i in 1:(n-1)){
                  temp=current
                  current=before+current
                  before=temp
                }
                current}))
}
sapply(1:20, fibonacci)
```

```
[1] 1 1 2 3 5 8 13 21 34 55 89 144 233 377
[15] 610 987 1597 2584 4181 6765
```

비네(Binet)의 피보나치 수 공식

$$u_n = u_{n-1} + u_{n-2}, n > 1$$

$$u_0 = 0$$

$$u_1 = 1$$

$$u_n = \frac{(1 + \sqrt{5})^n - (1 - \sqrt{5})^n}{2^n \sqrt{5}}$$

```
fiboBinet=function(n){  
  ((1+sqrt(5))^n-(1-sqrt(5))^n)/(2^n*sqrt(5))  
}  
sapply(1:20, fiboBinet)
```

```
[1] 1 1 2 3 5 8 13 21 34 55 89 144 233 377  
[15] 610 987 1597 2584 4181 6765
```

속도 측정

```
library(microbenchmark)
n=20
res=microbenchmark(
  recursion=fibonacci(n),
  resursiontail=fibonacci2(n),
  loop=fiboLoop(n),
  Binet=fiboBinet(n),
  times=10
)
res
```

Unit: nanoseconds

expr	min	lq	mean	median	uq	max
recursion	77267289	80711073	86613226.9	85183641.5	91144803	103925499
resursiontail	52692	53130	64177.4	60915.0	73705	92238
loop	8456	11283	12298.4	12635.5	14171	14666
Binet	626	755	2612.6	1028.0	5318	5392

neval cld

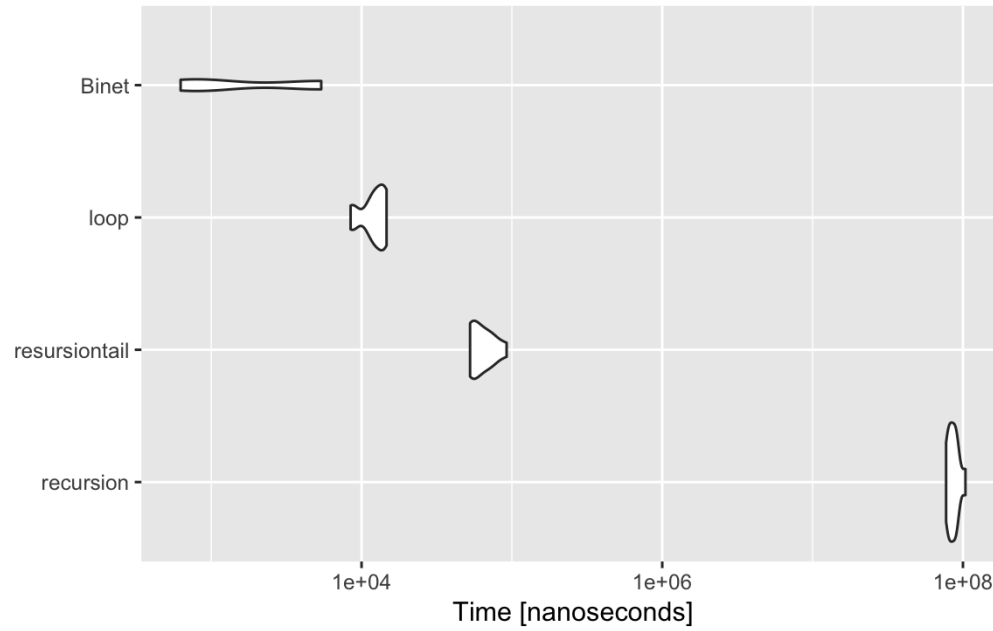
10 b

10 a

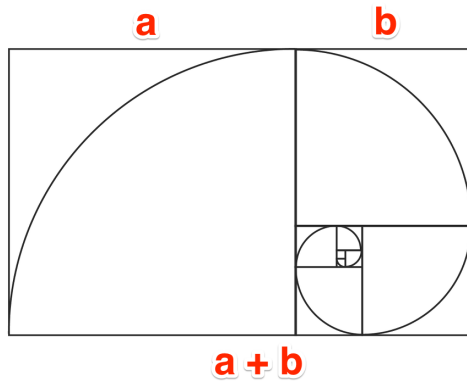
10 a

10 a

```
ggplot2::autoplot(res)
```



피보나치수와 황금비



$$\frac{a+b}{a} = \frac{a}{b} = \varphi$$

$$\left(\frac{a}{b}\right)^2 = \left(\frac{a}{b}\right) \times \left(\frac{a+b}{a}\right) = \frac{a}{b} + 1$$

$$\varphi^2 = \varphi + 1$$

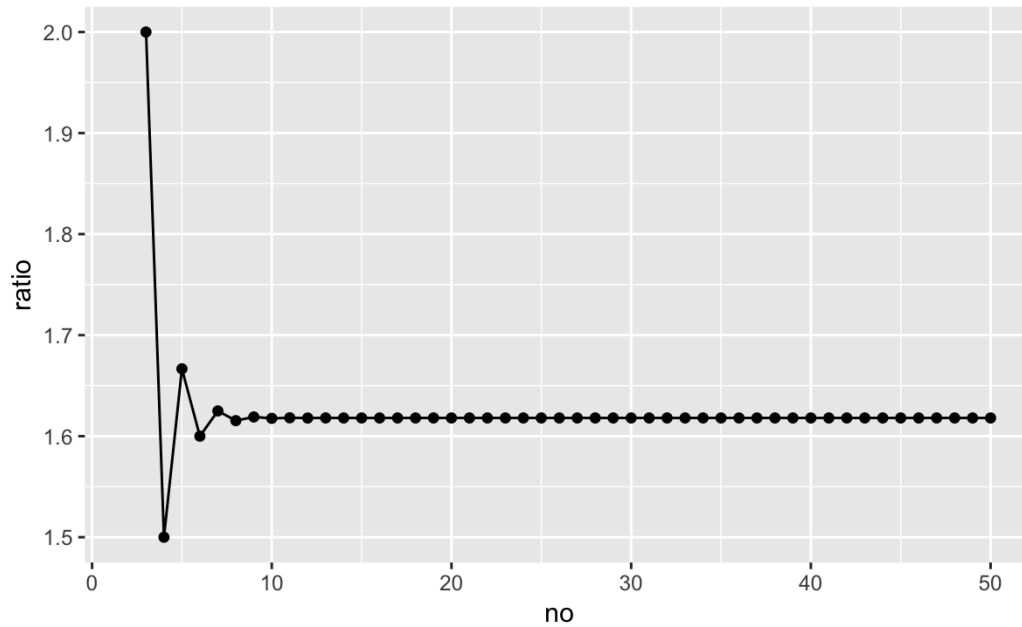
$$\varphi = \frac{1 \pm \sqrt{5}}{2} \approx 1.618$$

```
result=sapply(2:50, fiboBinet)
result
```

```
[1]          1          2          3          5          8
[6]         13         21         34         55         89
[11]        144        233        377        610        987
[16]       1597       2584       4181       6765       10946
[21]      17711      28657      46368      75025      121393
[26]     196418     317811     514229     832040     1346269
[31]    2178309    3524578    5702887    9227465    14930352
[36]   24157817   39088169   63245986  102334155  165580141
[41]  267914296  433494437  701408733 1134903170 1836311903
[46] 2971215073 4807526976 7778742049 12586269025
```

```
ratio=result/dplyr::lag(result)
df=data.frame(no=2:50,result,ratio)
```

```
require(ggplot2)
ggplot(data=df, aes(x=no, y=ratio))+geom_point()+geom_line()
```



ggplotAssist 개발시 문제점

```
?ggplot2::geom_point
```

비교적 간단한 함수의 경우 shiny app을 통한 구현이 쉽다.

```
geom_point(mapping = NULL, data = NULL, stat = "identity",  
            position = "identity", ..., na.rm = FALSE, show.legend = NA,  
            inherit.aes = TRUE)
```

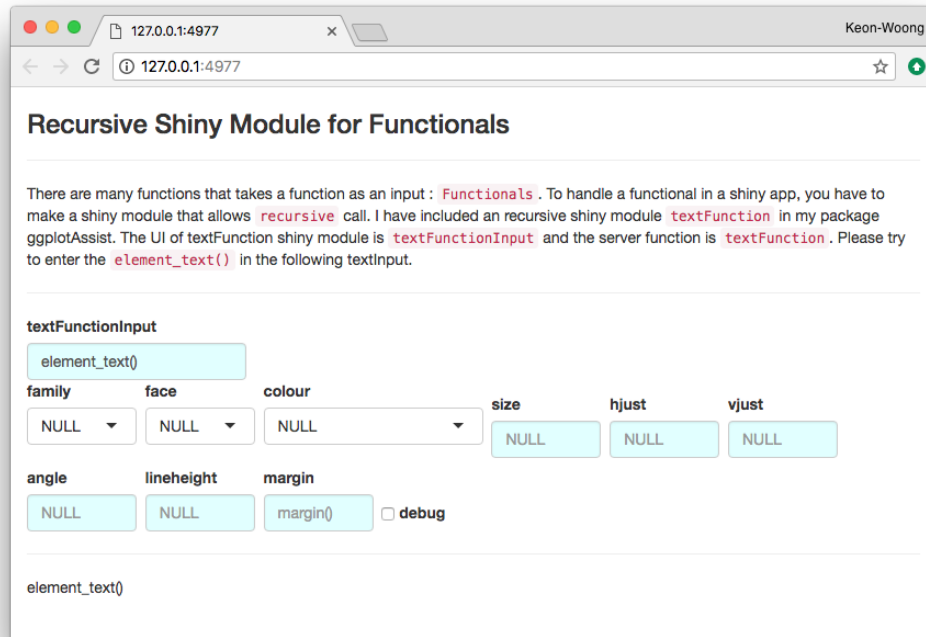

ggplotAssist 개발시 문제점

```
?ggplot2::guide_colourbar
```

어떤 함수는 함수의 인수로 함수가 들어가고 또 그 함수의 인수로 함수가 들어간다.

```
guide_colourbar(title = waiver(), title.position = NULL,  
  title.theme = NULL, title.hjust = NULL, title.vjust = NULL,  
  label = TRUE, label.position = NULL, label.theme = NULL,  
  label.hjust = NULL, label.vjust = NULL, barwidth = NULL,  
  barheight = NULL, nbin = 20, raster = TRUE, ticks = TRUE,  
  draw.ulim = TRUE, draw.llim = TRUE, direction = NULL,  
  default.unit = "line", reverse = FALSE, order = 0, ...)
```

해결방법: Recursive Shiny Module(1)



```
shiny::runApp(system.file('textFunctionExample',  
                           package='ggplotAssist'))
```

해결방법: Recursive Shiny Module(2)

Recursive Shiny Module for Functionals

There are many functions that takes a function as an input : `Functionals` . To handle a functional in a shiny app, you have to make a shiny module that allows `recursive` call. I have included an recursive shiny module `textFunction` in my package `ggplotAssist`. The UI of `textFunction` shiny module is `textFunctionInput` and the server function is `textFunction` . Please try to select one of the functions in the following selectInput.

Select function

`guide_colorbar()`

title	title.position	title.theme	title.hjust	title.vjust	label.position
<code>waiver()</code>		<code>element_text()</code>		<input checked="" type="checkbox"/> label	
label.theme	label.hjust	label.vjust	direction	default.unit	order
<code>element_text()</code>				line <input type="checkbox"/> reverse	0
barwidth	barheight	nbin	<input checked="" type="checkbox"/> raster	<input checked="" type="checkbox"/> ticks	<input checked="" type="checkbox"/> draw.ulim
A numeric or a unit() object	A numeric or a unit() object	20	<input checked="" type="checkbox"/> draw.llim		

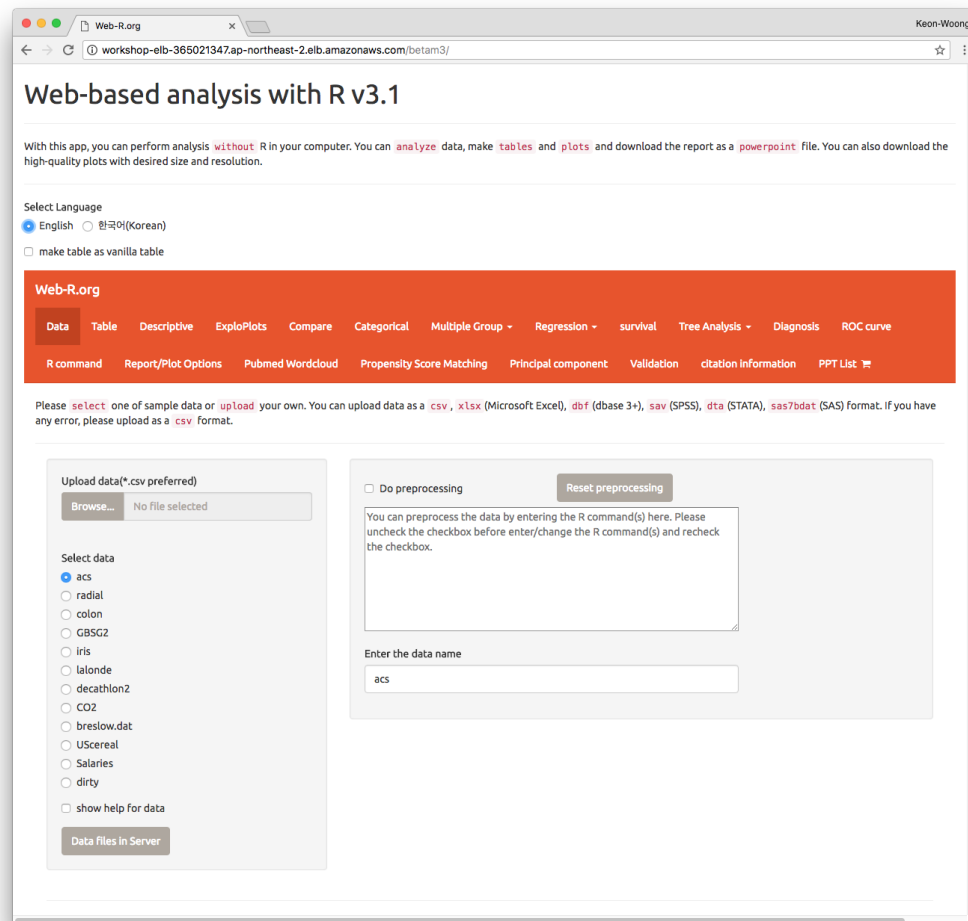
`guide_colorbar()`

```
shiny::runApp(system.file('textFunctionExample2',  
                           package='ggplotAssist'))
```

자동차 생산의 모듈화

자동차 생산 모듈화 - 네이버 블로그

"웹에서 하는 R 통계분석" 개발시 문제점



샤이니 모듈: pptxList

The screenshot shows a web browser window with the URL 127.0.0.1:3926. The page title is "Keon-Woong". The main content area is titled "Saved PowerPoint List" and contains a table with 5 entries. The table has columns for "type", "title", and "code". Below the table are navigation buttons for "Delete Row", "Add New", "Insert Row", "Edit Data", "New Col", "Remove Col", and "Manipulate". There is also a "Show" dropdown set to "10 entries" and a "Search:" input field. At the bottom of the table, it says "Showing 1 to 5 of 5 entries" with "Previous" and "Next" buttons. Below the table are three sections: "Upload PPTList(*.csv)" with a "Browse..." button and "No file selected" text; "Load sampleData" with a "load sampleData" button; and "Reset PPT List" with a "reset PPT List" button. At the bottom, there are download options: "download as csv", "download as HTML", "download as PDF", "download as PPTx", and "download as Word". There is also a "download Plots" section with "Resolution" (300), "units" (in), "plotWidth" (7), and "plotHeight" (5) fields, and a "Format As" section with radio buttons for "PNG", "SVG", and "PDF".

show preprocessing

Saved PowerPoint List

Delete Row Add New Insert Row Edit Data New Col Remove Col Manipulate

Show 10 entries Search:

	type	title	code
1	mytable	mytable	mytable(sex~.,data=acs)
2	table	iris[1:10,]	df2FlexTable(iris[1:10,])
3	plot	plot	plot(iris)
4	ggplot	ggplot	ggplot(iris,aes(x=Sepal.Length,y=Sepal.Width,color=Species))+geom_point()
5	Rcode	Regression Analysis	summary(mtcars) fit=lm(mpg~wt*hp,data=mtcars) summary(fit)

Showing 1 to 5 of 5 entries Previous 1 Next

Upload PPTList(*.csv)

Browse... No file selected

Load sampleData

load sampleData

Reset PPT List

reset PPT List

download as csv download as HTML download as PDF download as PPTx download as Word

download Plots Resolution 300 units in plotWidth 7 plotHeight 5 Format As PNG SVG PDF

샤이니 모듈: pptxList 소스파일

```
library(shiny)
library(rrtable)
library(ggplot2)
library(webr)
library(moonBook)

ui=fluidPage(
  pptxListInput("pptxlist")
)
server=function(input,output,session){
  mydf<-callModule(pptxList,"pptxlist")
}
shinyApp(ui,server)
```

web-R.org

웹R에서 사이니업을 공동개발할 개발팀/학술팀을 모집합니다.

The screenshot shows the homepage of web-R.org. The main heading is "웹에서 하는 R 통계" (R Statistics on the Web). Below the heading is a navigation menu with items: Home, 웹에서 하는 R 통계, 책 게시판, 기사판, Servers, and 정회원매뉴. The main content area features a large welcome message: "Web-R.org 에 오신 것을 환영합니다." (Welcome to Web-R.org). The message explains the site's purpose: to help researchers use R for statistical analysis through a web interface and a Shiny app. It also mentions a recruitment drive for members, with a 500,000 KRW membership fee and a 100,000 KRW annual fee for students. A sidebar on the right features a book cover titled "R 통계와 그래프" (R Statistics and Graphs for Medical Purposes). At the bottom, there are several statistics and lists:

cardiomoan 로그아웃	현재 접속중	접속통계	글작성	댓글작성	이번주 조회수
cardiomoan 로그아웃 최근 로그인: 2018-03-18 <ul style="list-style-type: none"> 회원정보 보기 스킨 설정 보기 좌장원 보기 작성 글 보기 친구 보기 공지사항 보기 관리 	cardiomoan	오늘: 246 어제: 350 전체: 224,162	1. cardiomoan ³ 2. 통계장님 ² 3. 스투언트 ¹ 4. schjoo ¹ 5. suein ¹ 6. 김가욱 ¹ 7. 설승환 ¹	1. cardiomoan ³⁰ 2. 박민영 ¹ 3. 스투언트 ¹ 4. suein ¹ 2018-03-12 ~ 2018-03-18	1. 스투언트 ¹⁹ 2. cardiomoan ⁹ 3. suein ⁹ 4. 통계장님 ⁶ 5. schjoo ⁵ 6. 김가욱 ² 7. 설승환 ²