Task	R	SAS
	Setting working directory before start!	

Working directory 설정 및 변경 ➔ `setwd("C:/KHE")`
(또는 상단의 메뉴 바에서 수동으로 [디렉토리 변경])
Working directory 위치　　　　➔ `getwd()`

Library 설정 ➔ libname KHE "C:₩KHE";

Library 제거 ➔ libname KHE;

R의 Data frame

- SAS 데이터 저장 구조인 table에 상응하는 R의 데이터 container
- 각 열은 서로 다른 속성(character, numeric, factor, logical, etc…)을 가짐

SAS의 table structure

- 행과 열이 직사각형으로 구성되어 있는 테이블 형태
- 각 열은 character 또는 numeric 속성을 가짐

Setting Before Start

id	initial	gender	age	company	ruser	interest
RC001	PKW	M	27	KTH	Y	1
RC002	KYG	M	30	그루터	Y	2
RC003	KYK	F	26	서울대	Y	1
RC004	KSG	F	28	삼성전자	N	3
RC005	PSH	M	32	Daum	Y	2
RC006	KHW	M	35	KAIST	N	3
RC007	KSW	M	28	Nexon	Y	3
RC008	KJS	M	32	Nexon	Y	1
RC009	PJY	F	28	코난 Tech	N	2
RC010	KCW	M	31	Geference	N	4

<예제 데이터: attendee1>
R User Conference 2011에 선 등록한 10명의 데이터

　　　　　　　　　　　　　　Prepared by Kwak HeeEun

Task	R	SAS
| **Data Import (1)** | ```
> a1=read.table(file="attendee1.csv",header=T,sep=",",
+ colClasses=c("character","character","factor",
+ "numeric","factor","factor","factor"))
> a1
      id initial gender age    company ruser interest
1  RC001     PKW      M  27        KTH     Y        1
2  RC002     KYG      M  30      그루터     Y        2
3  RC003     KYK      F  26       서울대     Y        1
4  RC004     KSG      F  28      삼성전자     N        3
5  RC005     PSH      M  32       Daum     Y        2
6  RC006     KHW      M  35      KAIST     N        3
7  RC007     KSW      M  28      Nexon     Y        3
8  RC008     KJS      M  32      Nexon     Y        1
9  RC009     PJY      F  28   코난 Tech     N        2
10 RC010     KCW      M  31  Geference     N        4
```<br>**<attendee1의 R코드와 R데이터>**<br><br>**read.table()의 arguments**<br><br>file　　　➔ 저장된 데이터명.포맷<br>header　➔ 변수명 유무 TRUE/FAULSE<br>sep　　　➔ separation (delimiter) 구분자<br>colClasses ➔ 변수의 형태<br>　　　　(numeric, character, factor, logical, integer) | ```
libname KHE "C:\KHE";

proc import out=khe.a1
     datafile="C:\KHE\attendee1.csv"
     dbms=csv;
  getnames=yes;
   datarow=2;
run;
```<br><br>| | id | initial | gender | age | company | ruser | interest |<br>|---|---|---|---|---|---|---|---|<br>| 1 | RC001 | PKW | M | 27 | KTH | Y | 1 |<br>| 2 | RC002 | KYG | M | 30 | 그루터 | Y | 2 |<br>| 3 | RC003 | KYK | F | 26 | 서울대 | Y | 1 |<br>| 4 | RC004 | KSG | F | 28 | 삼성전자 | N | 3 |<br>| 5 | RC005 | PSH | M | 32 | Daum | Y | 2 |<br>| 6 | RC006 | KHW | M | 35 | KAIST | N | 3 |<br>| 7 | RC007 | KSW | M | 28 | Nexon | Y | 3 |<br>| 8 | RC008 | KJS | M | 32 | Nexon | Y | 1 |<br>| 9 | RC009 | PJY | F | 28 | 코난 Tech | N | 2 |<br>| 10 | RC010 | KCW | M | 31 | Geference | N | 4 |<br><br>**<attendee1의 SAS코드와 데이터>** |

Task	R	SAS
	Reading a data within a program	

method 1>

```
> a1=data.frame(id=c("RC001","RC002","RC003"),
+ initial=c("PKW","KYG","KYK"), age=c(27,30,26),
+ company=c("KTH","그루터","서울대"),
+ ruser=c("Y","Y","Y"),
+ interest=c(1,2,1))
> a1
      id initial age company ruser interest
1 RC001     PKW  27     KTH     Y        1
2 RC002     KYG  30   그루터     Y        2
3 RC003     KYK  26   서울대     Y        1
```

Data Import (2)

method 2>

```
> a1 <- read.csv( stdin() )
0: id,initial,gender,age,company,ruser,interest
1: RC001,PKW,M,27,KTH,Y,1
2: RC002,KYG,M,30,그루터,Y,2
3:
> a1
      id initial gender age company ruser interest
1 RC001     PKW      M  27     KTH     Y        1
2 RC002     KYG      M  30   그루터     Y        2
```

```
data a1;
infile datalines delimiter=','
    MISSOVER DSD firstobs=1;
input id $ initial $ gender $ age company $ ruser $ interest;
datalines;
RC001,PKW,M,27,KTH,Y,1
RC002,KYG,M,30,그루터,Y,2
;
run;
```

	id	initial	gender	age	company	ruser	interest
1	RC001	PKW	M	27	KTH	Y	1
2	RC002	KYG	M	30	그루터	Y	2

infile의 option

DLM,DSD,EXPANDTABS

LRECL

LINESIZE

FLOWOVER,MISSOVER,STOPOVER

PAD

Task	R	SAS
Data Import (3)	method 3> ``` > x<- + ("id,initial,gender,age,company,ruser,interest + RC001,PKW,M,27,KTH,Y,1 + RC002,KYG,M,30,그루터,Y,2") > > a1<-read.table(textConnection(x), + header=TRUE,sep=",") > a1 id initial gender age company ruser interest 1 RC001 PKW M 27 KTH Y 1 2 RC002 KYG M 30 그루터 Y 2 ```	

	Variable type	

```
> summary(a1)
      id              initial            gender
 Length:10          Length:10          F:3
 Class :character   Class :character   M:7
 Mode  :character   Mode  :character


      age            company    ruser interest
 Min.   :26.00   Nexon    :2   N:4   1:3
 1st Qu.:28.00   그루터    :1   Y:6   2:3
 Median :29.00   삼성전자 :1          3:3
 Mean   :29.70   서울대   :1          4:1
 3rd Qu.:31.75   코난 Tech:1
 Max.   :35.00   Daum     :1
                 (Other)  :3
```

<attendee1에 대한 요약>

칼럼명	유형	길이	출력형식	입력형식
Aa id	문자열	5	$5,	$5,
Aa initial	문자열	3	$3,	$3,
Aa gender	문자열	1	$1,	$1,
age	숫자	8	BEST12,	BEST32,
Aa company	문자열	11	$11,	$11,
Aa ruser	문자열	1	$1,	$1,
interest	숫자	8	BEST12,	BEST32,

<attendee1의 속성>

Task	R	SAS
	Comparison of data importing code	

Data Import (4)	**read.table vs. read.csv/delim**
	
	read.csv ➔ 콤마(,)로 구분되어 있는 데이터
	　　　　header=T / sep=","가 default 값
	
	read.delim ➔ 탭으로 구분되어 있는 데이터
	　　　　header=T / sep="\t"가 default 값
	
	read.xls ➔ `install.packages("xlsReadWrite")`
	　　　　`library("xlsReadWrite")`
	　　　　colNames = T가 default 값
	
	read.table ➔ header=F / sep=""가 default 값
		PROC IMPORT PROCEDURE
		
		```PROC IMPORT OUT=```
		``` DATAFILE=```
		``` DBMS=CSV/TAB/EXCEL REPLACE;```
		``` GETNAMES=YES;```
		``` DATAROW=2; *엑셀인 경우 SHEET= 추가됨;```
		```RUN;```

	Reading a SAS file to R	

	`install.packages("sas7bdat")`
	`library(sas7bdat)`
	`read.sas7bdat("data.sas7bdat")`	

Task	R	SAS
Missing Values (1)		

id	initial	gender	age	company	ruser	interest
RC001	PKW	M	27	KTH	Y	1
RC002	KYG	M	30	그루터	Y	2
RC003	KYK		26	서울대	Y	1
RC004	KSG	F	28	삼성전자	N	3
RC005	PSH	M	32	Daum	Y	2
RC006	KHW	M	35	KAIST	N	3
RC007	KSW	M	28	Nexon	Y	3
RC008	KJS	M		Nexon	Y	1
RC009		F	28	코난 Tech	N	2
RC010	KCW	M	31	Geference	N	

<예제 데이터: attendee1_missing>

```
> a1.miss=read.table(file="attendee1_missing.csv",header=T,sep=",",
+ na.strings="",
+ colClasses=c("character","character","factor",
+ "numeric","factor","factor","factor"))
> a1.miss
      id initial gender age    company ruser interest
1  RC001     PKW      M  27        KTH     Y        1
2  RC002     KYG      M  30      그루터     Y        2
3  RC003     KYK   <NA>  26      서울대     Y        1
4  RC004     KSG      F  28      삼성전자    N        3
5  RC005     PSH      M  32       Daum     Y        2
6  RC006     KHW      M  35      KAIST     N        3
7  RC007     KSW      M  28      Nexon     Y        3
8  RC008     KJS      M  NA      Nexon     Y        1
9  RC009    <NA>      F  28    코난 Tech    N        2
10 RC010     KCW      M  31  Geference     N     <NA>
```

<attendee1_missing의 R코드와 데이터>

```
proc import out=khe.a1_miss
      datafile="C:\KHE\attendee1_missing.csv"
      dbms=csv;
getnames=yes;
      datarow=2;
run;
```

	id	initial	gender	age	company	ruser	interest
1	RC001	PKW	M	27	KTH	Y	1
2	RC002	KYG	M	30	그루터	Y	2
3	RC003	KYK		26	서울대	Y	1
4	RC004	KSG	F	28	삼성전자	N	3
5	RC005	PSH	M	32	Daum	Y	2
6	RC006	KHW	M	35	KAIST	N	3
7	RC007	KSW	M	28	Nexon	Y	3
8	RC008	KJS	M	.	Nexon	Y	1
9	RC009		F	28	코난 Tech	N	2
10	RC010	KCW	M	31	Geference	N	.

<attendee1_missing의 SAS코드와 데이터>

LSK Global PS
LSK Global Pharma Services Co., Ltd.

Task	R	SAS
	Missing value representation	
Missing Values (2)	character/factor variables ➔ <NA> numeric variables ➔ NA 　※NULL: no mode 　　　used for building up vectors in loops	character variables ➔ " " (blank) numeric variables ➔ . (period 마침표)
	Finding missing values	
| | ```
> a1.miss[!complete.cases(a1.miss),]
     id initial gender age    company ruser interest
3  RC003    KYK    <NA>  26     서울대     Y        1
8  RC008    KJS       M  NA      Nexon     Y        1
9  RC009   <NA>       F  28  코난 Tech     N        2
10 RC010    KCW       M  31  Geference     N     <NA>
> nrow(a1.miss[!complete.cases(a1.miss),])
[1] 4
``` | ```
⊟ proc print data=khe.a1_miss;
    where id="" or    initial="" or  gender="" or
        age=. or company="" or  ruser="" or interest=.;
    run;
``` |

OBS	id	initial	gender	age	company	ruser	interest
3	RC003	KYK		26	서울대	Y	1
8	RC008	KJS	M	.	Nexon	Y	1
9	RC009		F	28	코난 Tech	N	2
10	RC010	KCW	M	31	Geference	N	.

LSK Global PS
LSK Global Pharma Services Co., Ltd.

Task	R	SAS
Data Merge (1)		

no	initial	gender	age	company	eval	new
RC001	PKW	M	27	KTH	8	0
RC003	KYK	F	26	서울대	7	0
RC005	PSH	M	32	Daum	6	0
RC007	KSW	M	28	Nexon	9	0
RC009	PJY	F	28	코난 Tech	8	0
RC201	LHK	M	31	KT	8	1
RC202	LYK	M	29	유웨이어플라이	10	1
RC203	DHJ	F	30	LG CNS	9	1
RC204	JSH	M	28	NCSOFT	7	1
RC205	JJY	M	30	삼성전자	9	1

<예제 데이터: attendee2>

no: attendee1 데이터의 id와 동일한 변수

eval: 컨퍼런스 평가 점수

new: 추가 등록 여부

	R	SAS
일대일 가로결합	cbind(a1,a2)	set a1; set a2; 또는 merge
상하 자료결합	rbind(a1, a2)	set a1 a2;
가로 대응결합	merge(a1, a2)	merge a1, a2; by id;

Task	R	SAS
	Inner join	

R:
```
> merge1=merge(a1,a2)
> merge1=merge1[order(merge1$id),]
> merge1
  initial gender age    company    id ruser interest    no eval new
4     PKW      M  27        KTH RC001     Y           1 RC001    8   0
2     KYK      F  26       서울대 RC003    Y           1 RC003    7   0
5     PSH      M  32       Daum RC005     Y           2 RC005    6   0
1     KSW      M  28      Nexon RC007     Y           3 RC007    9   0
3     PJY      F  28  코난 Tech RC009     N           2 RC009    8   0
```

```
merge1=merge(a1,a2)

       =merge(a1,a2,x.by="id")

       =merge(a1,a2,y.by="no")

       =merge(a1,a2,x.by="id",y.by="no")
```

SAS:
```
data merge1 merge2 merge3 merge4 merge5;
  merge khe.a1(in=ind1) khe.a2(in=ind2 rename=(no=id));
    by id;
       if ind1 and ind2 then output merge1;
       if ind1 then output merge2;
       if ind2 then output merge3;
       if ind1 and ind2 = 0 then output merge4;
       if ind1 = 0 and ind2 then output merge5;
  run;
```

	id	initial	gender	age	company	ruser	interest	eval	new
1	RC001	PKW	M	27	KTH	Y	1	8	0
2	RC003	KYK	F	26	서울대	Y	1	7	0
3	RC005	PSH	M	32	Daum	Y	2	6	0
4	RC007	KSW	M	28	Nexon	Y	3	9	0
5	RC009	PJY	F	28	코난 Tech	N	2	8	0

Merge Data (2)

| | **Left join** | |

R:
```
> merge2=merge(a1,a2,all.x=T)
> merge2=merge2[order(merge2$id),]
> merge2
   initial gender age     company    id ruser interest    no eval  new
9      PKW      M  27         KTH RC001     Y          1 RC001    8    0
6      KYG      M  30      그루터 RC002     Y          2  <NA>   NA <NA>
7      KYK      F  26       서울대 RC003    Y          1 RC003    7    0
4      KSG      F  28     삼성전자 RC004    N          3  <NA>   NA <NA>
10     PSH      M  32        Daum RC005     Y          2 RC005    6    0
2      KHW      M  35       KAIST RC006     N          3  <NA>   NA <NA>
5      KSW      M  28       Nexon RC007     Y          3 RC007    9    0
3      KJS      M  32       Nexon RC008     Y          1  <NA>   NA <NA>
8      PJY      F  28   코난 Tech RC009     N          2 RC009    8    0
1      KCW      M  31  Geference RC010     N          4  <NA>   NA <NA>
```

	id	initial	gender	age	company	ruser	interest	eval	new
1	RC001	PKW	M	27	KTH	Y	1	8	0
2	RC002	KYG	M	30	그루터	Y	2	.	.
3	RC003	KYK	F	26	서울대	Y	1	7	0
4	RC004	KSG	F	28	삼성전자	N	3	.	.
5	RC005	PSH	M	32	Daum	Y	2	6	0
6	RC006	KHW	M	35	KAIST	N	3	.	.
7	RC007	KSW	M	28	Nexon	Y	3	9	0
8	RC008	KJS	M	32	Nexon	Y	1	.	.
9	RC009	PJY	F	28	코난 Tech	N	2	8	0
10	RC010	KCW	M	31	Geference	N	4	.	.

Task	R	SAS
	Right join	

R column:

```
> merge3=merge(a1,a2,all.y=T)
> merge3=merge3[order(merge3$id,merge3$no),]
> merge3
     initial gender age        company     id ruser interest      no eval new
9      PKW      M   27           KTH RC001     Y            1 RC001    8   0
5      KYK      F   26        서울대 RC003     Y            1 RC003    7   0
10     PSH      M   32          Daum RC005     Y            2 RC005    6   0
4      KSW      M   28         Nexon RC007     Y            3 RC007    9   0
8      PJY      F   28     코난 Tech RC009     N            2 RC009    8   0
6      LHK      M   31            KT  <NA>  <NA>         <NA> RC201    8   1
7      LYK      M   29 유웨이어플라이  <NA>  <NA>         <NA> RC202   10   1
1      DHJ      F   30        LG CNS  <NA>  <NA>         <NA> RC203    9   1
3      JSH      M   28        NCSOFT  <NA>  <NA>         <NA> RC204    7   1
2      JJY      M   30      삼성전자  <NA>  <NA>         <NA> RC205    9   1
```

SAS column (first table):

	id	initial	gender	age	company	ruser	interest	eval	new
1	RC001	PKW	M	27	KTH	Y	1	8	0
2	RC003	KYK	F	26	서울대	Y	1	7	0
3	RC005	PSH	M	32	Daum	Y	2	6	0
4	RC007	KSW	M	28	Nexon	Y	3	9	0
5	RC009	PJY	F	28	코난 Tech	N	2	8	0
6	RC201	LHK	M	31	KT	.		8	1
7	RC202	LYK	M	29	유웨이어플	.		10	1
8	RC203	DHJ	F	30	LG CNS	.		9	1
9	RC204	JSH	M	28	NCSOFT	.		7	1
10	RC205	JJY	M	30	삼성전자	.		9	1

Task: Merge Data (3)

R column (lower):

merge()의 arguments

x.by or y.by ➜ 매칭의 기준 변수 선택

all.x=T or all.y=T (left join or right join)

➜ all.x=T의 경우 매칭이 안된 x의 나머지 데이터도 함께 merge됨

all=F   ➜ inner join

all=T   ➜ outer join

SAS column (merge4 table):

	id	initial	gender	age	company	ruser	interest	eval	new
1	RC002	KYG	M	30	그루터	Y	2	.	.
2	RC004	KSG	F	28	삼성전자	N	3	.	.
3	RC006	KHW	M	35	KAIST	N	3	.	.
4	RC008	KJS	M	32	Nexon	Y	1	.	.
5	RC010	KCW	M	31	Geference	N	4	.	.

<merge4: a1- (a1∩a2)>

SAS column (merge5 table):

	id	initial	gender	age	company	ruser	interest	eval	new
1	RC201	LHK	M	31	KT	.		8	1
2	RC202	LYK	M	29	유웨이어플	.		10	1
3	RC203	DHJ	F	30	LG CNS	.		9	1
4	RC204	JSH	M	28	NCSOFT	.		7	1
5	RC205	JJY	M	30	삼성전자	.		9	1

<merge5: a2- (a1∩a2)>

Task	R	SAS
Merge Data (4)	 **SAS merge시 주의사항** **merge문에 배치되는 데이터의 순서 (merge a1 a2 or merge a2 a1)에 따라** **변수의 length 달라짐 (데이터 손실 가능성 있음)**	

<a1의 속성>

<a2의 속성>

```
data merge3_a1a2;
  merge khe.a1(in=ind1) khe.a2(in=ind2 rename=(no=id)) ;
    by id;
      if ind2 then output merge3_a1a2;
  run;
```

```
data merge3_a2a1;
  merge khe.a2(in=ind2 rename=(no=id)) khe.a1(in=ind1);
    by id;
      if ind2 then output merge3_a2a1;
  run;
```

Task	R	SAS
Merge Data (5)	 **<merge3_a2a1>**	

For the Variable Selection row:

R

```
> eval=merge3[c("no","initial","eval")]
> eval
     no initial eval
9   RC001    PKW    8
5   RC003    KYK    7
10  RC005    PSH    6
4   RC007    KSW    9
8   RC009    PJY    8
6   RC201    LHK    8
7   RC202    LYK   10
1   RC203    DHJ    9
3   RC204    JSH    7
2   RC205    JJY    9
```

```
eval=merge3[,c(1,8,9)]  ➔ 동일한 결과
```

SAS

```
data eval;
    set merge3;
        keep id eval;
run;
```
➔ keep 이용

	id	eval
1	RC001	8
2	RC003	7
3	RC005	6
4	RC007	9
5	RC009	8
6	RC201	8
7	RC202	10
8	RC203	9
9	RC204	7
10	RC205	9

```
data eval;
    set merge3;
    drop initial gender age company ruser interest new;
run;
```
➔ drop 이용

Task	R	SAS

Renam-ing	```> names(a2)```
	 ```[1] "no" "initial" "gender" "age" "company" "eval" "new"```
	 ```> names(a2)[1]="id"```
	 ```> names(a2)```
	 ```[1] "id" "initial" "gender" "age" "company" "eval" "new"```
	
	 ※ reshape 패키지의 rename 함수 이용
	 `install.packages("reshape")`
	 `library(reshape)`
	 `rename()`
		```data a2;```
		 ``` set a2;```
		 ``` rename no=id;```
		 ```run;```

If/Then/Else	female.new=a1[which(gender=="F" & new==1),]
	
	 female.new=a1[a1$gender=="F" & a1$new==1),]
	
	 if (cond) expression1 else expression2
	
	 if (cond) {expression1} else {expression2}
	
	 ifelse(cond,expression1,expression2)
		data female_new;
		
		 set a1;
		
		 where gender="F" and new=1;
		
		 run;
		
		 if cond then expression1;else expression2;
		
		 if cond then do; expression1-3; end;
		
		 ifn(if, then, else)

Loop	```for (i in 1:10){```
	 ```<list of R instructions>```
	 ```}```
		```do i=1 to 10```
		 ```<list of SAS instructions>```
		 ```end;```

Task	R	SAS
Creating New Function (1)	new_func=function(<list of parameters>) { <list of R instructions> } newfunc(list of parameters)	%macro new_func((<list of parameters>); <list of SAS instructions>; %mend new_func; %new_func(list of parameters)

R:
```
> se <- function(x) {
+ v <- var(x)
+ n <- length(x)
+ return(sqrt(v/n))
+ }
> se(1:10)
[1] 0.9574271
```

SAS:
```
%macro se(data, x);

proc means data=&data;
    var &x;
    output out=save n=_n var=_var;
run;

data _null_;
set save;
    stderr=sqrt(_var/_n);
    put stderr;
run;

%mend se;

data one;
input x @@;
cards;
1 2 3 4 5 6 7 8 9 10
;
run;

%se(one,x);
```

LSK Global PS
LSK Global Pharma Services Co., Ltd.

Task	R	SAS
| **Creating New Function (2)** | | ```
1718  %se(one,x);
MLOGIC(SE):  Beginning execution.
MLOGIC(SE):  Parameter DATA has value one
MLOGIC(SE):  Parameter X has value x
MPRINT(SE):   proc means data=one;
MPRINT(SE):   var x;
MPRINT(SE):   output out=save n=_n var=_var;
MPRINT(SE):   run;

NOTE: 10개의 관측치를 데이터셋 WORK.ONE. (으)로부터 읽었습니다.
NOTE: 데이터셋 'WORK.SAVE'은(는) 1개 관측치, 4개 변수를 가지고 있습니다.
NOTE: 프로시저 MEANS 실행:
      실행 시간          0.01 초
      cpu 시간           0.01 초

MPRINT(SE):   data _null_;
MPRINT(SE):   set save;
MPRINT(SE):   stderr=sqrt(_var/_n);
MPRINT(SE):   put stderr;
MPRINT(SE):   run;
0.9574271078
NOTE: 1개의 관측치를 데이터셋 WORK.SAVE. (으)로부터 읽었습니다.
NOTE: DATA 문장 실행:
      실행 시간          0.00 초
      cpu 시간           0.00 초
``` |
| **Execut-ing code** | path에 func_khe.txt. 또는 R 스크립트 파일 저장한 경우

source("path/new_func.txt") | path에 func_khe.sas 파일 저장한 경우

%include "path\ new_func.sas" |

Task	R	SAS
Vectori-zation (1)	**Vectorization** *Where functions are applied element-wise to vectors* ● **is One of the most powerful aspects of the R language** ● **operates directly on each element of a vector.** ● **leads the function to produce a vector of the same length, with each element resulting from applying the function to every element of the original vector.** ``` > v <- c(4,7,23.5,76.2,80) > x <- sqrt(v) > x [1] 2.000000 2.645751 4.847680 8.729261 8.944272 ``` **Achieving Better Speed in a Monte Carlo Simulation in The Art of R Programming** 1번 항아리에는 파란 구슬 10개, 노란 구슬 8개, 2번 항아리에는 파란 구슬 6개, 노란 구슬 6개가 들어있다. 항아리 1에서 무작위로 구슬 한 개를 뽑아서 항아리 2에 넣고, 다시 항아리 2에서 무작위로 구슬 하나를 뽑는다. 두 번째로 뽑은 구슬이 파란구슬일 확률은?	

Task	R	SAS

Vectori-zation (2)	method 1> straightforward way
	
	``
	sim1 <- function(nreps) {
	    nb1 <- 10 # 10 blue marbles in Urn 1
	    n1 <- 18 # number of marbles in Urn 1 at 1st pick
	    n2 <- 13 # number of marbles in Urn 2 at 2nd pick
	    count <- 0 # nmber of repetitions in which get blue from Urn 2
	
	    for (i in 1:nreps) {
	        nb2 <- 6 # 6 blue marbles orig. in Urn 2
	        # pick from Urn 1 and put in Urn 2. Is it blue?
	        if (runif(1) < nb2/n2) count <- count + 1
	    }
	
	    return(count/nreps) #estimate Prob(pick blue from Urn2)
	}
	``	

LSK Global PS
LSK Global Pharma Services Co., Ltd.

Task	R	SAS

Vectori-zation (3)	method 2> using apply()
	
	``
	sim2 <- function(nreps) {
	    nb1 <- 10
	    nb2 <- 6
	    n1 <- 18
	    n2 <- 13
	    count <- 0
	
	    # pre-generate all our random numbers, one row per repetition
	    u <- matrix(c(runif(2*nreps)), nrow=nreps, ncol=2)
	
	    # define simfun for use in apply(). simulates one repetition
	      simfun <- function(rw) {
	        # rw ("row") is a pair of randome numbers
	        # choose form Urn 1
	        if (rw[2] < nb1/n1) nb2 <- nb2 + 1
	        # choose from Urn 2, and return boolean on choosing blue
	        return (rw[2] < nb2/n2)
	      }
	
	    z <- apply(u, 1, simfun)
	    # z is a vector of booleans but they can be treated as 1s, 0s
	    return(mean(z))
	}
	``	

Task	R	SAS
| Vectori-zation (4) | **method 3> using vectorization**<br><br>```
sim3 <- function(nreps){
    nb1 <- 10
    nb2 <- 6
    n1 <- 18
    n2 <- 13
    u <- matrix(c(runif(2*nreps)), nrow=nreps, ncol=2)

    # set up the condition vector
    cndtn <- u[,1] <= nb1/n1 & u[,2] <= (nb2+1)/n2 |
             u[,1]  > nb1/n1 & u[,2] <= (nb2)/n2
    return(mean(cndtn))
}
``` | |

Comparison of Running time		
sim1	sim2	sim3
| ```
> system.time(print(sim1(100000)))
[1] 0.46275
   user   system elapsed
   1.01    0.00    1.02
``` | ```
> system.time(print(sim2(100000)))
[1] 0.53756
   user   system elapsed
   2.25    0.00    2.25
``` | ```
> system.time(print(sim3(100000)))
[1] 0.50513
   user   system elapsed
   0.08    0.00    0.08
``` |
| ➔ **vectorization을 이용한 sim3이 가장 빠름** | | |

19

Prepared by Kwak HeeEun   LSK Global PS
LSK Global Pharma Services Co., Ltd.

Task	R	SAS
Output Saving System (1)	(see R code below)	(see SAS code below)

R:

```
> save1=summary(a1$age)
> save1
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  26.00   28.00   29.00   29.70   31.75   35.00
```

※ **Regression 예제**

```
> ## Annette Dobson (1990)
> ## "An Introduction to Generalized Linear Models".
> ## Page 9: Plant Weight Data.
>
> ctl <- c(4.17,5.58,5.18,6.11,4.50,4.61,5.17,4.53,5.33,5.14)
> trt <- c(4.81,4.17,4.41,3.59,5.87,3.83,6.03,4.89,4.32,4.69)
> group <- gl(2,10,20, labels=c("Ctl","Trt"))
>
> weight <- c(ctl, trt)
> model <- lm(weight ~ group)
> model

Call:
lm(formula = weight ~ group)

Coefficients:
(Intercept)     groupTrt
      5.032       -0.371
```

SAS:

method 1> output out

```
proc means data=khe.a1;
  var age;
  output out=save1;
run;
```

	TYPE	_FREQ_	_STAT_	age
1	0	10	N	10
2	0	10	MIN	26
3	0	10	MAX	35
4	0	10	MEAN	29.7
5	0	10	STD	2.7908580919

method 2> ods trace on

```
ods trace on;
proc means data=khe.a1;
  var age;
run;
```

```
NOTE: HTML Body 파일 기록 중: sashtml3.htm
추가된 출력물:
-------------
이름:        Summary
라벨:        Summary statistics
템플릿:      base.summary
경로:        Means.Summary
-------------
```

Task	R	SAS
| **Output Saving System (2)** | ```
> summary(model)

Call:
lm(formula = weight ~ group)

Residuals:
    Min      1Q  Median      3Q     Max
-1.0710 -0.4938  0.0685  0.2462  1.3690

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   5.0320     0.2202  22.850 9.55e-15 ***
groupTrt     -0.3710     0.3114  -1.191    0.249
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.6964 on 18 degrees of freedom
Multiple R-squared: 0.07308,    Adjusted R-squared: 0.02158
F-statistic: 1.419 on 1 and 18 DF,  p-value: 0.249


> summary(model)$coefficients[2,4]
[1] 0.2490232
```<br><br>➔ 원하는 값 모두 쉽게 바로 저장 가능<br> attribute에 출력되는 모든 값 뽑아 쓸 수 있음 | **method3> ODS Output Object Table Names**<br><br>**SAS(R) 9.2 Output Delivery System: User's Guide**<br><br>*ODS Table Names Produced by the MEANS and SUMMARY Procedures*<br><br>For detail information, see the MEANS and SUMMARY procedures.<br><br>| Table Name | Description |<br>|---|---|<br>| Summary | Summary of descriptive statistics for variables across all observations and within groups of observations |<br><br>```
ods output Summary=save1;
proc means data=khe.a1;
var age;
run;
ods output close;
```<br> Save1<br><br>| | N | 평균값 | 표준편차 | 최소값 | 최대값 |<br>|---|---|---|---|---|---|<br>| 1 | 10 | 29,7 | 2,7908580919 | 26 | 35 |<br><br>➔ **Table name 과 Variable name**을 찾아야 하는 번거로움이 있음 |
| **Help System** | help.search("t.test") 또는 ??t.test
 ➔ 검색어와 관련된 함수 리스트 알 수 있음
?t.test   ➔ R documentation이 팝업으로 뜸
Example(t.test)
 ➔ 예제를 실행시켜서 보여줌! | SAS 도움말

http://support.sas.com/resources/ |

APPENDIX1. Statistical Analysis (from http://r4stats.com/)

Method	R	SAS

Analysis of Variance	```myModel <- aov(posttest ~ workshop,```
	```data = mydata100)```
	```summary(myModel)```
	
	```pairwise.t.test(posttest, workshop)```
	```TukeyHSD(myModel, "workshop")```
	```plot(TukeyHSD(myModel, "workshop"))```
		```PROC GLM;```
		```CLASS workshop;```
		```MODEL posttest = workshop;```
		```MEANS workshop / TUKEY;```

Correlate, Pearson	```cor(mydata[3:6],```
	``` method = "pearson",```
	``` use = "pairwise")```
	```cor.test(mydata$q1,```
	``` mydata$q2, use = "pairwise")```
	
	```library("Rcmdr")```
	```rcorr.adjust(mydata[3:6])```
		```PROC CORR; VAR q1-q4; RUN;```

Correlate, Spearman	```cor(mydata[3:6],```
	``` method = "spearman",```
	``` use = "pairwise")```
	```cor.test(mydata$q1,```
	``` mydata$q2,```
	``` use = "pairwise")```
	
	```library("Rcmdr")```
		```PROC CORR SPEARMAN; VAR q1-q4;```

| | ```
rcorr.adjust( mydata[3:6] )
``` | |
| **Crosstabulation & Chi-squared** | ```
myWG <- table(workshop, gender)
chisq.test(myWG)


library("gmodels")
CrossTable(workshop, gender,
  chisq = TRUE,
 format = "SAS")
``` | ```
PROC FREQ;
TABLES workshop*gender /CHISQ;
``` |
| **Descriptive Stats** | ```
summary(mydata)


library("Hmisc")
describe(mydata)
``` | ```
PROC MEANS;
VAR q1--posttest;


PROC UNIVARIATE; VAR q1--posttest;
``` |

| Frequencies | ```
summary(mydata)


library("Deducer")
frequencies(mydata)
``` | ```
PROC FREQ;
TABLES workshop--q4;
``` |
| --- | --- | --- |
| Kruskal-Wallis | ```
kruskal.test(posttest ~
  workshop)


pairwise.wilcox.test(posttest,
  workshop)
``` | ```
PROC npar1way;
  CLASS workshop;
  VAR posttest;
``` |
| Regression, Linear | ```
myModel <- lm(q4 ~ q1 + q2 + q3,
  data = mydata100)
summary(myModel)
plot(myModel)
``` | ```
PROC REG;
  MODEL q4=q1-q3;
``` |
| Sign Test | ```
library("PASWR")
SIGN.test(posttest, pretest,
  conf.level = .95)
``` | ```
myDiff=posttest-pretest;
PROC UNIVARIATE;
  VAR myDiff;
``` |
| t-Test, Independent | ```
t.test(q1 ~ gender,
  data = mydata100)
``` | ```
PROC TTEST;
  CLASS gender;
  VAR q1;
``` |

t-Test, Paired	`t.test(posttest, pretest,`
	`  paired = TRUE)`
		`PROC TTEST;`
		`PAIRED pretest*`
		`posttest;`

Variance Test	`# Bartlett's`
	`var.test(posttest ~ gender)`
	
	`# Levene's`
	`library("car")`
	`levene.test(posttest, gender)`	

Wilcoxon Rank Sum (Mann-Whitney)	`wilcox.test(q1 ~ gender,`
	`  data = mydata100)`
		`PROC NPAR1WAY;`
		`  CLASS gender;`
		`  VAR q1;`

Wilcoxon Signed Rank (Paired)	`wilcox.test(posttest, pretest,`
	`  paired = TRUE)`
		`myDiff=posttest-pretest;`
		`PROC UNIVARIATE;`
		`  VAR myDiff;`

※ **Reference**

Data Mining with R: learning by case studies / Luis Torgo

The Art of R Programming / Norman Matloff

R for SAS and SPSS Users / Robert A. Muenchen (http://r4stats.com/)

An Introduction to the SAS System / Phil Spector

http://www.stat.berkeley.edu/classes/s100/sas.pdf

http://www.statwith.pe.kr/STAT_PRO/functions001.htm

http://cran.r-project.org/

http://support.sas.com/