Getting Data into R

```r
data("anscombe")
```

```r
ans <- anscombe
```

```r
str(ans)
```

```
'data.frame':    11 obs. of  8 variables:
 $ x1: num  10 8 13 9 11 14 6 4 12 7 ...
 $ x2: num  10 8 13 9 11 14 6 4 12 7 ...
 $ x3: num  10 8 13 9 11 14 6 4 12 7 ...
 $ x4: num  8 8 8 8 8 8 8 19 8 8 ...
 $ y1: num  8.04 6.95 7.58 8.81 8.33 ...
 $ y2: num  9.14 8.14 8.74 8.77 9.26 8.1 6.13 3.1 9.13 7.26 ...
 $ y3: num  7.46 6.77 12.74 7.11 7.81 ...
 $ y4: num  6.58 5.76 7.71 8.84 8.47 7.04 5.25 12.5 5.56 7.91 ...
```

```r
head(ans)
```

```
  x1 x2 x3 x4   y1   y2    y3   y4
1 10 10 10  8 8.04 9.14  7.46 6.58
2  8  8  8  8 6.95 8.14  6.77 5.76
3 13 13 13  8 7.58 8.74 12.74 7.71
4  9  9  9  8 8.81 8.77  7.11 8.84
5 11 11 11  8 8.33 9.26  7.81 8.47
6 14 14 14  8 9.96 8.10  8.84 7.04
```

```r
data("injury")
str(injury)
```

```
'data.frame':    7150 obs. of  30 variables:
 $ durat   : num  1 1 84 4 1 1 7 2 175 60 ...
 $ afchnge : int  1 1 1 1 1 1 1 1 1 1 ...
 $ highearn: int  1 1 1 1 1 1 1 1 1 1 ...
 $ male    : int  1 1 1 1 1 1 1 1 1 1 ...
 $ married : int  0 1 1 1 1 1 1 1 1 1 ...
 $ hosp    : int  1 0 1 1 0 0 0 1 1 1 ...
 $ indust  : int  3 3 3 3 3 3 3 3 3 3 ...
 $ injtype : int  1 1 1 1 1 1 1 1 1 1 ...
 $ age     : int  26 31 37 31 23 34 35 45 41 33 ...
 $ prewage : num  405 644 398 528 529 ...
 $ totmed  : num  1188 361 8964 1100 373 ...
 $ injdes  : int  1010 1404 1032 1940 1940 1425 1110 1207 1425 1010 ...
 $ benefit : num  247 247 247 247 212 ...
 $ ky      : int  1 1 1 1 1 1 1 1 1 1 ...
 $ mi      : int  0 0 0 0 0 0 0 0 0 0 ...
```

```
$ ldurat  : num   0 0 4.43 1.39 0 ...
$ afhigh  : int   1 1 1 1 1 1 1 1 1 1 ...
$ lprewage: num   6 6.47 5.99 6.27 6.27 ...
$ lage    : num   3.26 3.43 3.61 3.43 3.14 ...
$ ltotmed : num   7.08 5.89 9.1 7 5.92 ...
$ head    : int   1 1 1 1 1 1 1 1 1 1 ...
$ neck    : int   0 0 0 0 0 0 0 0 0 0 ...
$ upextr  : int   0 0 0 0 0 0 0 0 0 0 ...
$ trunk   : int   0 0 0 0 0 0 0 0 0 0 ...
$ lowback : int   0 0 0 0 0 0 0 0 0 0 ...
$ lowextr : int   0 0 0 0 0 0 0 0 0 0 ...
$ occdis  : int   0 0 0 0 0 0 0 0 0 0 ...
$ manuf   : int   0 0 0 0 0 0 0 0 0 0 ...
$ construc: int   0 0 0 0 0 0 0 0 0 0 ...
$ highlpre: num   6 6.47 5.99 6.27 6.27 ...
- attr(*, "time.stamp")= chr "25 Jun 2011 23:03"
```

```
names(injury)
```

```
 [1] "durat"    "afchnge"  "highearn" "male"     "married"  "hosp"
 [7] "indust"   "injtype"  "age"      "prewage"  "totmed"   "injdes"
[13] "benefit"  "ky"       "mi"       "ldurat"   "afhigh"   "lprewage"
[19] "lage"     "ltotmed"  "head"     "neck"     "upextr"   "trunk"
[25] "lowback"  "lowextr"  "occdis"   "manuf"    "construc" "highlpre"
```

CSV

```
ifri_car_liv <- get_csv("ifri_car_liv.csv")

ifri <- ifri_car_liv
str(ifri)
```

```
Classes 'data.table' and 'data.frame':  100 obs. of  10 variables:
 $ forest_id: int   217 325 88 174 240 287 324 321 216 82 ...
 $ cid      : chr   "NEP" "IND" "UGA" "NEP" ...
 $ zliv     : num   -0.614 -0.654 -0.338 -0.786 -0.45 ...
 $ zbio     : num   -0.451 -0.365 -0.97 -1.325 -1.049 ...
 $ livcar1  : int   3 3 3 3 3 3 3 3 3 3 ...
 $ ownstate : int   1 1 1 1 1 1 1 1 1 0 ...
 $ distance : int   2 1 1 2 2 1 2 2 2 1 ...
 $ sadmin   : int   0 1 3 26 3 40 8 0 0 0 ...
 $ rulematch: int   0 0 0 0 1 1 0 0 1 1 ...
 $ lnfsize  : num   4.43 8.2 4.94 5.29 4.34 ...
 - attr(*, ".internal.selfref")=<externalptr>
```

STRATA

```r
ifir <- get_strata('ifri_car_liv.dta')
str(ifir)
```

```
Classes 'tbl_df', 'tbl' and 'data.frame':   80 obs. of  10 variables:
 $ forest_id: num  217 325 88 174 240 287 324 321 216 82 ...
  ..- attr(*, "label")= chr "Forest_ID"
  ..- attr(*, "format.stata")= chr "%8.0g"
 $ cid      : chr  "NEP" "IND" "UGA" "NEP" ...
  ..- attr(*, "label")= chr "Country"
  ..- attr(*, "format.stata")= chr "%9s"
 $ zliv     : num  -0.614 -0.654 -0.338 -0.786 -0.45 ...
  ..- attr(*, "label")= chr "Livelihood Contributions Index (Standardized)"
  ..- attr(*, "format.stata")= chr "%9.0g"
 $ zbio     : num  -0.451 -0.365 -0.97 -1.325 -1.049 ...
  ..- attr(*, "label")= chr "Carbon Storage (Basal Area)"
  ..- attr(*, "format.stata")= chr "%9.0g"
 $ livcar1  : 'haven_labelled' num  3 3 3 3 3 3 3 3 3 3 ...
  ..- attr(*, "label")= chr "Forest Commons Outcomes"
  ..- attr(*, "format.stata")= chr "%13.0g"
  ..- attr(*, "labels")= Named num  1 2 3 4
  .. ..- attr(*, "names")= chr  "Deferred Use" "Sustainable" "Overused" "Unsustainable"
 $ ownstate : 'haven_labelled' num  1 1 1 1 1 1 1 1 1 0 ...
  ..- attr(*, "label")= chr "Forest Commons Ownership"
  ..- attr(*, "format.stata")= chr "%9.0g"
  ..- attr(*, "labels")= Named num  0 1
  .. ..- attr(*, "names")= chr  "Community" "Government"
 $ distance : 'haven_labelled' num  2 1 1 2 2 1 2 2 2 1 ...
  ..- attr(*, "label")= chr "Distance of Users to Forest Commons"
  ..- attr(*, "format.stata")= chr "%9.0g"
  ..- attr(*, "labels")= Named num  1 2 3
  .. ..- attr(*, "names")= chr  "< 5KM" "5 - 10KM" ">10KM"
 $ sadmin   : num  0 1 3 26 3 40 8 0 0 0 ...
  ..- attr(*, "label")= chr "Distance of Forest to Nearest Administrative Center"
  ..- attr(*, "format.stata")= chr "%9.0g"
 $ rulematch: 'haven_labelled' num  0 0 0 0 1 1 0 0 1 1 ...
  ..- attr(*, "label")= chr "Local Autonomy "
  ..- attr(*, "format.stata")= chr "%9.0g"
  ..- attr(*, "labels")= Named num  0 1
  .. ..- attr(*, "names")= chr  "Low Autonomy" "High Autonomy"
 $ lnfsize  : num  4.43 8.2 4.94 5.29 4.34 ...
  ..- attr(*, "label")= chr "Log of Forest Size"
  ..- attr(*, "format.stata")= chr "%9.0g"
```

Search World Development Indicators

```r
WDIsearch("gdp.*capita.*PPP")
```

```
     indicator
[1,] "6.0.GDPpc_constant"
[2,] "NY.GDP.PCAP.PP.KD.ZG"
[3,] "NY.GDP.PCAP.PP.KD.87"
[4,] "NY.GDP.PCAP.PP.KD"
[5,] "NY.GDP.PCAP.PP.CD"
     name
[1,] "GDP per capita, PPP (constant 2011 international $) "
[2,] "GDP per capita, PPP annual growth (%)"
[3,] "GDP per capita, PPP (constant 1987 international $)"
[4,] "GDP per capita, PPP (constant 2011 international $)"
[5,] "GDP per capita, PPP (current international $)"
```

```r
wdi_data <- WDI(indicator =
                c("NY.GDP.PCAP.PP.KD",
                  "EN.ATM.CO2E.PC"),
              start = 2010,
              end = 2010,
              extra = T)

names(wdi_data)
```

```
 [1] "iso2c"             "country"          "year"
 [4] "NY.GDP.PCAP.PP.KD" "EN.ATM.CO2E.PC"   "iso3c"
 [7] "region"            "capital"          "longitude"
[10] "latitude"          "income"           "lending"
```