Lab 02: Introduction to the R language (1)
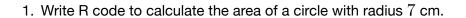
Seoncheol Park

1 Instructions

- Students should solve excercises provided in the in-class exercise session on every Friday.

- Students should write (a single) R code about answers of excercises, submit it on course website.

    - for 11628 (Fri 10:30~12:00)
    - for 13300 (Fri 09:00~10:30)

- Please **check** whether your answers are uploaded well.

- We will **not** give any scores for all **late submission**. Please keep the time.

- You may leave early after submitting your answers on the course website.

2 Exercise A: R as a calculator

1. Write R code to calculate the area of a circle with radius 7 cm.

2. Write R code to calculate the respective areas of the circles having radii $3, 4, \ldots, 100$.

3. Write a R function to calculate the area of a circle with radius r, where $r > 0$.

3 Exercise B: Vectors in R

1. Write a R code to calculate the sum $\sum_{j=0}^{n} r^j$, where r has been assigned the value $1.06, 1.08$, for $n = 10, 20, 30$ and 40.

2. Write a R code to calculate the sum $\sum_{j=0}^{n} r^j$, where r has been assigned the value 1.08, using the formula $(1 - r^{n+1})/(1 - r)$, for $n = 10, 20, 30$, and 40.

4 Exercise C: **rep( )** and **seq( )**

1. Using `rep( )` and `seq( )` as needed, create the vector

```
[1] 0 0 0 0 0 1 1 1 1 1 2 2 2 2 2 3 3 3 3 3 4 4 4 4 4
```

2. Using `rep( )` and `seq( )` as needed, create the vector

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

3. Using `rep( )`, `seq( )`, and + as needed, create the vector

```
[1] 1 2 3 4 5 2 3 4 5 6 3 4 5 6 7 4 5 6 7 8 5 6 7 8 9
```

5 Exercise D: Assigning variables

- First, note that names of variables in R may contain lowercase or capital letters, numbers, `.`, and `_`. The name must begin with a letter or `.`, and if it begins with `.`, the next character cannot be a number.

- You can use `;` to seperate multiple assignments on the same line. For example,

```
#addition and subtraction
2 + 2; 2 - 2
```

```
[1] 4
```

```
[1] 0
```

We want to make some names variables in R to covert between time units. For example, you can define `min.` using `sec.` variable.

```
sec. <- 1; min. <- 60*sec.
```

1. Write a R code to define `hr.`, `day.` `week.` using `min.`, `hr.` and `day`, respectively.

```
#define `hr.` using `min.`
hr. <-
#define `day.` using `hr.`
day. <-
#define `week.` using `day.`
week. <-
```

2. We can also define `yr.` and `century.` variable. What is the answer of `3*century./sec.` in R?

```
yr. = 365.25*day.; century. = 100*yr.
#what is the answer?
3*century./sec.
```