Environments

20173204 MYUNG BIN KWAK

2020-05-15

Environments

```
load("./r_modifying_values.RData")
ls()
```

```
##  [1] "a"             "ace"           "ace3"          "b"
##  [5] "c"             "deal"          "deck"          "deck2"
##  [9] "deck3"         "deck4"         "deck5"         "facecard"
## [13] "h"             "l"             "N"             "q"
## [17] "queen_spades"  "queen0fSpades" "s"             "shuffle"
## [21] "vec"           "w"
```

```
head(deck)
```

```
##    face   suit value
## 1  king spades    13
## 2 queen spades    12
## 3  jack spades    11
## 4   ten spades    10
## 5  nine spades     9
## 6 eight spades     8
```

```
deal
```

```
## function (cards)
## {
##     cards[1, ]
## }
```

```
deal(deck)
```

```
##   face   suit value
## 1 king spades    13
```

```
deal(deck)
```

```
##   face   suit value
## 1 king spades    13
```

```
deal(deck)
```

```
##   face   suit value
## 1 king spades    13
```

```
head(shuffle(deck))
```

```
##     face     suit value
## 16  jack    clubs    11
## 49  four   hearts     4
## 23  four    clubs     4
## 45 eight   hearts     8
## 32 eight diamonds     8
## 44  nine   hearts     9
```

```
head(deck)
```

```
##    face   suit value
## 1  king spades    13
## 2 queen spades    12
## 3  jack spades    11
## 4   ten spades    10
## 5  nine spades     9
## 6 eight spades     8
```

Environments

```
# install.packages("pryr", repos = "https://cran.rstudio.com")
library(pryr)
```

```
## Registered S3 method overwritten by 'pryr':
##   method      from
##   print.bytes Rcpp
```

```
parenvs()
```

```
##   label                      name
## 1 <environment: R_GlobalEnv> ""
```

```
parenvs(all = TRUE)
```

```
##    label                              name
## 1  <environment: R_GlobalEnv>         ""
## 2  <environment: package:pryr>        "package:pryr"
## 3  <environment: package:stats>       "package:stats"
## 4  <environment: package:graphics>    "package:graphics"
## 5  <environment: package:grDevices>   "package:grDevices"
## 6  <environment: package:utils>       "package:utils"
## 7  <environment: package:datasets>    "package:datasets"
## 8  <environment: package:methods>     "package:methods"
## 9  <environment: 0x0000000012ed96f8> "Autoloads"
## 10 <environment: base>                ""
## 11 <environment: R_EmptyEnv>          ""
```

```
search()
```

```
##  [1] ".GlobalEnv"        "package:pryr"      "package:stats"
##  [4] "package:graphics"  "package:grDevices" "package:utils"
##  [7] "package:datasets"  "package:methods"   "Autoloads"
## [10] "package:base"
```

Working with Environments

```
as.environment("package:stats")
```

```
## <environment: package:stats>
## attr(,"name")
## [1] "package:stats"
## attr(,"path")
## [1] "C:/Program Files/R/R-3.6.3/library/stats"
```

```
globalenv()
```

```
## <environment: R_GlobalEnv>
```

```
baseenv()
```

```
## <environment: base>
```

```
emptyenv()
```

```
## <environment: R_EmptyEnv>
```

```
parent.env(globalenv())
```

```
## <environment: package:pryr>
## attr(,"name")
## [1] "package:pryr"
## attr(,"path")
## [1] "C:/Users/kki96/OneDrive/문서/R/win-library/3.6/pryr"
```

```
# parent.env(emptyenv())
ls(emptyenv())
```

```
## character(0)
```

```
ls(globalenv())
```

```
##  [1] "a"             "ace"           "ace3"          "b"
##  [5] "c"             "deal"          "deck"          "deck2"
##  [9] "deck3"         "deck4"         "deck5"         "facecard"
## [13] "h"             "l"             "N"             "q"
## [17] "queen_spades"  "queen0fSpades" "s"             "shuffle"
## [21] "vec"           "w"
```

```
head(globalenv()$deck, 3)
```

```
##    face   suit value
## 1  king spades    13
## 2 queen spades    12
## 3  jack spades    11
```

```
assign("new", "Hello Global", envir = globalenv())
ls()
```

```
##  [1] "a"             "ace"           "ace3"          "b"
##  [5] "c"             "deal"          "deck"          "deck2"
##  [9] "deck3"         "deck4"         "deck5"         "facecard"
## [13] "h"             "l"             "N"             "new"
## [17] "q"             "queen_spades"  "queen0fSpades" "s"
## [21] "shuffle"       "vec"           "w"
```

```
globalenv()$new
```

```
## [1] "Hello Global"
```

The Active Environment

Read pp. 97

```
environment()
```

```
## <environment: R_GlobalEnv>
```

Scoping Rules

Read pp.98

Assignment

```
new
```

```
## [1] "Hello Global"
```

```
new <- "Hello Active"
new
```

```
## [1] "Hello Active"
```

```
roll <- function(){
  die <- 1:6
  dice <- sample(die, size = 2, replace = TRUE)
  sum(dice)
}
```

Evaluation

```
show_env <- function(){
  list(ran.in = environment(),
       parent = parent.env(environment()),
       objects = ls.str(environment()))
}
show_env()
```

```
## $ran.in
## <environment: 0x0000000018627980>
##
## $parent
## <environment: R_GlobalEnv>
##
## $objects
```

```
show_env()
```

```
## $ran.in
## <environment: 0x00000000186a3418>
##
## $parent
## <environment: R_GlobalEnv>
##
## $objects
```

```
#> origin environment (pp. 101)
environment(show_env)
```

```
## <environment: R_GlobalEnv>
```

```
environment(parenvs)
```

```
## <environment: namespace:pryr>
```

```
show_env <- function(){
  a <- 1
  b <- 2
  c <- 3
  list(ran.in = environment(),
       parent = parent.env(environment()),
       objects = ls.str(environment()))
}
show_env()
```

```
## $ran.in
## <environment: 0x000000001896a238>
##
## $parent
## <environment: R_GlobalEnv>
##
## $objects
## a :  num 1
## b :  num 2
## c :  num 3
```

```
#> R will copy over each argument to the runtime environment
foo <- "take me to your runtime"
show_env <- function(x = foo) {
  list(ran.in = environment(),
       parent = parent.env(environment()),
       objects = ls.str(environment()))
}
show_env()
```

```
## $ran.in
## <environment: 0x0000000018bd6910>
##
## $parent
## <environment: R_GlobalEnv>
##
## $objects
## x :  chr "take me to your runtime"
```

```
#> Calling environment : R calls the function from
rm("foo")
# fix(show_env)
show_env <- function() {
  list(ran.in = environment(),
       parent = parent.env(environment()),
       objects = ls.str(environment()))
}
show_env()
```

```
## $ran.in
## <environment: 0x0000000018d20670>
##
## $parent
## <environment: R_GlobalEnv>
##
## $objects
```

Fix deal() and shuffle()

```
deal <- function(){
  deck[1, ]
}
deal()
```

```
##   face   suit value
## 1 king spades    13
```

```
environment(deal)
```

```
## <environment: R_GlobalEnv>
```

```
deal()
```

```
##   face   suit value
## 1 king spades    13
```

```
DECK <- deck
deck <- deck[-1, ]
nrow(deck)
```

```
## [1] 51
```

```
head(deck, n = 3)
```

```
##    face   suit value
## 2 queen spades    12
## 3  jack spades    11
## 4   ten spades    10
```

```
deck <- DECK
deal <- function(){
  card <- deck[1, ]
  deck <- deck[-1, ]
  card
}
deal()
```

```
##   face   suit value
## 1 king spades    13
```

```
#> `deal` will just create a slighthly altered copy of `deck` in the runtime environment
head(deck, n = 6)
```

```
##    face   suit value
## 1  king spades    13
## 2 queen spades    12
## 3  jack spades    11
## 4   ten spades    10
## 5  nine spades     9
## 6 eight spades     8
```

```
deal <- function(){
  card <- deck[1, ]
  assign("deck", deck[-1, ], envir = globalenv())
  card
}
deal()
```

```
##   face   suit value
## 1 king spades    13
```

```
head(deck, 6)
```

```
##    face   suit value
## 2 queen spades    12
## 3  jack spades    11
## 4   ten spades    10
## 5  nine spades     9
## 6 eight spades     8
## 7 seven spades     7
```

```
deal()
```

```
##    face   suit value
## 2 queen spades    12
```

```
head(deck, 6)
```

```
##    face   suit value
## 3  jack spades    11
## 4   ten spades    10
## 5  nine spades     9
## 6 eight spades     8
## 7 seven spades     7
## 8   six spades     6
```

```
#> Fix `shuffle()`
shuffle <- function(cards) {
  random <- sample(1:52)
  cards[random, ]
}
deck <- DECK
head(deck, 3)
```

```
##    face   suit value
## 1  king spades    13
## 2 queen spades    12
## 3  jack spades    11
```

```
a <- shuffle(deck)
head(deck, 3)
```

```
##    face   suit value
## 1  king spades    13
## 2 queen spades    12
## 3  jack spades    11
```

```
head(a, 3)
```

```
##     face   suit value
## 19 eight  clubs     8
## 10  four spades     4
## 52   ace hearts    14
```

```
shuffle <- function(){
  random <- sample(1:52)
  assign("deck", DECK[random, ], envir = globalenv())
}
shuffle()
head(deck, 6)
```

```
##     face     suit value
## 19 eight    clubs     8
## 34   six diamonds     6
## 28 queen diamonds    12
## 29  jack diamonds    11
## 40  king   hearts    13
## 30   ten diamonds    10
```

Closures

```
shuffle()
deal()
```

```
##     face   suit value
## 41 queen hearts    12
```

```
deal()
```

```
##    face   suit value
## 44 nine hearts     9
```

```
setup <- function(deck) {
  DECK <- deck

  DEAL <- function(){
    card <- deck[1, ]
    assign("deck", deck[-1, ], envir = globalenv())
    card
  }

  SHUFFLE <- function(){
    random <- sample(1:52)
    assign("deck", DECK[random, ], envir = globalenv())
  }
  list(deal = DEAL, shuffle = SHUFFLE)
}
cards <- setup(deck)
ls()
```

```
##  [1] "a"             "ace"           "ace3"          "b"
##  [5] "c"             "cards"         "deal"          "deck"
##  [9] "DECK"          "deck2"         "deck3"         "deck4"
## [13] "deck5"         "facecard"      "h"             "l"
## [17] "N"             "new"           "q"             "queen_spades"
## [21] "queen0fSpades" "roll"          "s"             "setup"
## [25] "show_env"      "shuffle"       "vec"           "w"
```

```
head(deck)
```

```
##    face   suit value
## 25  two  clubs     2
## 26  ace  clubs    14
## 23 four  clubs     4
## 9  five spades     5
## 21  six  clubs     6
## 52  ace hearts    14
```

```
str(cards)
```

```
## List of 2
##  $ deal   :function ()
##   ..- attr(*, "srcref")= 'srcref' int [1:8] 7 11 11 3 11 3 7 11
##   .. ..- attr(*, "srcfile")=Classes 'srcfilecopy', 'srcfile' <environment: 0x00000000124ccdf
0>
##  $ shuffle:function ()
##   ..- attr(*, "srcref")= 'srcref' int [1:8] 13 14 16 3 14 3 13 16
##   .. ..- attr(*, "srcfile")=Classes 'srcfilecopy', 'srcfile' <environment: 0x00000000124ccdf
0>
```

```
deal <- cards$deal
shuffle <- cards$shuffle
deal
```

```
## function(){
##     card <- deck[1, ]
##     assign("deck", deck[-1, ], envir = globalenv())
##     card
##   }
## <environment: 0x000000001838a6c0>
```

```
shuffle
```

```
## function(){
##     random <- sample(1:52)
##     assign("deck", DECK[random, ], envir = globalenv())
##   }
## <environment: 0x000000001838a6c0>
```

```
environment(deal)
```

```
## <environment: 0x000000001838a6c0>
```

```
environment(shuffle)
```

```
## <environment: 0x000000001838a6c0>
```

```
setup <- function(deck) {
  DECK <- deck

  DEAL <- function() {
    card <- deck[1, ]
    assign("deck", deck[-1, ], envir = parent.env(environment()))
    card
  }

  SHUFFLE <- function() {
    random <- sample(1:52, size = 52)
    assign("deck", DECK[random, ], envir = parent.env(environment()))
  }

  list(deal = DEAL, shuffle = SHUFFLE)
}
cards <- setup(deck)
deal <- cards$deal
shuffle <- cards$shuffle
deal()
```

```
##    face  suit value
## 25  two clubs     2
```

```
shuffle()
rm(deck)
shuffle()
deal()
```

```
##    face  suit value
## 21  six clubs     6
```

```
deal()
```

```
##    face   suit value
## 42 jack hearts    11
```

```
environment(deal)
```

```
## <environment: 0x00000000189054c8>
```

```
environment(shuffle)
```

```
## <environment: 0x00000000189054c8>
```

comments

이번시간에는 environments에 대해 배웠습니다. 딜 이라는 함수와 셔플 이라는 함수가 제대로 작동하지 않았을 때 어떻게 해야하는지에 대해 배울수있었습니다.함수를 구동시킬떄마다 새로 변화하는것을 알수있었습니다. 함수를 변화시키는 법과 서로 같게 만드는법을 알수있게 되었습니다. 덱을 없애더라도 자신이 가지고 있는것을 통해 값을 도출해낼수 있게 되었습니다. environment를통해 딜과 셔플을통해 카드게임을 만들수 있는 방법등을 배운것같습니다.