

# 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

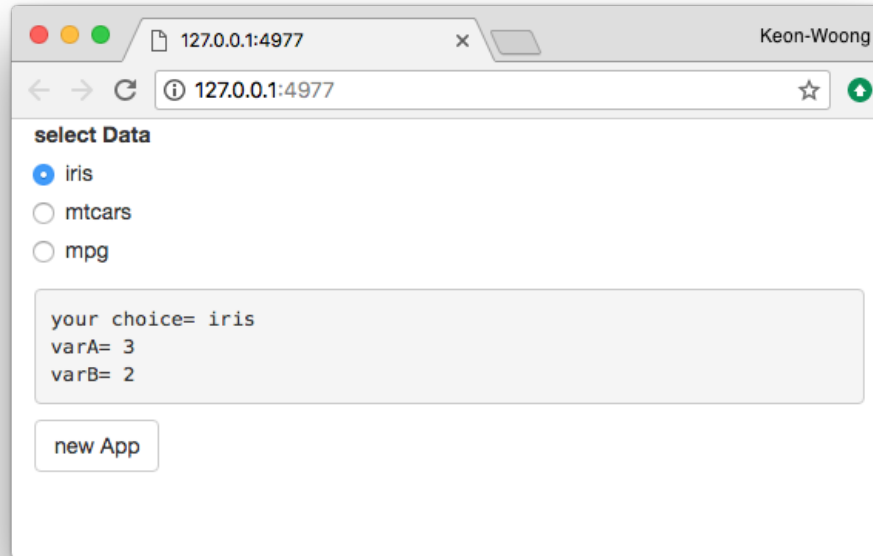
```
varA <- 1
varB <- 1

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

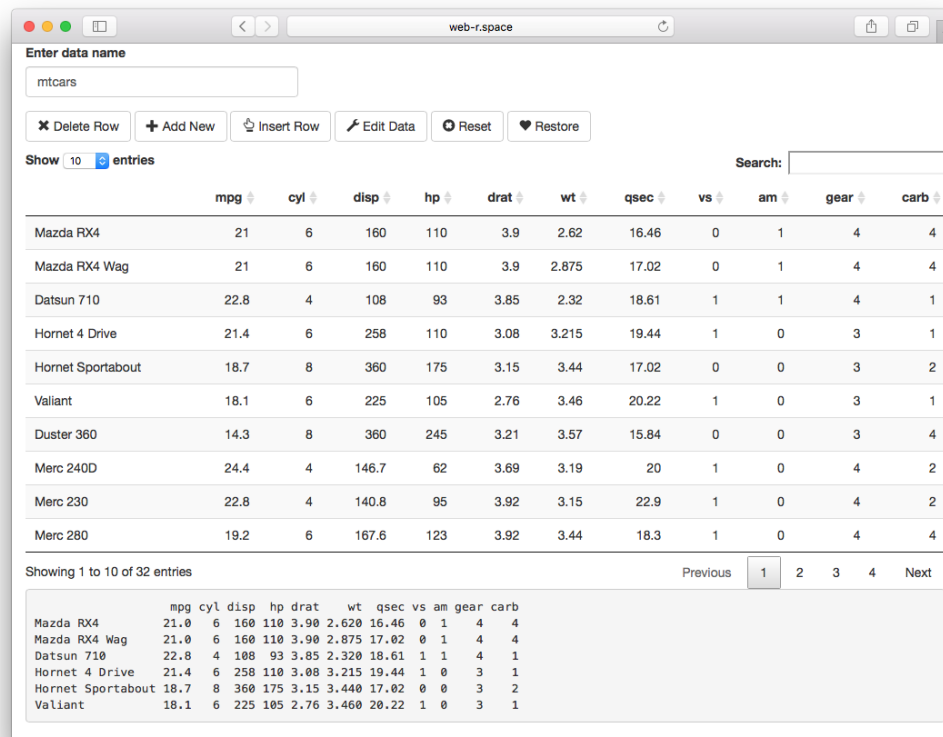
```
editableDT(input, output, session, dataname = reactive(""),  
  data = reactive(NULL), inputwidth = reactive(100), mode=reactive(1)
```

# editableDTUI

```
editableDTUI <- function(id){  
  
  ns=NS(id)  
  fluidPage(  
    fluidRow(  
      actionButton(ns("delRow"),"Delete Row",icon=icon("remove",1  
      actionButton(ns("addRow"),"Add New",icon=icon("plus",lib="{  
      actionButton(ns("insertRow"),"Insert Row",icon=icon("hand-u  
      actionButton(ns("editData"),"Edit Data",icon=icon("wrench":  
      actionButton(ns("newCol"),"New Col",icon=icon("plus-sign",1  
      actionButton(ns("removeCol"),"Remove Col",icon=icon("trash'  
      radioButtons3(ns("selection"),"Data Selection",choices=c("s  
        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 address bar displaying 'web-r.space'. The application interface includes a text input field labeled 'Enter data name' containing 'mtcars'. Below this are buttons for 'Delete Row', 'Add New', 'Insert Row', 'Edit Data', 'Reset', and 'Restore'. A 'Show' dropdown is set to '10 entries', and a 'Search:' field is present. The main data table lists car models with columns: mpg, cyl, disp, hp, drat, wt, qsec, vs, am, gear, carb. The first 10 rows are visible. At the bottom, a pagination bar shows '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

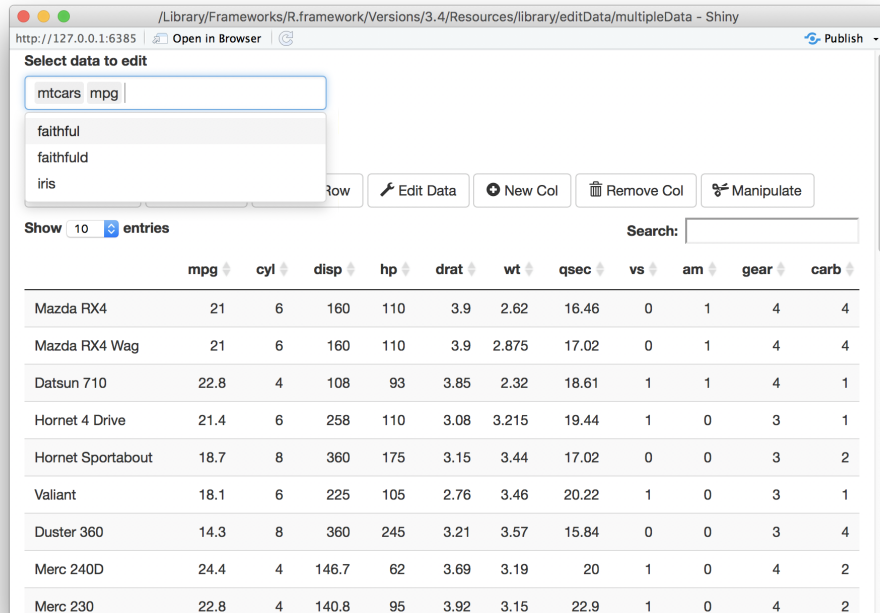
```
library(shiny)
library(editData)

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

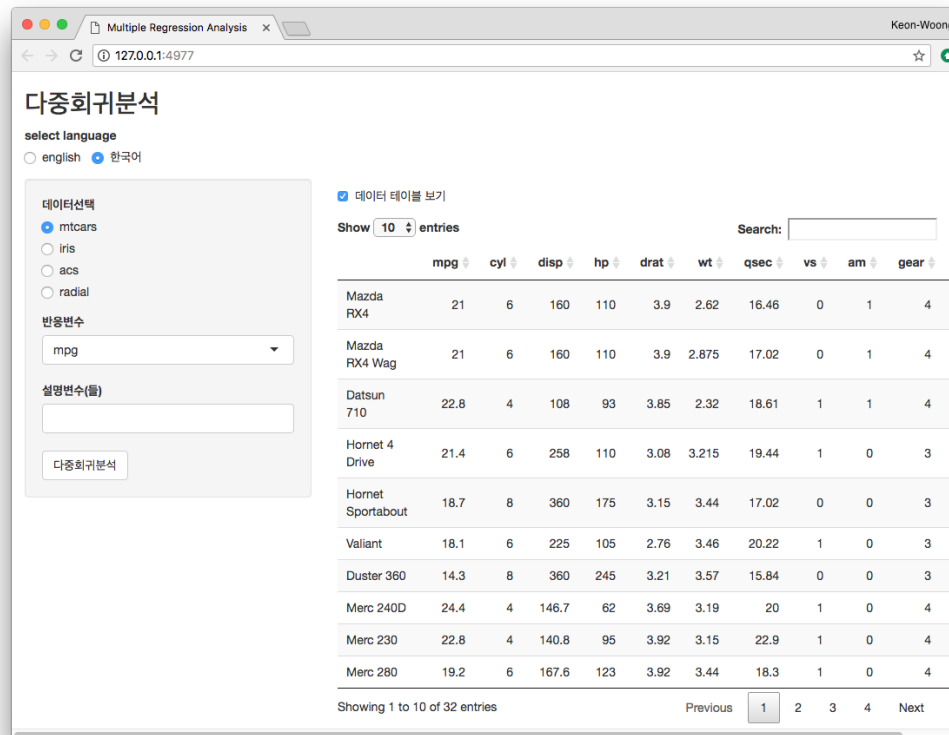
- multiple editData() function



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

# 샤이니 모듈의 시작 : uiOutput

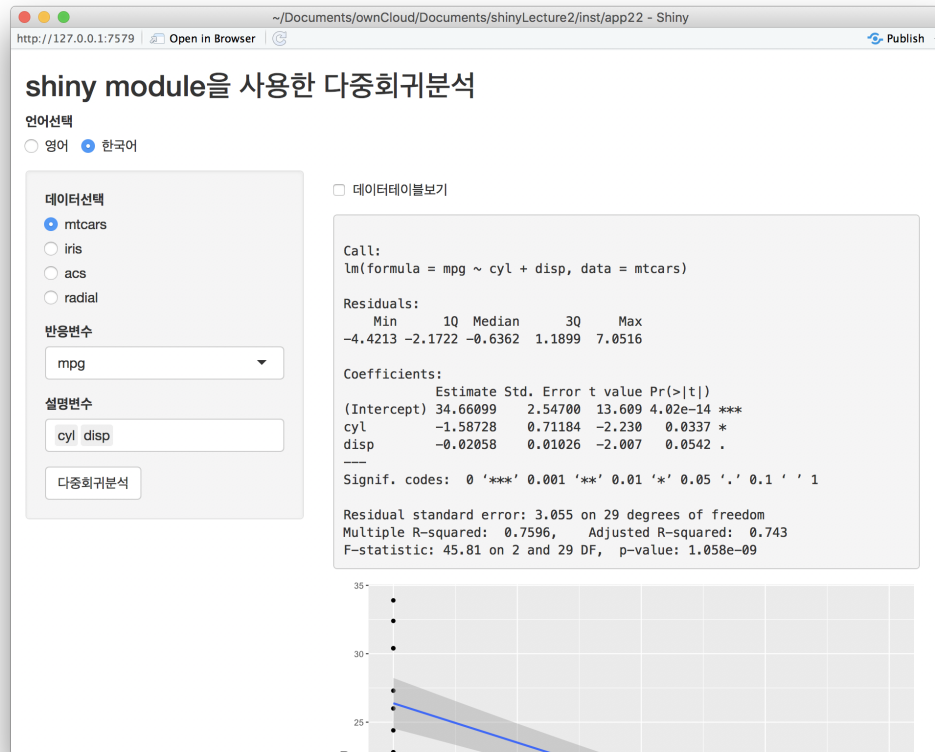
지난 시간 만들었던 다중회귀분석 앱을 한국어/영어를 지원하는 앱으로 바꾸어 본다.



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

# uiOutput을 샤이니 모듈로

바로 전에 만들었던 다중회귀분석 앱의 uiOutput을 샤이니 모듈로 바꾸어 본다.



## 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)
```

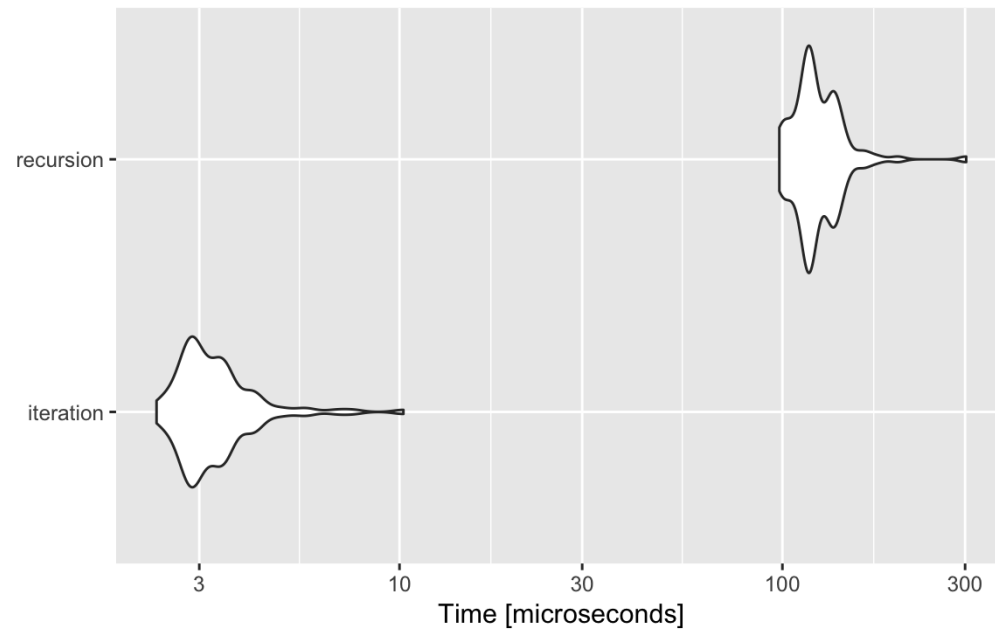
```
[1] 120
```

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

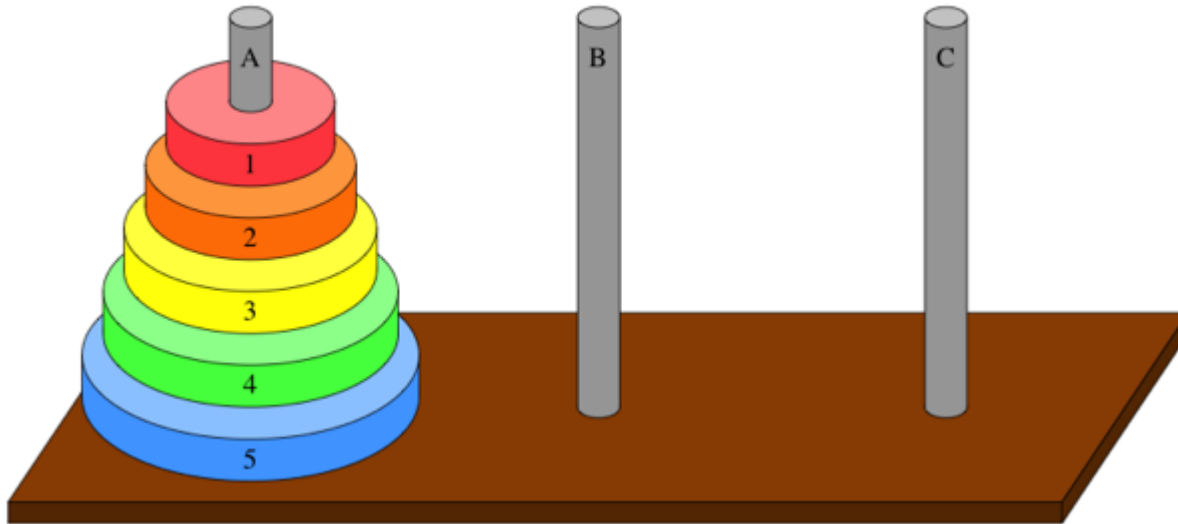
Unit: microseconds

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](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](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)

$n$ 번째 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
```

# 속도측정

```
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

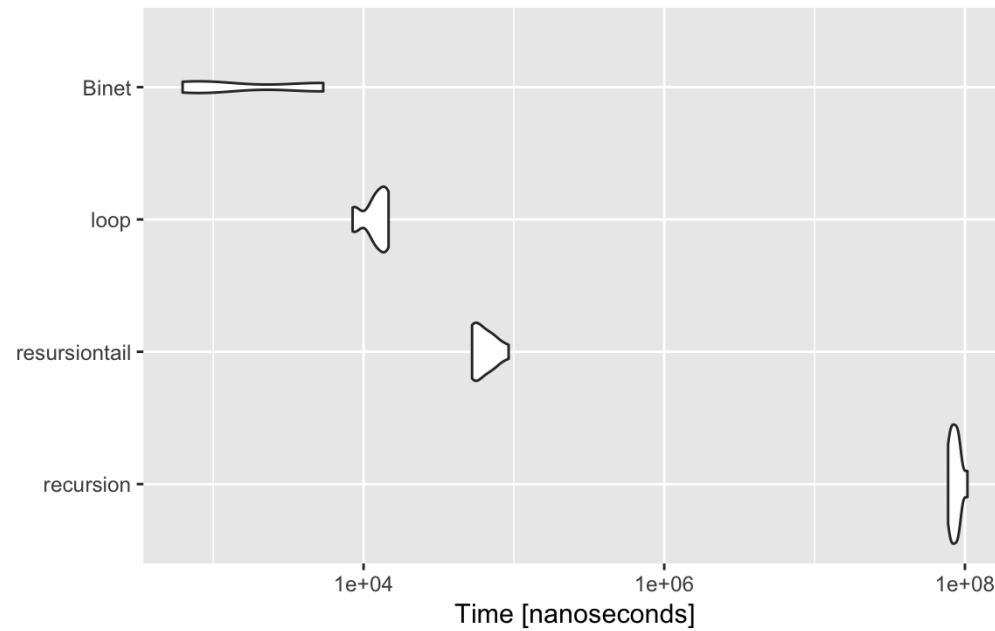
10 b

10 a

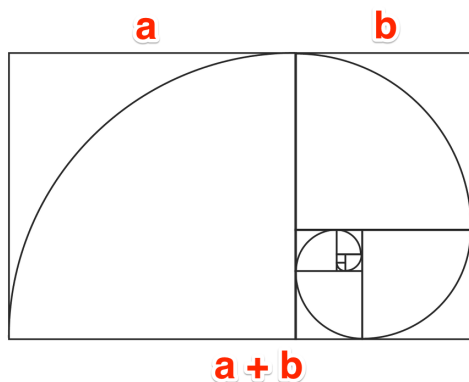
10 a

10 a

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



# 피보나치수와 황금비

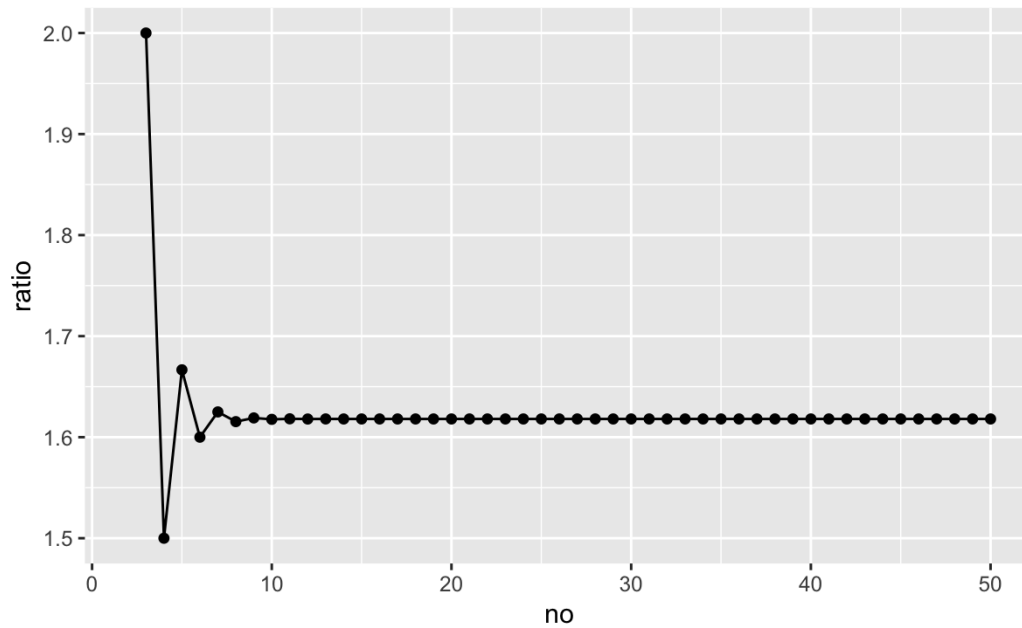


```
result=sapply(2:50,fiboBinet)
result
```

[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)

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 enter the `element_text()` in the following textInput.

**textFunctionInput**

`element_text()`

**family** **face** **colour** **size** **hjust** **vjust**

`NULL` `NULL` `NULL` `NULL` `NULL` `NULL`

**angle** **lineheight** **margin** ☐ **debug**

`NULL` `NULL` `margin()`

`element_text()`

```
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: `walver()`, title.position: `element_text()`, title.theme: `element_text()`, title.hjust: `label`, title.vjust: `label`, label.position: `label`

label.theme: `element_text()`, label.hjust: `label`, label.vjust: `label`, direction: `line`, default.unit: `reverse`, order: `0`

barwidth: `A numeric or a unit() object`, barheight: `A numeric or a unit() object`, nbins: `20`, raster: `checkbox`, ticks: `checkbox`, draw.ulim: `checkbox`

`guide_colorbar()`

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

# 자동차 생산의 모듈화

자동차 생산 모듈화 - 네이버 블로그

# "웹에서 하는 R 통계분석" 개발시 문제점

The screenshot shows the Web-R.org web application interface. At the top, the title "Web-based analysis with R v3.1" is displayed. Below the title, a paragraph explains the app's functionality: "With this app, you can perform analysis without R in your computer. You can analyze data, make tables and plots and download the report as a powerpoint file. You can also download the high-quality plots with desired size and resolution." Below this, there is a "Select Language" section with radio buttons for "English" (selected) and "한국어(Korean)". A checkbox for "make table as vanilla table" is also present. A red navigation bar contains various analysis categories: Data, Table, Descriptive, ExploPlots, Compare, Categorical, Multiple Group, Regression, survival, Tree Analysis, Diagnosis, and ROC curve. Below the navigation bar, a list of analysis types is shown: R command, Report/Plot Options, Pubmed Wordcloud, Propensity Score Matching, Principal component, Validation, citation Information, and PPT List. A paragraph instructs users to "Please select one of sample data or upload your own. You can upload data as a csv, xls (Microsoft Excel), dbf (dbase 3+), sav (SPSS), dta (STATA), sas7bdat (SAS) format. If you have any error, please upload as a csv format." The main content area is divided into two sections. The left section, titled "Upload data(\*.csv preferred)", includes a "Browse..." button, a "No file selected" status, a "Select data" list with radio buttons (acs is selected), and a "Data files in Server" button. The right section, titled "Do preprocessing", includes a "Reset preprocessing" button, a text box for entering R commands, and a text input field for "Enter the data name" with "acs" entered.

Web-R.org

Data Table Descriptive ExploPlots Compare Categorical Multiple Group Regression survival Tree Analysis Diagnosis ROC curve

R command Report/Plot Options Pubmed Wordcloud Propensity Score Matching Principal component Validation citation Information PPT List

Please select one of sample data or upload your own. You can upload data as a csv, xls (Microsoft Excel), dbf (dbase 3+), sav (SPSS), dta (STATA), sas7bdat (SAS) format. If you have any error, please upload as a csv format.

Upload data(\*.csv preferred)

Browse... No file selected

Select data

☒ acs

☐ radial

☐ colon

☐ GBSG2

☐ iris

☐ lalonde

☐ decathlon2

☐ CO2

☐ breslow.dat

☐ UScereal

☐ Salaries

☐ dirty

☐ show help for data

Data files in Server

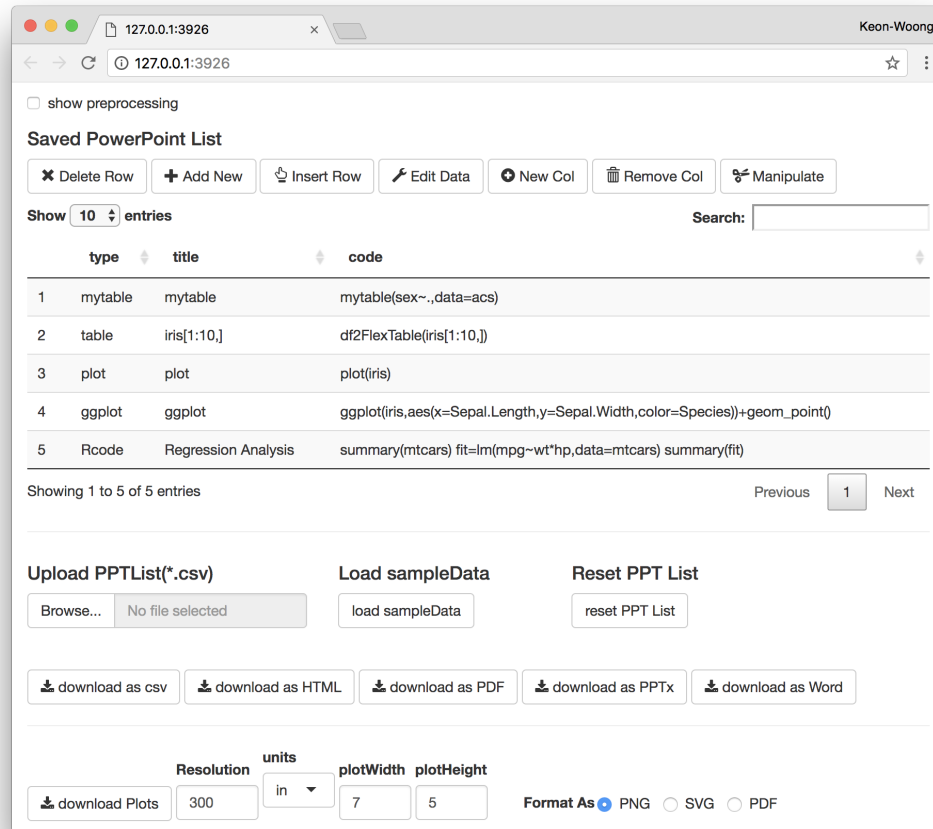
☐ Do preprocessing Reset preprocessing

You can preprocess the data by entering the R command(s) here. Please uncheck the checkbox before enter/change the R command(s) and recheck the checkbox.

Enter the data name

acs

# 샤이니 모듈: pptxList



# 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 샤이니업을 공동개발할 개발팀/학술팀을 모집합니다.

The screenshot shows the web-R.org website. The main heading is "웹에서 하는 R 통계". Below it, there's a navigation bar with links: Home, 웹에서 하는 R 통계, 책 게시판, 게시판, Servers, and 정회원매뉴. The main content area features a large welcome message: "Web-R.org 에 오신 것을 환영합니다." followed by a paragraph about the site's purpose and a call to action for new members. To the right of the welcome message is a book cover titled "R 통계와 그래프". Below the main content, there's a section titled "cardiomoon 로그아웃" with a list of links: 회원정보 보기, 스터디 보기, 장래회 보기, 작성 글 보기, 친구 보기, 관리자 보기, and 관리. To the right of this is a section titled "현재 접속중" with a list of members: cardiomoon. Below this is a section titled "집속통계" with a list of members: cardiomoon, 통계장님, 스튜던트, schjoo, suein, 김기욱, and 설승환. To the right of this is a section titled "글작성" with a list of members: cardiomoon, 박만영, 스튜던트, and suein. To the right of this is a section titled "댓글작성" with a list of members: cardiomoon, 박만영, 스튜던트, and suein. To the right of this is a section titled "이번주 조회수" with a list of members: 스튜던트, cardiomoon, suein, 통계장님, schjoo, 김기욱, and 설승환.

웹에서 하는 R 통계

cardiomoon 관리 로그아웃 검색

Home 웹에서 하는 R 통계 책 게시판 게시판 Servers 정회원매뉴

Web-R.org 에 오신 것을 환영합니다.

"웹에서 하는 R 통계"는 통계에는 관심이 있으나 R을 어려워하는 여러 연구자들을 위한 프로젝트입니다. R설치없이 클릭만으로 웹에 있는 서버를 이용하여 통계분석을 하고 보다 R을 쉽게 사용하기 위한 패키지 개발 및 Shiny app 공동개발을 목표로 하고 있습니다. R을 사랑하는 여러분의 많은 참여를 기다립니다.

현재는 회원가입만 하시면 준회원으로 일주일간 모든 서비스를 이용하실 수 있으나 추후 서비스가 제한될수 있습니다.

서버운영을 위해 일정금액을 후원금으로 보내주시면 정회원으로 등급시켜드립니다. 후원금은 편의상 5만원/1년으로 정했으며 학생은(대학생까지) 1만원/1년으로 해드립니다. 공자사향의 정회원이 되는 방법을 참조하시기 바랍니다.

새로 가입하신 분들은 가입인사를 남겨주세요.

R 통계와 그래프

의학논문 작성을 위한 R 통계와 그래프

cardiomoon 로그아웃

최근 로그인: 2018-03-18

- 회원정보 보기
- 스터디 보기
- 장래회 보기
- 작성 글 보기
- 친구 보기
- 관리자 보기
- 관리

현재 접속중

- cardiomoon

집속통계

오늘: 246  
어제: 350  
전체: 224,162

글작성

- cardiomoon<sup>3</sup>
- 통계장님<sup>2</sup>
- 스튜던트<sup>1</sup>
- schjoo<sup>1</sup>
- suein<sup>1</sup>
- 김기욱<sup>1</sup>
- 설승환<sup>1</sup>

2018-03-12 ~ 2018-03-18

댓글작성

- cardiomoon<sup>30</sup>
- 박만영<sup>1</sup>
- 스튜던트<sup>1</sup>
- suein<sup>1</sup>

2018-03-12 ~ 2018-03-18

이번주 조회수

- 스튜던트<sup>19</sup>
- cardiomoon<sup>9</sup>
- suein<sup>9</sup>
- 통계장님<sup>6</sup>
- schjoo<sup>5</sup>
- 김기욱<sup>2</sup>
- 설승환<sup>2</sup>

2018-03-12 ~ 2018-03-18