

기초 프로그래밍(R) 중간고사(샘플)

_____학과 학번_____이름_____

2019-04-09

I. Dice

die <- 1:6 를 실행하였다. 다음 물음에 답하십시오.

1. 다음 코드 중 에러 메시지가 나올 것으로 예상되는 것은?

가. die + 1

나. die + 1:2

다. die + 1:3

라. die + 1:4

2. 다음 중 그 결과가 다른 것을 하나 고르시오.

가. die * die

나. die ^ 2

다. die ** 2

라. die %**% die

3. 사칙 연산자 중 * 에 대한 도움말을 구하고자 할 때 옳지 않은 것은?

가. help(*)

나. help("*")

다. ?`*`

라. ?" "*"

4. 다음 중 나머지 셋과 다른 연산을 수행하고 있는 것은?

가. die %%% 2

나. "%%%"(die, 2)

다. `%%`(die, 2)

라. `%%`(2, die)

5. roll2 를 아래 글상자와 같이 정의한 함수라 할 때, 어떤 경우에 에러가 출력되는가?

가. roll2(bones = 1:4)

나. roll2(bones = 1:5)

다. roll2(1:20)

라. roll2()

```
roll2 <- function(bones) {  
  dice <- sample(bones, size = 2, replace = TRUE)  
  sum(dice)  
}
```

6. 위에서 정의한 `roll2` 를 나중에 다시 불러들여 사용하기 위하여 저장할 때 적절한 코드는?

- 가. `dump(list = "roll2", file = "./roll2.R")`
- 나. `save(list = "roll2", file = "./roll2.R")`
- 다. `save.image(file = "./roll2.R")`
- 라. `write(list = "roll2", file = "./roll2.R")`

II. Simulation

7. 공평한 주사위 두 개를 던져 합을 구하는 게임을 20번 반복하는 코드는 다음 중 어느 것인가? 위에서 정의한 `roll2` 를 이용하시오.

- 가. `replicate(20, roll2())`
- 나. `replicate(20, roll2(bones = 1:6))`
- 다. `replicate(roll2(), 20)`
- 라. `replicate(roll2(bones = 1:6), 20)`

8. `roll` 함수를 편집하여 확률 조건을 추가한 `roll.w` 를 만들려고 할 때 필요한 코드는? (단, 여기서 `;` 는 그 다음 코드를 이어서 수행하라는 것이다.)

- 가. `roll.w <- roll ; fix(roll.w)`
- 나. `roll.w <- roll ; vi(roll.w)`
- 다. `roll <- roll.w ; fix(roll.w)`
- 라. `roll <- roll.w ; vi(roll.w)`

III. R Objects

9. `is.vector(5)` 의 답은 무엇인가?

- 가. `TRUE`
- 나. `FALSE`
- 다. `Yes`
- 라. `No`

10. `attributes(die)` 를 수행했더니 `$names` 항목이 발견되었다. 이 속성을 없애고자 할 때 적당한 코드는?

- 가. `names(die) <- ""`
- 나. `names(die) <- NA`
- 다. `names(die) <- NULL`
- 라. `names(die) <- NaN`

11. `today <- Sys.Date()` ; `class(today)` 의 결과는?

- 가. `[1] "integer"`
- 나. `[1] "Date"`
- 다. `[1] "matrix"`
- 라. `[1] "array"`

IV. list and data frame

list2 <- list(number = 100:130, char = "R", logical = list(TRUE, FALSE)) 에서

12. list2[[2]] 의 결과는 무엇인가?

가.

```
$char  
[1] "R"
```

나.

```
[1] "R"
```

다.

```
$R  
[1] "char"
```

라.

```
[1] "char"
```

13. 다음 중 그 결과가 다른 것 하나를 고르시오.

가. list2\$logical

나. list2[[3]]

다. list2[3]

라. list2[3][[1]]

14. logical 중 첫번째 TRUE 를 불러내는 코드는 무엇인가? (즉, [1] TRUE 가 나오는 코드는?)

가. list2[3][[1]][[1]]

나. list2[3][[1]][1]

다. list2[3][1][1]

라. list2[3][1][[1]]

df 를 다음과 같이 정의할 때

```
df <- data.frame(face = c("ace", "two", "six"),  
                  suit = rep("clubs", 3),  
                  value = c(1, 2, 6))
```

15. str(df) 의 결과를 올바르게 나타낸 것은?

가.

```
'data.frame':   3 obs. of  3 variables:  
 $ face : chr   "ace" "two" "six"  
 $ suit : chr   "clubs" "clubs" "clubs"  
 $ value: num    1 2 6
```

나.

```
'data.frame': 3 obs. of 3 variables:
 $ face : Factor w/ 3 levels "ace","two","six": 1 2 3
 $ suit : Factor w/ 1 level "clubs": 1 1 1
 $ value: num 1 2 6
```

다.

```
'data.frame': 3 obs. of 3 variables:
 $ face : Factor w/ 3 levels "ace","six","two": 1 3 2
 $ suit : Factor w/ 1 level "clubs": 1 1 1
 $ value: num 1 2 6
```

라.

```
'data.frame': 3 obs. of 3 variables:
 $ face : chr "ace","two","six": 1 2 3
 $ suit : chr "clubs": 1 1 1
 $ value: num 1 2 6
```

다음과 같은 형식으로 되어 있는 cards.csv 의 URL이

https://raw.githubusercontent.com/coop711/r_programming/master/data/cards.csv 라고 하자.

```
1 "face","suit","value"
2 "king","spades",13
3 "queen","spades",12
4 "jack","spades",11
```

16. 이를 읽어서 deck 에 저장하는 코드로 적절한 것은? 단, string 은 character 로 읽어들이는다.

가.

```
url <- "https://raw.githubusercontent.com/coop711/r_programming/master/data/cards.csv"
deck <- read.csv(url)
```

나.

```
url <- https://raw.githubusercontent.com/coop711/r_programming/master/data/cards.csv
deck <- read.csv(url)
```

다.

```
url <- "https://raw.githubusercontent.com/coop711/r_programming/master/data/cards.csv"
deck <- read.csv(url, stringsAsFactors = FALSE)
```

라.

```
url <- https://raw.githubusercontent.com/coop711/r_programming/master/data/cards.csv
deck <- read.csv(url, stringsAsFactors = FALSE)
```

17. 위와 같이 읽어들이는 deck 을 작업 디렉토리나 같은 위치에 있는 data 폴더에 cards.csv 라는 이름으로 저장하고자 한다. 단, 첫 열에 나오는 일련번호를 없애고자 할 때 적절한 코드는?

가. `write.csv(deck, file = "../data/cards.csv")`

나. `write.csv(deck, file = "../data/cards.csv", row.names = FALSE)`

다. `save(deck, file = "../data/cards.csv")`

라. `save(deck, file = "../data/cards.csv", row.names = FALSE)`

V. Notation

18. 앞에서 정의한 `deck` 에서 맨 처음 카드를 뽑아 `face`, `suit`, `value` 를 다 보고자 한다. 이 때 필요한 코드는?

가. `deck[1,]`

나. `deck[, 1]`

다. `deck[1, 1]`

라. `deck[1, 1, drop = FALSE]`

VI. Modifying Values

19. `vec <- 1:6` 에 7번째 원소를 추가하려다가 실수로 `vec[8] <- 7` 이라고 입력하여 7번째 원소만 없애고자 할 때 적합한 코드는 ?

가. `vec[7] <- ""`

나. `vec[7] <- NULL`

다. `vec[7] <- NaN`

라. `vec <- vec[-7]`

20. 위의 `deck` 에서 매 13번째 카드의 `value` 를 찾아보는 코드로 적합치 않은 것은?

가. `deck2$value[seq(13, 52, by = 13),]`

나. `deck2[seq(13, 52, by = 13), "value"]`

다. `deck2[seq(13, 52, by = 13), 3]`

라. `deck2[seq(13, 52, by = 13), c(FALSE, FALSE, TRUE)]`