Chapter 1

리스트/데이터프레임/요인 - 데이터 조작실무

이번 챕터부터는 실질적으로 데이터를 다루게되는 경우를 집중적으로 살펴보도록 하겠습니다. 데이터를 다루는 데 있어서 이전 챕터에서 다루었던, 벡터, 행렬, 그리고 배열이라는 데이터형 외에 요인, 데이터프레임과 리스트라는 형식을 반드시 알고 있어야 합니다. 이러한 형식의 제공은 통계적 프로그래밍의 관점에서 제공되어지는 것입니다. 따라서, 데이터 프레임과 리스트에 대한 내용을 먼저 설명한 뒤에 데이터 클리닝에 필요한 다양한 조작법들에 대해서 알아보도록 합니다.

1.1 리스트

데이터 프레임을 설명하기 전에 리스트라는 데이터형을 먼저 설명합니다. 그 이유는 데이터 프레임은 리스트의 특수한 경우이기 때문입니 다. 리스트는 벡터와 같은 방식으로 생성되고 사용되지만, 한가지 다른 점이 있습니다. 벡터의 경우에는 해당 벡터의 모든 구성요소는 모두 같은 데이터형을 가지고 있어야 합니다. 예를들어, 구성요소가 숫자라면 수치형 벡터라고 특징지을 수 있는데, 이는 해당벡터의 모든 구성요소가 예외없이 숫자형만을 가져야 하기 때문입니다. 문자형 벡터의 경우에도 마찬가지 입니다. 벡터의 구성요소 각각이 모두 문자형만을 가져야만 합니다.

```
> x <- 1:5
> x
[1] 1 2 3 4 5
> y <- LETTERS[1:5]
> y
[1] "A" "B" "C" "D" "E"
> mode(x)
[1] "numeric"
> mode(y)
[1] "character"
```

그러나, 리스트는 구성요소의 데이터형에 구애받지 않습니다. 예를들면, 리스트의 첫번째 구성요소가

문자형 값을 가질때 두번째 구성요소는 숫자형 값을 가질 수 있습니다. 좀 더 나아가 리스트의 첫번째 구성요소가 문자형 벡터를 가질 때, 두번째 구성요소는 수치형 행렬을 가지고, 세번째 구성요소는 숫자형 배열을 가질 수도 있습니다. 그리고, 네번째 구성요소는 우리가 다음 섹션에서 다루게 될 데이터 프레임이라는 형식을 가질 수도 있으며, 다섯번째 구성요소가 현재 설명하고 있는 리스트형을 가질 수도 있습니다.

이러한 리스트는 list()함수를 이용하여 아래와 같은 방법으로 생성하게 됩니다.

```
> x <- 1:5
> y <- LETTERS[1:5]
> z <- matrix(c(1,2,3,4,5,6), ncol=3)
> xyz <- list(x,y,z)
> xyz
[[1]]
[1] 1 2 3 4 5

[[2]]
[1] "A" "B" "C" "D" "E"

[[3]]
     [,1] [,2] [,3]
[1,]    1    3    5
[2,]    2    4    6
```

이렇게 생성된 리스트의 구성요소들은 [[와]] 를 이용하여 가져올 수 있습니다.

```
> xyz[[1]]
[1] 1 2 3 4 5
> xyz[[2]]
[1] "A" "B" "C" "D" "E"
> xyz[[3]]
     [,1] [,2] [,3]
[1,]    1    3    5
[2,]    2    4    6
```

또한, 리스트의 구성요소를 구성하는 구성요소들을 아래와 같은 방법으로 가져올 수 있습니다.

```
> xyz[[1]][3]
[1] 3
> xyz[[3]][2,3]
[1] 6
> xyz[[2]][3]
[1] "C"
```

이렇게 주어진 리스트는 아래와 같이 mode()를 이용하여 확인된 것과 같이 list라는 형식을 가지고 있으며, is.list()라는 함수를 통하여 확인이 가능합니다.

```
> mode(xyz)
[1] "list"
> is.list(xyz)
[1] TRUE
```

그런데, 만약 리스트의 구성요소의 개수들이 많아진다면 이 리스트의 구조를 살펴보는 것이 리스트라는 데이터형을 다루는데 도움이 될 것이며, 이를 위해서는 str()이라는 함수를 사용합니다.

```
> str(xyz)
List of 3
 $ : int [1:5] 1 2 3 4 5
 $ : chr [1:5] "A" "B" "C" "D" ...
 $ : num [1:2, 1:3] 1 2 3 4 5 6
```

이러한 리스트에 이름을 붙이는 방법은 아래와 같습니다.

```
> mylist <- list(x=x, y=y, z=z)
> mylist
$x
[1] 1 2 3 4 5

$y
[1] "A" "B" "C" "D" "E"

$z
     [,1] [,2] [,3]
[1,]    1    3    5
[2,]    2    4    6
```

이렇게 이름이 붙여진 리스트 mylist 라는 것은 아래와 같이 names()라는 함수를 이용하여 확인이 가능합니다.

```
> names(mylist)
[1] "x" "y" "z"
```

이렇게 이름이 부여된 이후에는 리스트 구성요소의 이름을 이용하여 불러올 수 있습니다.

```
> mylist$x
[1] 1 2 3 4 5
> mylist$y
[1] "A" "B" "C" "D" "E"
```

```
> mylist$z
     [,1] [,2] [,3]
[1,]    1    3    5
[2,]    2    4    6
>
```

R에서 제공하는 많은 통계관련 함수들은 이러한 리스트의 특징을 활용합니다. list()는 이후에 설명하겠지만, 사용자 함수의 작성시에 여러개의 값들을 한번에 반환하고자 할때 return() 대신 이용됩니다.

1.2 데이터 프레임

이제 데이터 프레임이라는 데이터형에 대해서 살펴보도록 합니다. 위에서 언급한 바와 같이 데이터 프레임이란 리스트의 특수한 경우입니다. 데이터의 생성과 활용방법은 리스트와 동일하지만 두 가지 부분에서 다릅니다. 하나는 리스트의 구성요소들이 벡터형이어야 합니다. 이때 벡터가 숫자형인지 문자형인지에 대한 종류에는 관계가 없습니다. 또 다른 하나는 모든 리스트의 구성요소들이 같은 길이를 가져야 합니다. 이와 같은 두개의 조건이 성립할때 리스트를 데이터 프레임이라고도 합니다. 이 데이터프레임은 통계분석에 있어서 데이터가 저장되어 있는 스프레드 형식을 가지기 때문에 매우 유용하게 활용될 수 있습니다.

다음은 데이터 프레임을 생성하는 방법인데, 리스트를 생성하는 방법과 동일하다는 것을 알 것입니다.

```
> v1 <- c(163, 178, 170, 167, 169)
> v2 <- c("f", "m", "m", "f", "m")
> mydata <- data.frame(v1,v2)
> mydata
   v1 v2
1 163  f
2 178  m
3 170  m
4 167  f
5 169  m
```

이 데이터프레임의 이름을 변경할 수 있으며, 이 또한 리스트의 이름을 변경하는 것과 동일하다는 것을 알 수 있습니다.

```
> names(mydata)
[1] "v1" "v2"
> names(mydata) <- c("Height", "Gender")
> mydata
  Height Gender
1    163      f
2    178      m
3    170      m
4    167      f
5    169      m
>
```

이 데이터프레임은 어떤 속성이 있는지 attributes() 함수를 이용하여 확인해 봅니다.

```
> attributes(mydata)
$names
[1] "Height" "Gender"

$row.names
[1] 1 2 3 4 5

$class
[1] "data.frame"
>
```

행의 이름 또한 아래와 같은 방법으로 변경이 가능합니다.

```
> row.names(mydata)
[1] "1" "2" "3" "4" "5"
> row.names(mydata) <- c("ID-1", "ID-2", "ID-3", "ID-4", "ID-5")
> mydata
     Height Gender
ID-1    163      f
ID-2    178      m
ID-3    170      m
ID-4    167      f
ID-5    169      m
>
```

데이터 프레임은 리스트의 특수한 경우이기도 하지만, 행렬에 변수명이라는 속성을 붙인 것으로도 볼 수 있습니다. 따라서, 행렬에서 사용되는 함수들을 이용하여 차원, 열의 개수, 행의 개수들을 확인할 수 있습니다.

```
> dim(mydata)
[1] 5 2
> nrow(mydata)
[1] 5
> ncol(mydata)
[1] 2
>
```

1.3 요인

위의 데이터 프레임에서 여러가지 속성들을 덧붙인 mydata의 구조를 str()을 이용하여 살펴보면 다음과 같습니다.

```
> str(mydata)
'data.frame':        5 obs. of  2 variables:
 $ Height: num  163 178 170 167 169
 $ Gender: Factor w/ 2 levels "f","m": 1 2 2 1 2
>
```

그런데, Gender 라는 변수에 대해서 한가지 새로운 사실을 알게 됩니다. 분명히 v2라는 벡터를 이용했는데, str()를 이용하여 구조를 확인한 결과로는 2개의 수준들을 가진 요인 (Factor w/ 2 levels)라고 정보가 나타납니다. 이는 R이 통계분석적인 측면에서 디자인되었기 때문에 가지는 특징입니다. 모든 문자형 벡터들은 요인으로 간주되어지며, 해당 벡터가 가지는 값의 범위를 수준으로 인식하게 됩니다. 이러한 요인은 주로 범주형 데이터를 표시할때 이용됩니다.

1.4 데이터 조작 실무예제

이전 섹션까지 R에서 다루는 다양한 종류의 데이터형들에 대해서 알아보았습니다. 통계분석 실무에서는 분석자가 보통 얻게 되는 데이터는 분석에 사용되는 통계모형의 적용요건에 부합하는 경우는 드물기 때문에 분석자 스스로가 이러한 데이터를 형성하는 것은 필요한 기술중에 하나라고 할 수 있습니다.

서울종합과학대학원 사회학과의 신종화 교수님께서 본 섹션을 위해서 dart8.xls 이라는 데이터를 제공해 주셨습니다. 실무에서 접할 수 있는 다양한 점들을 부각하고자, 신종화 교수님으로부터 전달받은 데이터에는 그 어떠한 수정도 이루어지지 않았습니다. 분석을 위한 데이터들을 만들기 위해서 R은 이 데이터를 어떻게 받아들이고 처리했는지 상세히 기록하고자 하였습니다.

데이터셋에 대한 간단한 설명: 제공된 데이터셋은 8개의 여행사들의 "광고선전비", "교육훈련비", "매출액"을 2000년 12월부터 2011년 9월까지 월별로 기록한 내용이 dart8.xls 이라는 엑셀파일에 저장되어 있습니다. 이 자료는 여행사별 자료는 개별 워크시트에 따로 저장이 되어 있으며, 변수명은 한글이 사용되어 있습니다.

본 데이터는 여기 다운로드 링크를 눌러 다운받을 수 있습니다.

엑셀 데이터 불러오기: R을 이용하여 분석을 준비하고자 한다면 데이터를 R로 불러오는 것이 그 첫번째 작업일 것입니다. 이 데이터는 한 개의 엑셀파일에 있는 8개의 워크시트에 분산되어 있기 때문에 먼저 하나로 모두 모아야 합니다. 다행이도 각 워크시트에 정리되어 있는 데이터는 동일한 개수의 변수들이 있고, 변수들의 순서도 일치합니다. 이를 수행하기 위해서 gdata이라는 패키지를 먼저 불러옵니다.

```
> library(gdata)
```

예상치 못한 문제의 발생과 원인: 먼저 엑셀 파일에 제대로 읽힐 수 있는가를 확인하기 위해서 하나의 워크시트를 테스트용으로 읽어봅니다.

```
> tmp <- read.xls(xls="dart8.xls", sheet=1)
Wide character in print at /usr/lib/R/site-library/gdata/perl/xls2csv.pl line 211.
Wide character in print at /usr/lib/R/site-library/gdata/perl/xls2csv.pl line 270.
>
```

예상하지 않았던 문제가 발생했습니다. "wide character in print"라는 메시지는 R의 버그가 아니라, 엑셀 데이터를 파싱하는데 사용되는 Perl이라는 언어가 데이터내에 유니코드문자가 있을때 발생시키는 메시지

입니다. 즉, 한글을 표현하는 멀티바이트 인코딩 때문에 발생하는 것입니다. 이러한 메시지가 나왔을지라도 실제로 데이터를 확인해보시면 데이터에는 아무런 손상이 없음을 알 수 있습니다.

XLConnect 패키지의 활용: 그런데, 이러한 메시지를 보게되면 웬지 모르게 꺼림칙합니다. 아니면 원본데이터의 변수명 자체를 없애거나 변경하거나 해야할 것입니다. 그러나, 데이터클리닝시에는 원본 데이터에는 절대로 손을 대어서는 안됩니다. 모든 클리닝 작업은 기록화 되어 추후에도 자동적으로 처리가 될 수 있도록 해야합니다. 이것은 추후에 제3자에 의한 재생산연구(reproducible research) 가 가능하도록 하여 분석의 객관성을 유지할 수 있습니다.

따라서, XLConnect 라는 운영체제에 관계없이 사용될 수 있는 또 다른 종류의 패키지를 아래와 같이 불러왔습니다.

```
> library(XLConnect)
```

그리고 아래와 같이 데이터를 읽어왔더니, 어떠한 메시지 없이 잘 수행되었음을 볼 수 있었습니다.

```
> library(XLConnect)
> tmp <- readWorksheetFromFile(file="dart8.xls", sheet=1)
```

데이터의 처음과 마지막 부분을 확인: 정말로 잘 수행되었는지를 확인하기 위해서 데이터의 처음과 마지막을 살펴볼 수 있는 head()와 tail()함수를 이용해 봅니다.

```
> head(tmp)
  구....분 광고선전비 교육훈련비       매출액
1  2000.12  161702806   18002000  5616224889
2  2001.03   80485618   28146500  7188763335
3  2001.06  170827271   12965900  7948588645
4  2001.09   65667863   26468000 11509839298
5  2001.12   27804868   16838062  7799015935
6  2002.03   81945640   12752112 10491229385
> tail(tmp)
   구....분 광고선전비 교육훈련비       매출액
39  2010.06 2611994098   34151731 48451368536
40  2010.09 1723098483          0 66245202818
41  2010.12 2930265435   98244331 54941857566
42  2011.03 2449466000   36710000 63509496808
43  2011.06 2818383000   58809000 47554958058
44  2011.09 2357446000   27714000 66186330843
>
```

원본데이터 dart8.xls와 비교를 해보니 불러온 데이터에 아무런 오류가 없음을 확인할 수 있었습니다.

엑셀 파일내 모든 워크시트 다 불러오기: 그러나, 이것은 하나의 워크시트만을 불러온 것으로 전체 워크시트를 불러오고자 하는 우리의 목적을 달성한 것은 아닙니다. 따라서 아래와 같이 워크시트를 모두 불러오는 오도록 합니다.

```
> wb <- loadWorkbook("dart8.xls")
> wb
[1] "dart8.xls"
> tmp <- readWorksheet(wb, sheet=getSheets(wb))
>
```

각각의 워크시트가 리스트 tmp의 각 구성요소에 성공적으로 불러들여졌습니다. 실제로 이것은 아래와 같이 반복문의 개념을 통해 이루어진 것입니다.

모든 워크시트의 이름 확인: 각각의 워크시트를 읽어오기 위해서는 먼저 어떤 이름을 가진 워크시트가 몇개가 있는지 알아야 할 것입니다.

```
> wid <- getSheets(wb)
> wid
[1] "하나투어"     "레드캡투어"   "모두투어"     "세중"         "참좋은여행"
[6] "롯데관광개발" "자유투어"     "비티앤아이"
>
```

이 이름의 목록이 이미 불러들어온 목록과 일치함을 알 수 있습니다.

```
> names(tmp)
[1] "하나투어"     "레드캡투어"   "모두투어"     "세중"         "참좋은여행"
[6] "롯데관광개발" "자유투어"     "비티앤아이"
>
```

워크시트를 순차적으로 불러오는 반복문을 수행해 봅니다.

```
> tmp <- list()
> for(idx in getSheets(wb)) tmp[[idx]] <- readWorksheetFromFile(file="dart8.xls", sheet=idx)
> tmp
$하나투어
   구....분 광고선전비 교육훈련비      매출액
1   2000.12  161702806   18002000  5616224889
2   2001.03   80485618   28146500  7188763335
3   2001.06  170827271   12965900  7948588645
4   2001.09   65667863   26468000 11509839298
5   2001.12   27804868   16838062  7799015935
6   2002.03   81945640   12752112 10491229385
7   2002.06  303080492    8428100 10962473266
8   2002.09  438342395    3475560 18635461755
9   2002.12  528533759    8138020 12683418184
```

```
10  2003.03  429523269   14535990 14859884927
11  2003.06   64015168    5109500  7029384084
12  2003.09  336786062   11394917 20892617700
13  2003.12  438942853   14812754 15647266644
14  2004.03  315572853   10338300 17635564925
15  2004.06  758663210   21191288 15474196887
16  2004.09 1153100003    5783080 26788322244
17  2004.12 1034522413   54309865 19861453701
18  2005.03  641446421    5388960 23268279482
19  2005.06 1165252715   45028986 22965942213
20  2005.09  992627324   11348510 37673171188
21  2005.12 1372387152   62163640 27122123547
22  2006.03  920580625   13672836 39746045641
23  2006.06 2123930825  132961683 31913477123
24  2006.09 1928612977    2985880 48605117233
25  2006.12 2393203854  102367110 46035346970
26  2007.03 1239215427   35432682 49819661853
27  2007.06 2058854025   94954440 42425180001
28  2007.09 1741477478          0 60634299844
29  2007.12 1771196490   40011868 46418905817
30  2008.03 1432606264   10453124 57624282255
31  2008.06 2098126845   65203394 43908392041
32  2008.09 1398280115   15837118 43516939794
33  2008.12  727493963   -3735654 27729106510
34  2009.03  528828105   10678454 30625278205
35  2009.06 1005134062    4777258 29636267486
36  2009.09  679725152    5375226 34845539027
37  2009.12  528153534  -20830938 28792370983
38  2010.03 1055742790   44554342 48480421492
39  2010.06 2611994098   34151731 48451368536
40  2010.09 1723098483          0 66245202818
41  2010.12 2930265435   98244331 54941857566
42  2011.03 2449466000   36710000 63509496808
43  2011.06 2818383000   58809000 47554958058
44  2011.09 2357446000   27714000 66186330843
```

```
$테드 캡투 어
    구....분  광고선전비 교육훈련비       매출액
1   2000.03     1840000          0  1093168868
2   2000.06     1134000          0  1515376363
3   2000.09      930000    1246020  2130147815
4   2000.12    10009090          0  3973329930
```

5	2001.03	5440000	100000	1261010772
6	2001.06	4180000	110000	3054480728
7	2001.09	644372	0	1695942024
8	2001.12	NA	0	4096816401
9	2002.03	80000	50000	1843152110
10	2002.06	5863845	80000	1434683008
11	2002.09	160000	400000	767838212
12	2002.12	0	1320000	2202840022
13	2003.03	840000	180000	1018896498
14	2003.06	100000	80000	817451869
15	2003.09	0	130000	302535025
16	2003.12	9000	44000	204728129
17	2004.03	2000000	0	552025936
18	2004.06	3578412	240000	953800877
19	2004.09	0	892430	1187171653
20	2004.12	0	-97570	673807053
21	2005.03	840000	160000	1281797282
22	2005.06	0	0	1637418175
23	2005.09	550000	0	1219981048
24	2005.12	0	80000	604751948
25	2006.03	840000	80000	568053703
26	2006.06	550000	240000	605465938
27	2006.09	1980000	0	938502536
28	2006.12	1625000	120000	1270813542
29	2007.03	767663233	27479356	13912283508
30	2007.06	1043098560	62698927	14327194117
31	2007.09	1479171609	43398756	16370510221
32	2007.12	525165387	57918044	26673357429
33	2008.03	0	0	19926326125
34	2008.06	1826496649	96007272	20911613216
35	2008.09	2748341000	145562000	19269912551
36	2008.12	-1432312659	-59481092	18218384709
37	2009.03	215422000	5027000	20855159928
38	2009.06	181660000	54300000	20806465770
39	2009.09	503662000	30857000	20201843391
40	2009.12	22083000	52101000	21278162428
41	2010.03	285347000	3554000	26266456633
42	2010.06	349981000	30268000	33396639702
43	2010.09	377363000	50701000	28589391364
44	2010.12	279352000	94395000	29306664228
45	2011.03	396275000	34736000	35928440908
46	2011.06	385406000	47663000	35180189334

```
47  2011.09    279924000  101540000 33937697440
```

```
$모두투어
    구....분 광고선전비 교육훈련비       매출액
1   2005.03  240939162          0  8144659648
2   2005.06  463698765          0  8548149085
3   2005.09  548353360          0 12789229280
4   2005.12   55201184          0  9392450163
5   2006.03  720601983          0 14671603224
6   2006.06  868056093          0 12358657708
7   2006.09  986967530    2075000 19989393935
8   2006.12 1944355509   19036000 19364220706
9   2007.03 1603949325   21327441 22693131593
10  2007.06 1486981449   13137318 19369500867
11  2007.09 3423372271   53335300 29525164926
12  2007.12 1260295463   28193029 22763656306
13  2008.03 1204193071   13200204 26530405258
14  2008.06 2679972645   66144746 20939237126
15  2008.09 1460678899   34482480 23136419134
16  2008.12  701137900   12212590 12659502899
17  2009.03  544481614    4611600 13037790654
18  2009.06  420150066   18224856 14039046595
19  2009.09  586547017    6552350 17831640182
20  2009.12  461808648    5522500 16466995715
21  2010.03  627109367   11581130 25003802066
22  2010.06  896659514   29407378 26329123344
23  2010.09 1265995826   20890280 36466418562
24  2010.12 1176649774   16647180 29286304144
25  2011.03 1119329000   21106000 33856914381
26  2011.06 1262617000   23868000 25788504600
27  2011.09 1066938000   16640000 36339683324
```

```
$세중
    구....분 광고선전비 교육훈련비       매출액
1   2000.09  -14386068    3890650   910409285
2   2000.12  993896486     159218  2817571796
3   2001.03  127116000    6420000  1883837000
4   2001.06  469787614    4204553  1674407342
5   2001.09  405677848    9285253  2043272736
6   2001.12  120953120   19780923  1992872884
7   2002.03   86683346   10914207  1454849568
8   2002.06  206628774   24193922  1404032357
```

9	2002.09	60978586	4247382	1178532818
10	2002.12	78896343	1647320	1499189369
11	2003.03	32975680	1100000	1072460156
12	2003.06	44874732	860000	1584639526
13	2003.09	31769502	1365000	1219759186
14	2003.12	756978698	61904440	3380350787
15	2004.03	44584722	2977870	756603288
16	2004.06	163037461	4677680	1959745267
17	2004.09	166148967	2033485	1276020776
18	2004.12	635019271	2850530	3146868792
19	2005.03	53200074	900000	1096624195
20	2005.06	163908857	412000	1446127886
21	2005.09	70423589	589500	900739681
22	2005.12	105881433	819320	1724624667
23	2006.03	72483619	500208	3086832585
24	2006.06	95598931	680000	3699008908
25	2006.09	1061239746	5708314	18492449432
26	2006.12	2606931469	60229861	35565924672
27	2007.03	685084202	16190280	16958575156
28	2007.06	1279207876	29395300	19918091583
29	2007.09	1152432010	17369340	19269681940
30	2007.12	1349318078	-7782930	16195353771
31	2008.03	930137416	29007679	16463319852
32	2008.06	1006599592	1090739	18276554526
33	2008.09	267514376	9583390	19159107089
34	2008.12	131415228	24538080	18050882379
35	2009.03	128959936	1252642	13597706178
36	2009.06	164430118	5023976	15680114832
37	2009.09	56881950	6980850	16130093114
38	2009.12	76691949	7046329	15292534175
39	2010.03	92244120	15798800	14808015873
40	2010.06	190779700	23429324	19124988108
41	2010.09	48997402	-4759221	20282845926
42	2010.12	102439249	4092070	20409449651
43	2011.03	0	0	19384007562
44	2011.06	341933000	0	22377775711
45	2011.09	19735000	0	21832230367

$참좋은데져

	구....분	광고선전비	교육훈련비	매출액
1	2007.03	500000	0	2915134989
2	2007.06	2836412	0	6782580656

```
3   2007.09   67680500         NA  4926211503
4   2007.12   24490909         NA  3857238621
5   2008.03   70500000         NA  5790815204
6   2008.06   15638356         NA 11098677865
7   2008.09  332926664         NA 11705620840
8   2008.12  401777158         NA  6093075622
9   2009.03  505167621         NA 11135171990
10  2009.06  672752955         NA 14040480647
11  2009.09  563375028         NA 14608039919
12  2009.12  754611174         NA  7019677299
13  2010.03  663249619         NA  8464425371
14  2010.06  670122403         NA 13644390979
15  2010.09  734878924         NA 14717211161
16  2010.12  694355548         NA  7595046012
17  2011.03  773227000         NA 14640171827
18  2011.06  743366000         NA 17297683586
19  2011.09  610507000         NA 14351258818
```

```
$톳 데관광 개발
   구....분 광고선전비 교육훈련비      매출액
1   2006.03 1036504876    6881040  8989686669
2   2006.06 2045872542    5399250 10486867068
3   2006.09 2739658080   15348250 14941497865
4   2006.12 1254813126   21589250 12175786065
5   2007.03 1195205643   27700180 12366948305
6   2007.06 1309156992   26722010 11500679409
7   2007.09 1062194930   32002591 16877064858
8   2007.12  986694351   19226091 10845317936
9   2008.03  977711617   23188682 12954724212
10  2008.06  984506080   24090040 11001911869
11  2008.09 1158096243   36495000 10797072664
12  2008.12  486885406   25375633  6545406791
13  2009.03  477624164   19607350  5582468824
14  2009.06  569612024   20698260  7088618820
15  2009.09  600097366   28783950  7726065331
16  2009.12  546760303   18322830  5515137791
17  2010.03  660158348   28187000  7761111255
18  2010.06  613649450   17553344  8633822363
19  2010.09  609214711   30269450 12561340891
20  2010.12  665034444   18269420  9821126136
21  2011.03  607102000          0  9707491187
22  2011.06  670256000   58419000 11048769593
```

```
23  2011.09  796352000   27963000 13488179269
```

```
$자유투어
    구....분 광고선전비 교육훈련비       매출액
1   2001.09    3360000    5544980  4696875020
2   2001.12    1620000    2584050  8194414929
3   2002.03   27836700   24824070  2846210082
4   2002.06    7883500    5848337  5351832866
5   2002.09   -2487500    1392550  3061660200
6   2002.12   20300000    3866420  1726503925
7   2003.03   16939000   10214408  1189291101
8   2003.06    7540000    3859000  2233223000
9   2003.09    7364000    1955000  1014622000
10  2003.12    1799636    2518441  3207216649
11  2004.03    8724000    3903000  1802926000
12  2004.06    7234000     540000  1986166000
13  2004.09    4484000     581000  1066335000
14  2004.12    8747995     459883  5431549827
15  2005.03    3140000     160000   203880000
16  2005.06          0    2974000   464884000
17  2005.09    1035000     500000   357113000
18  2005.12  594003623     489750  2622344849
19  2006.03  844918000     400000  3761433000
20  2006.06 1159472000     184000  3316223000
21  2006.09 1640866000     720000  4112823000
22  2006.12 1147109888     914990  3657119429
23  2007.03 1167079621     180000  4505787439
24  2007.06 1193941000     100000  3820233000
25  2007.09 1105316438          0  5894269520
26  2007.12 1007781948     620000  4010276376
27  2008.03  975582113     100000  4803096281
28  2008.06 1043178000    2090000  4483386000
29  2008.09 1310815272    2480800  4250004693
30  2008.12  510565418    1347200  2469971423
31  2009.03  653303451    1066590  4823797414
32  2009.06  565423497          0  2740884428
33  2009.09  650179780    -706010  7022767720
34  2009.12  740856031     514920 11683666130
35  2010.03  817154204     516680 11317681906
36  2010.06  914617215     364020  6840916664
37  2010.09  932169708     852300 10455493093
38  2010.12 1020382617    1204920  5568666350
```

```
39  2011.03 1025380000     796000  8350833119
40  2011.06  906785000     820000  6310593776
41  2011.09 2857454000    1669000  5680437873
```

$비티앤아이

```
    구....분 광고선전비 교육훈련비      매출액
1   2001.03  119792000    8233359 2955754046
2   2001.06  271838219   12481915 2717441792
3   2001.09  113744005   13981556 2192595823
4   2001.12  125040916    8300270 2639250179
5   2002.03   92779137   13502440 1742604837
6   2002.06  253821435    8046800 2489897462
7   2002.09  114540480   10029658 2086142031
8   2002.12  115121935    5321615 2830037688
9   2003.03  106609957   11754857 1786877110
10  2003.06  160728146   14233740 2135968443
11  2003.09   86676615   17696670 1846643602
12  2003.12  180364176   14576780 2711030930
13  2004.03  107319947    7782450 1483498874
14  2004.06  178475843   16184590 2240846626
15  2004.09   92313337   19306137 2600847541
16  2004.12  209758381   14388830 2531327791
17  2005.03  184206999    6283925 2201535575
18  2005.06  220888403   12288850 2747664546
19  2005.09  145568323   13268710 2346511534
20  2005.12  263592370   15133596 2804331806
21  2006.03  130276877    8108340 1482134189
22  2006.06  193554491   11101620 1893886573
23  2006.09  119956546    6036140 1791486824
24  2006.12  286486447    3967950 2406356544
25  2007.03   65900692    3990710 1180081629
26  2007.06  207270104    6455710 1411557360
27  2007.09  133377642    8215550 1185888133
28  2007.12  206995961    5550920 1450245912
29  2008.03  106548846    2848200 1012965573
30  2008.06  230633746    6965970 1274014295
31  2008.09  205953497   13846645 3704152805
32  2008.12  375688451    3531000 3474731068
33  2009.03   81561788   13427260 2542768350
34  2009.06  122025637     260000 2489723263
35  2009.09  120160963     720000 2677270912
36  2009.12  173144605          0 3465555998
```

```
37  2010.03  116821788    2240000 2270390985
38  2010.06  209095000   25614000 3770621023
39  2010.09  148188000    6260000 4986588998
40  2010.12  165032212    5740000 1431440795
41  2011.03  180268000    1300000 5490062976
42  2011.06  198959000     598000 3262398180
43  2011.09  144049000     500000 1392998665

>
```

리스트로 불러들인 데이터를 하나로 합치기: 모든 데이터가 성공적으로 불러들여왔음을 확인할 수 있었습니다. 또한 모든 워크시트는 동일한 개수의 변수명 목록을 가지고 있으며, 이들은 모두 같은 변수명을 가집니다. 그런데, 이 데이터는 현재 리스트라는 데이터형식에 들어있습니다. 분석을 위해서는 데이터프레임에 하나로 통합된 데이터가 좋을 것입니다.

따라서, 아래와 같이 수행합니다.

```
> mydata <- do.call(rbind, tmp)
> head(mydata)
           구....분 광고선전비 교육훈련비       매출액
하나투어.1  2000.12  161702806   18002000  5616224889
하나투어.2  2001.03   80485618   28146500  7188763335
하나투어.3  2001.06  170827271   12965900  7948588645
하나투어.4  2001.09   65667863   26468000 11509839298
하나투어.5  2001.12   27804868   16838062  7799015935
하나투어.6  2002.03   81945640   12752112 10491229385
>
> tail(mydata)
              구....분 광고선전비 교육훈련비       매출액
비티앤아이.38  2010.06  209095000   25614000 3770621023
비티앤아이.39  2010.09  148188000    6260000 4986588998
비티앤아이.40  2010.12  165032212    5740000 1431440795
비티앤아이.41  2011.03  180268000    1300000 5490062976
비티앤아이.42  2011.06  198959000     598000 3262398180
비티앤아이.43  2011.09  144049000     500000 1392998665
>
```

이제서야 하나로 잘 정리된 데이터로 만들어졌습니다.

변수명 바꾸기 그런데, 첫번째 변수명이 원본데이터에서는 "구 분" 이라고 되어 있으나, 불러들인 데이터에서는 "구....분"이라고 되어 있습니다. 이는 XLConnect 패키지에서 변수명을 처리할때 빈공간 (화이트스페이스)를 ... 으로 대체했기 때문입니다. 점 하나가 스페이스 하나입니다. 그런데, 생각을 해보니 "구분" 이라는 변수명이 데이터를 표현하는데 적절하지 않은 것 같습니다. 이 변수의 값들은 날짜를 의미하기 때문에 "분기" 라고 변경하는 것이 더욱 적절할 것입니다.

그래서, 아래와 같이 변수명을 변경합니다.

```
> names(mydata)[1] <- c("년도별분기")
> names(mydata)
[1] "년도별분기" "광고선전비" "교육훈련비" "매출액"
>
```

데이터 구조확인: 이제 데이터의 구조를 살펴봅니다.

```
> str(mydata)
'data.frame':        289 obs. of  4 variables:
 $ 년도별분기: num  2000 2001 2001 2001 2001 ...
 $ 광고선전비: num  1.62e+08 8.05e+07 1.71e+08 6.57e+07 2.78e+07 ...
 $ 교육훈련비: num  18002000 28146500 12965900 26468000 16838062 ...
 $ 매출액    : num  5.62e+09 7.19e+09 7.95e+09 1.15e+10 7.80e+09 ...
>
```

총 289개의 관측치가 4개의 변수로부터 측정되었음을 확인할 수 있었습니다. 그런데, 데이터형이 data.frame입니다. 그 이유는 이전에 do.call()를 이용하여 한데 묶었기 때문입니다. 정말 데이터프레임일까요? 이전에 설명했듯이 데이터프레임은 리스트의 특수한 경우이기 때문입니다.

```
> is.data.frame(mydata)
[1] TRUE
> is.list(mydata)
[1] TRUE
```

중복을 확인하기: 그런데, 합쳐진 데이터 mydata를 다시 살펴보니 데이터가 어떤 워크시트로부터 몇 번째 데이터인지를 구분해주는 지시자가 없습니다. 이 지시자의 특징은 각 행별로 절대로 중복이 없는 유일한 값이어야 한다는 점입니다. 이것을 우리는 프라이머리키(primary key)라고 합니다. 그런데, mydata의 행이름을 보니, 이 정보를 포함하고 있습니다. 먼저, mydata의 행의 이름이 어떠한 중복이 있는지 확인해 봅니다.

```
> rownames(mydata)
  [1] "하나투어.1"      "하나투어.2"      "하나투어.3"      "하나투어.4"
  [5] "하나투어.5"      "하나투어.6"      "하나투어.7"      "하나투어.8"
  [9] "하나투어.9"      "하나투어.10"     "하나투어.11"     "하나투어.12"
 [13] "하나투어.13"     "하나투어.14"     "하나투어.15"     "하나투어.16"
 [17] "하나투어.17"     "하나투어.18"     "하나투어.19"     "하나투어.20"
 [21] "하나투어.21"     "하나투어.22"     "하나투어.23"     "하나투어.24"
 [25] "하나투어.25"     "하나투어.26"     "하나투어.27"     "하나투어.28"
 [29] "하나투어.29"     "하나투어.30"     "하나투어.31"     "하나투어.32"
 [33] "하나투어.33"     "하나투어.34"     "하나투어.35"     "하나투어.36"
 [37] "하나투어.37"     "하나투어.38"     "하나투어.39"     "하나투어.40"
 [41] "하나투어.41"     "하나투어.42"     "하나투어.43"     "하나투어.44"
 [45] "레드캡투어.1"    "레드캡투어.2"    "레드캡투어.3"    "레드캡투어.4"
```

```
 [49] "레드 캡투 어.5"    "레드 캡투 어.6"    "레드 캡투 어.7"    "레드 캡투 어.8"
 [53] "레드 캡투 어.9"    "레드 캡투 어.10"   "레드 캡투 어.11"   "레드 캡투 어.12"
 [57] "레드 캡투 어.13"   "레드 캡투 어.14"   "레드 캡투 어.15"   "레드 캡투 어.16"
 [61] "레드 캡투 어.17"   "레드 캡투 어.18"   "레드 캡투 어.19"   "레드 캡투 어.20"
 [65] "레드 캡투 어.21"   "레드 캡투 어.22"   "레드 캡투 어.23"   "레드 캡투 어.24"
 [69] "레드 캡투 어.25"   "레드 캡투 어.26"   "레드 캡투 어.27"   "레드 캡투 어.28"
 [73] "레드 캡투 어.29"   "레드 캡투 어.30"   "레드 캡투 어.31"   "레드 캡투 어.32"
 [77] "레드 캡투 어.33"   "레드 캡투 어.34"   "레드 캡투 어.35"   "레드 캡투 어.36"
 [81] "레드 캡투 어.37"   "레드 캡투 어.38"   "레드 캡투 어.39"   "레드 캡투 어.40"
 [85] "레드 캡투 어.41"   "레드 캡투 어.42"   "레드 캡투 어.43"   "레드 캡투 어.44"
 [89] "레드 캡투 어.45"   "레드 캡투 어.46"   "레드 캡투 어.47"   "모두투 어.1"
 [93] "모 두투 어.2"      "모 두투 어.3"      "모 두투 어.4"      "모 두투 어.5"
 [97] "모 두투 어.6"      "모 두투 어.7"      "모 두투 어.8"      "모 두투 어.9"
[101] "모 두투 어.10"     "모 두투 어.11"     "모 두투 어.12"     "모 두투 어.13"
[105] "모 두투 어.14"     "모 두투 어.15"     "모 두투 어.16"     "모 두투 어.17"
[109] "모 두투 어.18"     "모 두투 어.19"     "모 두투 어.20"     "모 두투 어.21"
[113] "모 두투 어.22"     "모 두투 어.23"     "모 두투 어.24"     "모 두투 어.25"
[117] "모 두투 어.26"     "모 두투 어.27"     "세중.1"            "세중.2"
[121] "세중.3"            "세중.4"            "세중.5"            "세중.6"
[125] "세중.7"            "세중.8"            "세중.9"            "세중.10"
[129] "세중.11"           "세중.12"           "세중.13"           "세중.14"
[133] "세중.15"           "세중.16"           "세중.17"           "세중.18"
[137] "세중.19"           "세중.20"           "세중.21"           "세중.22"
[141] "세중.23"           "세중.24"           "세중.25"           "세중.26"
[145] "세중.27"           "세중.28"           "세중.29"           "세중.30"
[149] "세중.31"           "세중.32"           "세중.33"           "세중.34"
[153] "세중.35"           "세중.36"           "세중.37"           "세중.38"
[157] "세중.39"           "세중.40"           "세중.41"           "세중.42"
[161] "세중.43"           "세중.44"           "세중.45"           "참좋은 레져.1"
[165] "참좋은 레져.2"     "참좋은 레져.3"     "참좋은 레져.4"     "참좋은 레져.5"
[169] "참좋은 레져.6"     "참좋은 레져.7"     "참좋은 레져.8"     "참좋은 레져.9"
[173] "참좋은 레져.10"    "참좋은 레져.11"    "참좋은 레져.12"    "참좋은 레져.13"
[177] "참좋은 레져.14"    "참좋은 레져.15"    "참좋은 레져.16"    "참좋은 레져.17"
[181] "참좋은 레져.18"    "참좋은 레져.19"    "툿데관광개발.1"    "툿데관광개발.2"
[185] "툿데관광개발.3"    "툿데관광개발.4"    "툿데관광개발.5"    "툿데관광개발.6"
[189] "툿데관광개발.7"    "툿데관광개발.8"    "툿데관광개발.9"    "툿데관광개발.10"
[193] "툿데관광개발.11"   "툿데관광개발.12"   "툿데관광개발.13"   "툿데관광개발.14"
[197] "툿데관광개발.15"   "툿데관광개발.16"   "툿데관광개발.17"   "툿데관광개발.18"
[201] "툿데관광개발.19"   "툿데관광개발.20"   "툿데관광개발.21"   "툿데관광개발.22"
[205] "툿데관광개발.23"   "자유투 어.1"       "자유투 어.2"       "자유투 어.3"
[209] "자유투 어.4"       "자유투 어.5"       "자유투 어.6"       "자유투 어.7"
[213] "자유투 어.8"       "자유투 어.9"       "자유투 어.10"      "자유투 어.11"
```

```
[217] "자유투어.12"     "자유투어.13"     "자유투어.14"     "자유투어.15"
[221] "자유투어.16"     "자유투어.17"     "자유투어.18"     "자유투어.19"
[225] "자유투어.20"     "자유투어.21"     "자유투어.22"     "자유투어.23"
[229] "자유투어.24"     "자유투어.25"     "자유투어.26"     "자유투어.27"
[233] "자유투어.28"     "자유투어.29"     "자유투어.30"     "자유투어.31"
[237] "자유투어.32"     "자유투어.33"     "자유투어.34"     "자유투어.35"
[241] "자유투어.36"     "자유투어.37"     "자유투어.38"     "자유투어.39"
[245] "자유투어.40"     "자유투어.41"     "비티앤아이.1"    "비티앤아이.2"
[249] "비티앤아이.3"    "비티앤아이.4"    "비티앤아이.5"    "비티앤아이.6"
[253] "비티앤아이.7"    "비티앤아이.8"    "비티앤아이.9"    "비티앤아이.10"
[257] "비티앤아이.11"   "비티앤아이.12"   "비티앤아이.13"   "비티앤아이.14"
[261] "비티앤아이.15"   "비티앤아이.16"   "비티앤아이.17"   "비티앤아이.18"
[265] "비티앤아이.19"   "비티앤아이.20"   "비티앤아이.21"   "비티앤아이.22"
[269] "비티앤아이.23"   "비티앤아이.24"   "비티앤아이.25"   "비티앤아이.26"
[273] "비티앤아이.27"   "비티앤아이.28"   "비티앤아이.29"   "비티앤아이.30"
[277] "비티앤아이.31"   "비티앤아이.32"   "비티앤아이.33"   "비티앤아이.34"
[281] "비티앤아이.35"   "비티앤아이.36"   "비티앤아이.37"   "비티앤아이.38"
[285] "비티앤아이.39"   "비티앤아이.40"   "비티앤아이.41"   "비티앤아이.42"
[289] "비티앤아이.43"
```

그런데 이렇게 일일이 눈으로는 확인할 수 없지 않겠나... 하는 생각이 불현듯 떠오릅니다. 그래서 중복이 있고 없고를 한번에 알 수 있는 길이 없을까 하는 생각을 합니다.

```
> all(!duplicated(rownames(mydata)))
[1] TRUE
```

duplicated()라는 함수는 중복을 체크하여 TRUE 또는 FALSE를 알려줍니다. !(느낌표)는 반대라는 의미를 나타내는 연산자입니다. 즉, !duplicated()란 중복이 없나요? 를 물어보는 것입니다. 그리고 all()이라는 함수는 벡터내에 있는 값이 모두 TRUE인지를 확인해줍니다. 이 결과가 TRUE이므로 행의 이름이 중복이 되지 않았음을 확인하였습니다. 따라서, 이 정보를 프라이머리 키로 사용해도 될 것 같습니다.

문자열를 주어진 문자를 이용하여 분리하기: 그런데 생각을 해보니 여행사별로 분석을 수행할 수 있는데 이를 구분해 줄 수 있는 변수가 없습니다. 따라서, "여행사"라는 변수를 새로 만들어 mydata 데이터셋에 넣고자 합니다. 이를 수행하기 위해서는 strpsplit() 함수를 이용하여 아래와 같이 행이름의 문자열을 어떤 특수한 문자에 의해서 나누어 주는 것입니다.

```
> head(strsplit(rownames(mydata), ".", fixed=TRUE))
[[1]]
[1] "하나투어" "1"

[[2]]
[1] "하나투어" "2"
```

```
[[3]]
[1] "하나투어" "3"

[[4]]
[1] "하나투어" "4"

[[5]]
[1] "하나투어" "5"

[[6]]
[1] "하나투어" "6"

>
```

그리고, 이렇게 리스트로 쪼개어진 변수명을 do.call()함수를 이용하여 행렬의 형태로 재조합한 것을 활용하는 것입니다.

```
> head(do.call(rbind, strsplit(rownames(mydata), ".", fixed=TRUE)))
     [,1]       [,2]
[1,] "하나투어" "1"
[2,] "하나투어" "2"
[3,] "하나투어" "3"
[4,] "하나투어" "4"
[5,] "하나투어" "5"
[6,] "하나투어" "6"
>
```

데이터프레임에 변수 추가하기 그리고 여행사라는 변수를 생성합니다. 행이름은 더이상 필요하지 않으므로 삭제합니다.

```
> mydata$"여행사" <- do.call(rbind, strsplit(rownames(mydata), ".", fixed=TRUE))[,1]
> mydata$"번호" <- do.call(rbind, strsplit(rownames(mydata), ".", fixed=TRUE))[,2]
> rownames(mydata) <- NULL
> head(mydata)
  년도별분기 광고선전비 교육훈련비       매출액   여행사 번호
1    2000.12  161702806   18002000  5616224889 하나투어    1
2    2001.03   80485618   28146500  7188763335 하나투어    2
3    2001.06  170827271   12965900  7948588645 하나투어    3
4    2001.09   65667863   26468000 11509839298 하나투어    4
5    2001.12   27804868   16838062  7799015935 하나투어    5
6    2002.03   81945640   12752112 10491229385 하나투어    6
```

이러한 방법으로 년도별 분기 변수를 좀 더 상세화 할 수 있을 것입니다.

```
> yrQ <- as.data.frame(do.call(rbind, strsplit(as.character(mydata$"년도별분기"), ".", fixed=TRUE)))
> names(yrQ) <- c("년도", "월")
> mydata <- data.frame(mydata, yrQ)
> head(mydata)
  년도별분기 광고선전비 교육훈련비      매출액   여행사 번호 년도 월
1    2000.12  161702806   18002000  5616224889 하나투어    1 2000 12
2    2001.03   80485618   28146500  7188763335 하나투어    2 2001 03
3    2001.06  170827271   12965900  7948588645 하나투어    3 2001 06
4    2001.09   65667863   26468000 11509839298 하나투어    4 2001 09
5    2001.12   27804868   16838062  7799015935 하나투어    5 2001 12
6    2002.03   81945640   12752112 10491229385 하나투어    6 2002 03
```

결측치 확인하고 제거하기 그런데, 데이터에 결측치들이 얼마나 있는지 살펴보아야 할 것입니다. 만약 있다면 어디에서 어떤 변수에서 결측치가 있으며, 이들을 삭제할 것인지 결정해야 합니다. 그래서 원본데이터 tmp를 살펴보았더니, 아래와 같이 NA 가 있습니다.

```
> tmp$"참좋은 데져"
   구....분 광고선전비 교육훈련비      매출액
1   2007.03     500000          0  2915134989
2   2007.06    2836412          0  6782580656
3   2007.09   67680500         NA  4926211503
4   2007.12   24490909         NA  3857238621
5   2008.03   70500000         NA  5790815204
6   2008.06   15638356         NA 11098677865
7   2008.09  332926664         NA 11705620840
8   2008.12  401777158         NA  6093075622
9   2009.03  505167621         NA 11135171990
10  2009.06  672752955         NA 14040480647
11  2009.09  563375028         NA 14608039919
12  2009.12  754611174         NA  7019677299
13  2010.03  663249619         NA  8464425371
14  2010.06  670122403         NA 13644390979
15  2010.09  734878924         NA 14717211161
16  2010.12  694355548         NA  7595046012
17  2011.03  773227000         NA 14640171827
18  2011.06  743366000         NA 17297683586
19  2011.09  610507000         NA 14351258818
>
```

그럼 하나로 뭉친 mydata 파일에서 어떻게 이러한 데이터를 찾아야 할까요? is.na() 함수의 사용은 아래와 같은 결과를 줍니다.

```
> head(is.na(mydata))
     년도 별분 기 광고선전비 교육훈련비 매출액 여행사 번호 년도 월
[1,]       FALSE      FALSE      FALSE  FALSE  FALSE FALSE FALSE FALSE
[2,]       FALSE      FALSE      FALSE  FALSE  FALSE FALSE FALSE FALSE
[3,]       FALSE      FALSE      FALSE  FALSE  FALSE FALSE FALSE FALSE
[4,]       FALSE      FALSE      FALSE  FALSE  FALSE FALSE FALSE FALSE
[5,]       FALSE      FALSE      FALSE  FALSE  FALSE FALSE FALSE FALSE
[6,]       FALSE      FALSE      FALSE  FALSE  FALSE FALSE FALSE FALSE
>
```

그렇다면 TRUE 라고 된 부분이 결측일 것입니다. 데이터를 한 눈에 살펴볼 수 없기 때문에 아래와 같이 합니다.

```
> idx <- which(is.na(mydata))
> mydata[idx%%nrow(mydata), ]
    년도 별분 기 광고선전비 교육훈련비       매출액      여행사 번호 년도 월
52     2001.12         NA          0  4096816401 레드캡투어     8 2001 12
166    2007.09   67680500         NA  4926211503 참좋은여행     3 2007 09
167    2007.12   24490909         NA  3857238621 참좋은여행     4 2007 12
168    2008.03   70500000         NA  5790815204 참좋은여행     5 2008 03
169    2008.06   15638356         NA 11098677865 참좋은여행     6 2008 06
170    2008.09  332926664         NA 11705620840 참좋은여행     7 2008 09
171    2008.12  401777158         NA  6093075622 참좋은여행     8 2008 12
172    2009.03  505167621         NA 11135171990 참좋은여행     9 2009 03
173    2009.06  672752955         NA 14040480647 참좋은여행    10 2009 06
174    2009.09  563375028         NA 14608039919 참좋은여행    11 2009 09
175    2009.12  754611174         NA  7019677299 참좋은여행    12 2009 12
176    2010.03  663249619         NA  8464425371 참좋은여행    13 2010 03
177    2010.06  670122403         NA 13644390979 참좋은여행    14 2010 06
178    2010.09  734878924         NA 14717211161 참좋은여행    15 2010 09
179    2010.12  694355548         NA  7595046012 참좋은여행    16 2010 12
180    2011.03  773227000         NA 14640171827 참좋은여행    17 2011 03
181    2011.06  743366000         NA 17297683586 참좋은여행    18 2011 06
182    2011.09  610507000         NA 14351258818 참좋은여행    19 2011 09
```

이와 같은 논리를 이용하여 R은 결측치에 해당하는 레코드를 지워주는 na.exclude()라는 함수를 제공합니다.

```
> mydatax <- na.exclude(mydata)
> mydatax[163:170, ]
    년도 별분 기 광고선전비 교육훈련비      매출액      여행사 번호 년도 월
164    2007.03     500000          0  2915134989 참좋은여행     1 2007 03
165    2007.06    2836412          0  6782580656 참좋은여행     2 2007 06
```

```
183    2006.03 1036504876    6881040  8989686669 톳데관광개발    1 2006 03
184    2006.06 2045872542    5399250 10486867068 톳데관광개발    2 2006 06
185    2006.09 2739658080   15348250 14941497865 톳데관광개발    3 2006 09
186    2006.12 1254813126   21589250 12175786065 톳데관광개발    4 2006 12
187    2007.03 1195205643   27700180 12366948305 톳데관광개발    5 2007 03
188    2007.06 1309156992   26722010 11500679409 톳데관광개발    6 2007 06
>
```

데이터프레임에서 변수삭제하기: "년도"와 "월"이라는 변수를 따로 생성하였기 때문에 이제 "년도별분기"라는 변수는 불필요하므로 변수를 삭제하도록 합니다.

```
> mydatax <- mydatax[c(FALSE, rep(TRUE, 7))]
> head(mydatax)
  광고선전비 교육훈련비       매출액   여행사 번호 년도 월
1  161702806   18002000  5616224889 하나투어    1 2000 12
2   80485618   28146500  7188763335 하나투어    2 2001 03
3  170827271   12965900  7948588645 하나투어    3 2001 06
4   65667863   26468000 11509839298 하나투어    4 2001 09
5   27804868   16838062  7799015935 하나투어    5 2001 12
6   81945640   12752112 10491229385 하나투어    6 2002 03
>
```

그러고 보니, "월"이라는 변수는 분기별로 데이터를 모은 것이므로 "분기"로 변형하는 것이 좋을 듯 합니다. 먼저, "월"이라는 변수가 정말 3,6,9,12 월에 해당하는 값들만 가지고 있는지 확인을 해야할 것입니다.

```
> names(table(mydatax$"월"))
[1] "03" "06" "09" "12"
```

따라서, "월"이라는 변수를 "분기"라는 변수로 변경합니다. 또한, 문자형을 요인형으로 변경하면서, 수준에 따라 라벨링을 함께 합니다.

```
> mydatax$"월" <- factor(mydatax$"월", levels=c("03", "06", "09", "12"), labels=c("1분기", "2분기", "3분
> names(mydatax)[7] <- c("분기")
> head(mydatax)
  광고선전비 교육훈련비       매출액   여행사 번호 년도  분기
1  161702806   18002000  5616224889 하나투어    1 2000 4분기
2   80485618   28146500  7188763335 하나투어    2 2001 1분기
3  170827271   12965900  7948588645 하나투어    3 2001 2분기
4   65667863   26468000 11509839298 하나투어    4 2001 3분기
5   27804868   16838062  7799015935 하나투어    5 2001 4분기
6   81945640   12752112 10491229385 하나투어    6 2002 1분기
>
```

분할표 생성해보기: 이제 간단한 분기와 년도에 따른 contingency table을 생성해봅니다.

```
> ftable(mydatax$"분 기", mydatax$"년 도")
       2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011

1분 기     1    4    5    5    5    6    7    8    7    7    7    7
2분 기     1    4    5    5    5    6    7    8    7    7    7    7
3분 기     2    5    5    5    5    6    7    7    7    7    7    7
4분 기     3    4    5    5    5    6    7    7    7    7    7    0
>
```

이 분할표를 여행사별로 출력해봅니다.

```
> ftable(mydatax$"여 행 사", mydatax$"분 기", mydatax$"년 도")
                   2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011

세중          1분 기    0    1    1    1    1    1    1    1    1    1    1    1
              2분 기    0    1    1    1    1    1    1    1    1    1    1    1
              3분 기    1    1    1    1    1    1    1    1    1    1    1    1
              4분 기    1    1    1    1    1    1    1    1    1    1    1    0
하나투 어     1분 기    0    1    1    1    1    1    1    1    1    1    1    1
              2분 기    0    1    1    1    1    1    1    1    1    1    1    1
              3분 기    0    1    1    1    1    1    1    1    1    1    1    1
              4분 기    1    1    1    1    1    1    1    1    1    1    1    0
모 두투 어    1분 기    0    0    0    0    0    1    1    1    1    1    1    1
              2분 기    0    0    0    0    0    1    1    1    1    1    1    1
              3분 기    0    0    0    0    0    1    1    1    1    1    1    1
              4분 기    0    0    0    0    0    1    1    1    1    1    1    0
자유투 어     1분 기    0    0    1    1    1    1    1    1    1    1    1    1
              2분 기    0    0    1    1    1    1    1    1    1    1    1    1
              3분 기    0    1    1    1    1    1    1    1    1    1    1    1
              4분 기    0    1    1    1    1    1    1    1    1    1    1    0
레드 캡투 어  1분 기    1    1    1    1    1    1    1    1    1    1    1    1
              2분 기    1    1    1    1    1    1    1    1    1    1    1    1
              3분 기    1    1    1    1    1    1    1    1    1    1    1    1
              4분 기    1    0    1    1    1    1    1    1    1    1    1    0
참좋은 레저   1분 기    0    0    0    0    0    0    0    1    0    0    0    0
              2분 기    0    0    0    0    0    0    0    1    0    0    0    0
              3분 기    0    0    0    0    0    0    0    0    0    0    0    0
              4분 기    0    0    0    0    0    0    0    0    0    0    0    0
비티앤아이    1분 기    0    1    1    1    1    1    1    1    1    1    1    1
              2분 기    0    1    1    1    1    1    1    1    1    1    1    1
              3분 기    0    1    1    1    1    1    1    1    1    1    1    1
```

```
              4분기    0    1    1    1    1    1    1    1    1    1    1    0
툇데관광개발 1분기    0    0    0    0    0    0    1    1    1    1    1    1
              2분기    0    0    0    0    0    0    1    1    1    1    1    1
              3분기    0    0    0    0    0    0    1    1    1    1    1    1
              4분기    0    0    0    0    0    0    1    1    1    1    1    0
>
```

매번 데이터셋이름을 같이 쓰기가 너무 불편합니다. 따라서, 아래와 같이 with()를 사용해봅니다.

```
> with(mydatax, ftable(분기, 년도))
      년도 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011
분기
1분기         1    4    5    5    5    6    7    8    7    7    7    7
2분기         1    4    5    5    5    6    7    8    7    7    7    7
3분기         2    5    5    5    5    6    7    7    7    7    7    7
4분기         3    4    5    5    5    6    7    7    7    7    7    0
>
```

데이터의 선택적 부분지정: 이제 "하나투어"에 해당하는 자료를 뽑고, 그 중에서도 "4분기"에 해당하는 레코드를 뽑고자 합니다.

```
> subset(x=mydatax, subset=(여행사=="하나투어" & 분기=="4분기"))
   광고선전비 교육훈련비       매출액 여행사 번호 년도  분기
1   161702806   18002000  5616224889 하나투어    1 2000 4분기
5    27804868   16838062  7799015935 하나투어    5 2001 4분기
9   528533759    8138020 12683418184 하나투어    9 2002 4분기
13  438942853   14812754 15647266644 하나투어   13 2003 4분기
17 1034522413   54309865 19861453701 하나투어   17 2004 4분기
21 1372387152   62163640 27122123547 하나투어   21 2005 4분기
25 2393203854  102367110 46035346970 하나투어   25 2006 4분기
29 1771196490   40011868 46418905817 하나투어   29 2007 4분기
33  727493963   -3735654 27729106510 하나투어   33 2008 4분기
37  528153534  -20830938 28792370983 하나투어   37 2009 4분기
41 2930265435   98244331 54941857566 하나투어   41 2010 4분기
>
```

위에서 조건에 맞는 레코드들을 추출했지만, 변수가 모두 필요한 것은 아닙니다. 따라서, 교육훈련비, 년도, 분기 세가지 변수만 뽑아봅니다.

```
> subset(x=mydatax, subset=(여행사=="하나투어" & 분기=="4분기"), select=c(교육훈련비, 년도, 분기))
   교육훈련비 년도  분기
1    18002000 2000 4분기
5    16838062 2001 4분기
```

```
9     8138020 2002 4분기
13   14812754 2003 4분기
17   54309865 2004 4분기
21   62163640 2005 4분기
25  102367110 2006 4분기
29   40011868 2007 4분기
33   -3735654 2008 4분기
37  -20830938 2009 4분기
41   98244331 2010 4분기
>
```

그룹별 연산하기: 평균교육훈련비를 연도별로 산출한 뒤, 연도별 분할표를 생성해봅니다.

```
> as.data.frame(as.table(with(mydatax, tapply(교육훈련비, 년도, mean))))
   Var1     Freq
1  2000  3328270
2  2001 10320313
3  2002  7423926
4  2003  9416275
5  2004  8417142
6  2005  7416239
7  2006 15046704
8  2007 22127430
9  2008 21516616
10 2009 10721259
11 2010 21786660
12 2011 21945286
>
```

연산자 활용: 여기에서부터는 2008, 2009, 2010 년에서 1분기와 3분기에 해당하는 "다트8"이라는 데이터를 생성하여 작업하도록 하겠습니다. 그 이유는 단순히 결과를 효과적으로 보여주기 위해서입니다.

```
다트8 <- subset(mydatax, subset=(년도 %in% c("2008", "2009", "2010") & 분기 %in% c("1분기", "3분기")))
다트8
```

```
> 다트8
    광고선전비 교육훈련비      매출액      여행사 번호 년도   분기
30 1432606264  10453124 57624282255 하나투어   30 2008 1분기
32 1398280115  15837118 43516939794 하나투어   32 2008 3분기
34  528828105  10678454 30625278205 하나투어   34 2009 1분기
36  679725152   5375226 34845539027 하나투어   36 2009 3분기
38 1055742790  44554342 48480421492 하나투어   38 2010 1분기
40 1723098483         0 66245202818 하나투어   40 2010 3분기
```

77	0	0	19926326125	레드 캡투 어	33	2008	1분 기
79	2748341000	145562000	19269912551	레드 캡투 어	35	2008	3분 기
81	215422000	5027000	20855159928	레드 캡투 어	37	2009	1분 기
83	503662000	30857000	20201843391	레드 캡투 어	39	2009	3분 기
85	285347000	3554000	26266456633	레드 캡투 어	41	2010	1분 기
87	377363000	50701000	28589391364	레드 캡투 어	43	2010	3분 기
104	1204193071	13200204	26530405258	모두투어	13	2008	1분 기
106	1460678899	34482480	23136419134	모두투어	15	2008	3분 기
108	544481614	4611600	13037790654	모두투어	17	2009	1분 기
110	586547017	6552350	17831640182	모두투어	19	2009	3분 기
112	627109367	11581130	25003802066	모두투어	21	2010	1분 기
114	1265995826	20890280	36466418562	모두투어	23	2010	3분 기
149	930137416	29007679	16463319852	세중	31	2008	1분 기
151	267514376	9583390	19159107089	세중	33	2008	3분 기
153	128959936	1252642	13597706178	세중	35	2009	1분 기
155	56881950	6980850	16130093114	세중	37	2009	3분 기
157	92244120	15798800	14808015873	세중	39	2010	1분 기
159	48997402	-4759221	20282845926	세중	41	2010	3분 기
168	70500000	NA	5790815204	참좋은여행	5	2008	1분 기
170	332926664	NA	11705620840	참좋은여행	7	2008	3분 기
172	505167621	NA	11135171990	참좋은여행	9	2009	1분 기
174	563375028	NA	14608039919	참좋은여행	11	2009	3분 기
176	663249619	NA	8464425371	참좋은여행	13	2010	1분 기
178	734878924	NA	14717211161	참좋은여행	15	2010	3분 기
191	977711617	23188682	12954724212	투데관광개발	9	2008	1분 기
193	1158096243	36495000	10797072664	투데관광개발	11	2008	3분 기
195	477624164	19607350	5582468824	투데관광개발	13	2009	1분 기
197	600097366	28783950	7726065331	투데관광개발	15	2009	3분 기
199	660158348	28187000	7761111255	투데관광개발	17	2010	1분 기
201	609214711	30269450	12561340891	투데관광개발	19	2010	3분 기
232	975582113	100000	4803096281	자유투어	27	2008	1분 기
234	1310815272	2480800	4250004693	자유투어	29	2008	3분 기
236	653303451	1066590	4823797414	자유투어	31	2009	1분 기
238	650179780	-706010	7022767720	자유투어	33	2009	3분 기
240	817154204	516680	11317681906	자유투어	35	2010	1분 기
242	932169708	852300	10455493093	자유투어	37	2010	3분 기
275	106548846	2848200	1012965573	비티앤아이	29	2008	1분 기
277	205953497	13846645	3704152805	비티앤아이	31	2008	3분 기
279	81561788	13427260	2542768350	비티앤아이	33	2009	1분 기
281	120160963	720000	2677270912	비티앤아이	35	2009	3분 기
283	116821788	2240000	2270390985	비티앤아이	37	2010	1분 기
285	148188000	6260000	4986588998	비티앤아이	39	2010	3분 기

>

현재 여행사는 데이터를 그룹화 할 수 있는 고유한 키라고 할 수 있습니다.

정렬하기: 그런데, 이 데이터에 특징이 하나 있다면 그것은 동일한 여행사로부터 매년 분기별로 여러번 반복하여 얻은 데이터라는 것입니다. 따라서, 간혹 데이터를 여행사별로 정렬하기보다는 년도별로 정렬하고 싶을 경우가 있습니다. 이런 경우는 아래와 같이 합니다.

```
다트8[order(다트8$년도),]
```

	광고선전비	교육훈련비	매출액	여행사	번호	년도	분기
30	1432606264	10453124	57624282255	하나투어	30	2008	1분기
32	1398280115	15837118	43516939794	하나투어	32	2008	3분기
77	0	0	19926326125	레드캡투어	33	2008	1분기
79	2748341000	145562000	19269912551	레드캡투어	35	2008	3분기
104	1204193071	13200204	26530405258	모두투어	13	2008	1분기
106	1460678899	34482480	23136419134	모두투어	15	2008	3분기
149	930137416	29007679	16463319852	세중	31	2008	1분기
151	267514376	9583390	19159107089	세중	33	2008	3분기
168	70500000	NA	5790815204	참좋은레져	5	2008	1분기
170	332926664	NA	11705620840	참좋은레져	7	2008	3분기
191	977711617	23188682	12954724212	툿데관광개발	9	2008	1분기
193	1158096243	36495000	10797072664	툿데관광개발	11	2008	3분기
232	975582113	100000	4803096281	자유투어	27	2008	1분기
234	1310815272	2480800	4250004693	자유투어	29	2008	3분기
275	106548846	2848200	1012965573	비티앤아이	29	2008	1분기
277	205953497	13846645	3704152805	비티앤아이	31	2008	3분기
34	528828105	10678454	30625278205	하나투어	34	2009	1분기
36	679725152	5375226	34845539027	하나투어	36	2009	3분기
81	215422000	5027000	20855159928	레드캡투어	37	2009	1분기
83	503662000	30857000	20201843391	레드캡투어	39	2009	3분기
108	544481614	4611600	13037790654	모두투어	17	2009	1분기
110	586547017	6552350	17831640182	모두투어	19	2009	3분기
153	128959936	1252642	13597706178	세중	35	2009	1분기
155	56881950	6980850	16130093114	세중	37	2009	3분기
172	505167621	NA	11135171990	참좋은레져	9	2009	1분기
174	563375028	NA	14608039919	참좋은레져	11	2009	3분기
195	477624164	19607350	5582468824	툿데관광개발	13	2009	1분기
197	600097366	28783950	7726065331	툿데관광개발	15	2009	3분기
236	653303451	1066590	4823797414	자유투어	31	2009	1분기
238	650179780	-706010	7022767720	자유투어	33	2009	3분기
279	81561788	13427260	2542768350	비티앤아이	33	2009	1분기
281	120160963	720000	2677270912	비티앤아이	35	2009	3분기

```
38  1055742790   44554342 48480421492     하나투어   38 2010 1분기
40  1723098483          0 66245202818     하나투어   40 2010 3분기
85   285347000    3554000 26266456633   레드캡투어   41 2010 1분기
87   377363000   50701000 28589391364   레드캡투어   43 2010 3분기
112  627109367   11581130 25003802066     모두투어   21 2010 1분기
114 1265995826   20890280 36466418562     모두투어   23 2010 3분기
157   92244120   15798800 14808015873         세중   39 2010 1분기
159   48997402   -4759221 20282845926         세중   41 2010 3분기
176  663249619         NA  8464425371   참좋은여적   13 2010 1분기
178  734878924         NA 14717211161   참좋은여적   15 2010 3분기
199  660158348   28187000  7761111255 롯데관광개발   17 2010 1분기
201  609214711   30269450 12561340891 롯데관광개발   19 2010 3분기
240  817154204     516680 11317681906     자유투어   35 2010 1분기
242  932169708     852300 10455493093     자유투어   37 2010 3분기
283  116821788    2240000  2270390985   비티앤아이   37 2010 1분기
285  148188000    6260000  4986588998   비티앤아이   39 2010 3분기
>
```

중복되는 데이터의 처음과 끝 확인하기 또 다른 경우는 각 년도별로 첫번째 레코드가 무엇인지 마지막 레코드가 무엇인지 알고 싶을 경우가 있습니다. 이런 경우는 아래와 같이 할 수 있습니다.

```
다트8.1 <- 다트8[order(다트8$년도),]
다트8.1$first <- !duplicated(다트8.1$년도)
다트8.1
```

```
    광고선전비 교육훈련비       매출액       여행사 번호 년도  분기 first
30  1432606264   10453124 57624282255     하나투어   30 2008 1분기  TRUE
32  1398280115   15837118 43516939794     하나투어   32 2008 3분기 FALSE
77           0          0 19926326125   레드캡투어   33 2008 1분기 FALSE
79  2748341000  145562000 19269912551   레드캡투어   35 2008 3분기 FALSE
104 1204193071   13200204 26530405258     모두투어   13 2008 1분기 FALSE
106 1460678899   34482480 23136419134     모두투어   15 2008 3분기 FALSE
149  930137416   29007679 16463319852         세중   31 2008 1분기 FALSE
151  267514376    9583390 19159107089         세중   33 2008 3분기 FALSE
168   70500000         NA  5790815204   참좋은여적    5 2008 1분기 FALSE
170  332926664         NA 11705620840   참좋은여적    7 2008 3분기 FALSE
191  977711617   23188682 12954724212 롯데관광개발    9 2008 1분기 FALSE
193 1158096243   36495000 10797072664 롯데관광개발   11 2008 3분기 FALSE
232  975582113     100000  4803096281     자유투어   27 2008 1분기 FALSE
234 1310815272    2480800  4250004693     자유투어   29 2008 3분기 FALSE
275  106548846    2848200  1012965573   비티앤아이   29 2008 1분기 FALSE
277  205953497   13846645  3704152805   비티앤아이   31 2008 3분기 FALSE
```

```
34   528828105   10678454 30625278205      하나투어   34 2009 1분기  TRUE
36   679725152    5375226 34845539027      하나투어   36 2009 3분기 FALSE
81   215422000    5027000 20855159928    레드캡투어   37 2009 1분기 FALSE
83   503662000   30857000 20201843391    레드캡투어   39 2009 3분기 FALSE
108  544481614    4611600 13037790654      모두투어   17 2009 1분기 FALSE
110  586547017    6552350 17831640182      모두투어   19 2009 3분기 FALSE
153  128959936    1252642 13597706178          세중   35 2009 1분기 FALSE
155   56881950    6980850 16130093114          세중   37 2009 3분기 FALSE
172  505167621         NA 11135171990   참좋은여정    9 2009 1분기 FALSE
174  563375028         NA 14608039919   참좋은여정   11 2009 3분기 FALSE
195  477624164   19607350  5582468824 툿데관광개발   13 2009 1분기 FALSE
197  600097366   28783950  7726065331 툿데관광개발   15 2009 3분기 FALSE
236  653303451    1066590  4823797414      자유투어   31 2009 1분기 FALSE
238  650179780    -706010  7022767720      자유투어   33 2009 3분기 FALSE
279   81561788   13427260  2542768350    비티앤아이   33 2009 1분기 FALSE
281  120160963     720000  2677270912    비티앤아이   35 2009 3분기 FALSE
38  1055742790   44554342 48480421492      하나투어   38 2010 1분기  TRUE
40  1723098483          0 66245202818      하나투어   40 2010 3분기 FALSE
85   285347000    3554000 26266456633    레드캡투어   41 2010 1분기 FALSE
87   377363000   50701000 28589391364    레드캡투어   43 2010 3분기 FALSE
112  627109367   11581130 25003802066      모두투어   21 2010 1분기 FALSE
114 1265995826   20890280 36466418562      모두투어   23 2010 3분기 FALSE
157   92244120   15798800 14808015873          세중   39 2010 1분기 FALSE
159   48997402   -4759221 20282845926          세중   41 2010 3분기 FALSE
176  663249619         NA  8464425371   참좋은여정   13 2010 1분기 FALSE
178  734878924         NA 14717211161   참좋은여정   15 2010 3분기 FALSE
199  660158348   28187000  7761111255 툿데관광개발   17 2010 1분기 FALSE
201  609214711   30269450 12561340891 툿데관광개발   19 2010 3분기 FALSE
240  817154204     516680 11317681906      자유투어   35 2010 1분기 FALSE
242  932169708     852300 10455493093      자유투어   37 2010 3분기 FALSE
283  116821788    2240000  2270390985    비티앤아이   37 2010 1분기 FALSE
285  148188000    6260000  4986588998    비티앤아이   39 2010 3분기 FALSE
>
```

이와 유사한 논리로 각 년도별 마지막 레코드를 활용하고자 하는 지시자를 생성할 수도 있습니다.

```
다트8.1$last <- !duplicated(다트8.1$년도, fromLast=TRUE)
다트8.1
```

```
    광고선전비 교육훈련비       매출액       여행사 번호 년도  분기 first  last
30  1432606264   10453124 57624282255      하나투어   30 2008 1분기  TRUE FALSE
32  1398280115   15837118 43516939794      하나투어   32 2008 3분기 FALSE FALSE
77           0          0 19926326125    레드캡투어   33 2008 1분기 FALSE FALSE
```

79	2748341000	145562000	19269912551	레드캡투어	35	2008	3분기	FALSE	FALSE
104	1204193071	13200204	26530405258	모두투어	13	2008	1분기	FALSE	FALSE
106	1460678899	34482480	23136419134	모두투어	15	2008	3분기	FALSE	FALSE
149	930137416	29007679	16463319852	세중	31	2008	1분기	FALSE	FALSE
151	267514376	9583390	19159107089	세중	33	2008	3분기	FALSE	FALSE
168	70500000	NA	5790815204	참좋은레져	5	2008	1분기	FALSE	FALSE
170	332926664	NA	11705620840	참좋은레져	7	2008	3분기	FALSE	FALSE
191	977711617	23188682	12954724212	툿데관광개발	9	2008	1분기	FALSE	FALSE
193	1158096243	36495000	10797072664	툿데관광개발	11	2008	3분기	FALSE	FALSE
232	975582113	100000	4803096281	자유투어	27	2008	1분기	FALSE	FALSE
234	1310815272	2480800	4250004693	자유투어	29	2008	3분기	FALSE	FALSE
275	106548846	2848200	1012965573	비티앤아이	29	2008	1분기	FALSE	FALSE
277	205953497	13846645	3704152805	비티앤아이	31	2008	3분기	FALSE	TRUE
34	528828105	10678454	30625278205	하나투어	34	2009	1분기	TRUE	FALSE
36	679725152	5375226	34845539027	하나투어	36	2009	3분기	FALSE	FALSE
81	215422000	5027000	20855159928	레드캡투어	37	2009	1분기	FALSE	FALSE
83	503662000	30857000	20201843391	레드캡투어	39	2009	3분기	FALSE	FALSE
108	544481614	4611600	13037790654	모두투어	17	2009	1분기	FALSE	FALSE
110	586547017	6552350	17831640182	모두투어	19	2009	3분기	FALSE	FALSE
153	128959936	1252642	13597706178	세중	35	2009	1분기	FALSE	FALSE
155	56881950	6980850	16130093114	세중	37	2009	3분기	FALSE	FALSE
172	505167621	NA	11135171990	참좋은레져	9	2009	1분기	FALSE	FALSE
174	563375028	NA	14608039919	참좋은레져	11	2009	3분기	FALSE	FALSE
195	477624164	19607350	5582468824	툿데관광개발	13	2009	1분기	FALSE	FALSE
197	600097366	28783950	7726065331	툿데관광개발	15	2009	3분기	FALSE	FALSE
236	653303451	1066590	4823797414	자유투어	31	2009	1분기	FALSE	FALSE
238	650179780	-706010	7022767720	자유투어	33	2009	3분기	FALSE	FALSE
279	81561788	13427260	2542768350	비티앤아이	33	2009	1분기	FALSE	FALSE
281	120160963	720000	2677270912	비티앤아이	35	2009	3분기	FALSE	TRUE
38	1055742790	44554342	48480421492	하나투어	38	2010	1분기	TRUE	FALSE
40	1723098483	0	66245202818	하나투어	40	2010	3분기	FALSE	FALSE
85	285347000	3554000	26266456633	레드캡투어	41	2010	1분기	FALSE	FALSE
87	377363000	50701000	28589391364	레드캡투어	43	2010	3분기	FALSE	FALSE
112	627109367	11581130	25003802066	모두투어	21	2010	1분기	FALSE	FALSE
114	1265995826	20890280	36466418562	모두투어	23	2010	3분기	FALSE	FALSE
157	92244120	15798800	14808015873	세중	39	2010	1분기	FALSE	FALSE
159	48997402	-4759221	20282845926	세중	41	2010	3분기	FALSE	FALSE
176	663249619	NA	8464425371	참좋은레져	13	2010	1분기	FALSE	FALSE
178	734878924	NA	14717211161	참좋은레져	15	2010	3분기	FALSE	FALSE
199	660158348	28187000	7761111255	툿데관광개발	17	2010	1분기	FALSE	FALSE
201	609214711	30269450	12561340891	툿데관광개발	19	2010	3분기	FALSE	FALSE
240	817154204	516680	11317681906	자유투어	35	2010	1분기	FALSE	FALSE

```
242  932169708     852300 10455493093     자유투어   37 2010 3분기 FALSE FALSE
283  116821788    2240000  2270390985   비티앤아이   37 2010 1분기 FALSE FALSE
285  148188000    6260000  4986588998   비티앤아이   39 2010 3분기 FALSE  TRUE
>
```

데이터를 종횡과 횡형으로 변형하기 이렇게 처음과 마지막 레코드를 확인할 수 있는 지시자를 이용하여 어떤 분석자는 "다트8.1"과 같은 데이터가 주어졌을때, 각 여행사별로 2008년 1분기 매출액과 2010년 4분기의 매출액을 비교하여 그 차이를 알아내기 위해서 아래와 같은 데이터를 조작할 수 있습니다.

```
다트8.2 <- 다트8.1[c(FALSE, FALSE, TRUE, TRUE, FALSE, TRUE, FALSE, FALSE, FALSE)]
다트8.3 <- 다트8.2[order(다트8.2$여행사, 다트8.2$년도), ]
다트8.3$first <- !duplicated(다트8.3$여행사)
다트8.3$last <- !duplicated(다트8.3$여행사, fromLast=TRUE)
다트8.4 <- subset(다트8.3, subset=(first == TRUE | last == TRUE))
```

```
> 다트8.4
         매출액       여행사 년도 first  last
149 16463319852         세중 2008  TRUE FALSE
159 20282845926         세중 2010 FALSE  TRUE
30  57624282255     하나투어 2008  TRUE FALSE
40  66245202818     하나투어 2010 FALSE  TRUE
104 26530405258     모두투어 2008  TRUE FALSE
114 36466418562     모두투어 2010 FALSE  TRUE
232  4803096281     자유투어 2008  TRUE FALSE
242 10455493093     자유투어 2010 FALSE  TRUE
77  19926326125   레드캡투어 2008  TRUE FALSE
87  28589391364   레드캡투어 2010 FALSE  TRUE
168  5790815204   참좋은레저 2008  TRUE FALSE
178 14717211161   참좋은레저 2010 FALSE  TRUE
275  1012965573   비티앤아이 2008  TRUE FALSE
285  4986588998   비티앤아이 2010 FALSE  TRUE
191 12954724212 롯데관광개발 2008  TRUE FALSE
201 12561340891 롯데관광개발 2010 FALSE  TRUE
```

처음과 마지막 레코드를 명시하는 지시자는 불필요하므로 데이터로부터 제거합니다.

```
다트8.5 <- 다트8.4[,-c(4:5)]
```

```
> 다트8.5
         매출액       여행사 년도
149 16463319852         세중 2008
159 20282845926         세중 2010
```

```
30  57624282255      하나투어 2008
40  66245202818      하나투어 2010
104 26530405258      모두투어 2008
114 36466418562      모두투어 2010
232  4803096281      자유투어 2008
242 10455493093      자유투어 2010
77  19926326125   레드캡투어 2008
87  28589391364   레드캡투어 2010
168  5790815204   참좋은레저 2008
178 14717211161   참좋은레저 2010
275  1012965573   비티앤아이 2008
285  4986588998   비티앤아이 2010
191 12954724212 롯데관광개발 2008
201 12561340891 롯데관광개발 2010
>
```

그런데, 데이터가 종형으로 배열되어 있기 때문에 2008년과 2010년 매출액의 차이를 쉽게 구할 수 없습니다. 그래서, 아래와 같이 데이터를 횡형으로 재배열 해야 합니다.

```
> 다트8.6 <- reshape(다트8.5, timevar="년도", idvar="여행사", direction="wide")
          여행사 매출액.2008 매출액.2010
149         세중 16463319852 20282845926
30      하나투어 57624282255 66245202818
104     모두투어 26530405258 36466418562
232     자유투어  4803096281 10455493093
77   레드캡투어 19926326125 28589391364
168  참좋은레저  5790815204 14717211161
275  비티앤아이  1012965573  4986588998
191 롯데관광개발 12954724212 12561340891
>
```

이제서야 원하는 차이를 구할 수 있습니다.

```
> 다트8.6$차이 <- with(다트8.6, 매출액.2010 - 매출액.2008)
> 다트8.6
          여행사 매출액.2008 매출액.2010       차이
149         세중 16463319852 20282845926 3819526074
30      하나투어 57624282255 66245202818 8620920563
104     모두투어 26530405258 36466418562 9936013304
232     자유투어  4803096281 10455493093 5652396812
77   레드캡투어 19926326125 28589391364 8663065239
168  참좋은레저  5790815204 14717211161 8926395957
275  비티앤아이  1012965573  4986588998 3973623425
```

```
191 롯데관광개발 12954724212 12561340891 -393383321
>
```

위에서는 종형으로 이루어진 데이터를 횡형으로 변경하였으나, 우리는 이 횡형으로 된 데이터를 다시 종형으로도 되돌릴 수 있습니다. 이와 같이 하기 위해서는 아래와 같이 하면 됩니다.

```
> reshape(다트8.6,  v.names=c("매출액"), varying=c("매출액.2008", "매출액.2010"), direction="long", time
              여행사 년도       매출액  id
149.2008          세중 2008 16463319852 149
30.2008      하나투어 2008 57624282255  30
104.2008     모두투어 2008 26530405258 104
232.2008     자유투어 2008  4803096281 232
77.2008    레드캡투어 2008 19926326125  77
168.2008   참좋은여저 2008  5790815204 168
275.2008   비티앤아이 2008  1012965573 275
191.2008 롯데관광개발 2008 12954724212 191
149.2010          세중 2010 20282845926 149
30.2010      하나투어 2010 66245202818  30
104.2010     모두투어 2010 36466418562 104
232.2010     자유투어 2010 10455493093 232
77.2010    레드캡투어 2010 28589391364  77
168.2010   참좋은여저 2010 14717211161 168
275.2010   비티앤아이 2010  4986588998 275
191.2010 롯데관광개발 2010 12561340891 191
>
```

현재의 데이터셋을 가지고 보여줄 수 있는 추가적인 사항들 – (지금 이것들 전부다 문자열과 관계되는 부분임)

- 두 문자형 변수 결합하기
- 특정 문자열 뽑아내기
- 변수의 길이 파악하기

아래와 같은 내용을 보여주기 위해서는 다른 데이터셋이 필요함

- 주어진 데이터셋으로부터 랜덤샘플 추출하기
- 데이터셋 합치기와 머지하기
- 대소문자 전환
- 시간과 날짜 데이터 다루기

1.5 추가적인 유용한 조작팁들

결측치를 바로 윗값으로 채워넣기: 아래와 같이 주어진 데이터에 변수 ID는 결측값 없이 모든 값이 완전하게 잘 들어가 있는데, Week 변수에는 각 ID의 첫번째 레코드에만 해당하는 부분에 값이 들어가 있고 나머지부분에는 NA값이 들어가 있습니다.

```
mydata <- data.frame(ID=c(rep(1,4), rep(2,4), rep(3,2)), Week=c(15, NA, NA, NA, 18, NA, NA, NA, 20, NA))
```

```
> mydata
```

```
   ID Week
1   1   15
2   1   NA
3   1   NA
4   1   NA
5   2   18
6   2   NA
7   2   NA
8   2   NA
9   3   20
10  3   NA
```

이와 같은 데이터를 아래와 같이 자동으로 채워주려면 어떻게 해야 할까요?

```
   ID Week
1   1   15
2   1   15
3   1   15
4   1   15
5   2   18
6   2   18
7   2   18
8   2   18
9   3   20
10  3   20
```

이를 수행하는데에는 여러 가지 종류의 함수들이 다양한 패지키 안에 존재합니다. 그러나, 이를 수행하는 기본 알고리즘은 동일하며, R 기본시스템만으로 작성이 가능합니다. 아래의 함수를 복사하여 사용하시면 됩니다.

```
fill <- function(x, first, last){
        n <- last-first+1
        for(i in c(1:length(first))) x[first[i]:last[i]] <- rep(x[first[i]], n[i])
        return(x)
}
```

TODO:

- 그룹별 연산하는 방법에 대해서 설명을 해줘야 함 – aggregate(), tapply(), mapply(), sapply(), lapply(),
- 여기에서는 데이터 조작만으로 한정짓고, 통계량을 구하는 방법은 모두 통계 파트로 넘김 – 즉, apply()계열의 함수를 모두 통계파트로 넘김.
- expand.grid() 이건 수치해석 쪽으로 넘김.
- gl() 은 여기에서 다루어야 함.