```
---
title: 'calss 6 : R Functions'
author: "May Wu PID:A59010588"
date: "10/15/2021"
output: html_document
---
```

````
```{r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
```
````

quick R Markdown intro
We can write text of course just like any file, we can **style the text to be bold** or *italic*

Do:
-this
-that
-and another thing

this is new text (two whitespaces to create a new line)
and this is a new line
---
we can include some code:

````
```{r}
# this is a comment
student1 <- c(100, 100, 100, 100, 100, 100, 100, 90)
student2 <- c(100, NA, 90, 90, 90, 90, 97, 80)
student3 <- c(90, NA, NA, NA, NA, NA, NA, NA)
```
````

short cut for adding code chunck
option + command + I (for R code)

Q1. Write a function grade() to determine an overall grade from a vector of student homework
assignment scores dropping the lowest single score. If a student misses a homework (i.e. has an
NA value) this can be used as a score to be potentially dropped. Your final function should be
adquately explained with code comments and be able to work on an example class gradebook
such as this one in CSV format: "https://tinyurl.com/gradeinput" [3pts]

First, find the lowest score using **min()**, and the **which.min()** function to find where it is (ie. the position in the vector)

````
```{r}
which.min(student1)
```
````

````
```{r}
student1[-which.min(student1)]
```
````

Now I can call the **mean()** function to get the average.

````
```{r}
mean(student1[-which.min(student1)])
```
````

does this work for student2?

````
```{r}
````

```r
mean(student1[-which.min(student1)])
```
No! why not?
```{r}
student2
which.min(student2)

```
```{r}
student3
```
one great idea is to replace the NA value with zeros
```{r}
which(is.na(student2))
is.na(student2)

```
```{r}
student2.prime <- student2
student2.prime [is.na(student2.prime)] = 0
student2.prime

mean(student2.prime[-which.min(student2.prime)])
```

for stident3
```{r}
student2.prime <- student3
student2.prime [is.na(student2.prime)] = 0
student2.prime
mean(student2.prime[-which.min(student2.prime)])
```

```{r}
x = 1:5
x
x[2] = 100
x
x[3] = 200
x
```

```{r}
x= student3
x[is.na(x)] = 0
mean(x[-which.min(x)])
```
```{r}
student4 <- c(100, NA, 90, "90", 90, 90, 97, 80)
f = as.numeric(student4)
f[is.na(f)] = 0
mean(f[-which.min(f)])

```
we finally can write our function:
all functions have at least 3 things:
- a name
- input args

– a body
```{r}
grade = function(x){
x= as.numeric(x)
x[is.na(x)] = 0
mean(x[-which.min(x)])
}
```

```{r}
grade(student1)
```

Q2. Using your grade() function and the supplied gradebook, Who is the top scoring student
overall in the gradebook? [3pts]

```{r}
gradebook = "https://tinyurl.com/gradeinput"
scores = read.csv(gradebook, row.names = 1)
hi = apply(scores, 1, grade)
hi[which.max(hi)]
cat("the student with highest grade is", names(hi[which.max(hi)]))
```

Q3. From your analysis of the gradebook, which homework was toughest on students (i.e. obtained
the lowest scores overall? [2pts]
```{r}
score_without_na = scores
score_without_na[is.na(score_without_na)] = 0
hard_hw = apply(score_without_na, 2, mean)
hard_hw[which.min(hard_hw)]
cat("the hardest homework is", names(hard_hw[which.min(hard_hw)]))
```

replace or mask NA values to zero

```{r}
```

Q4. Optional Extension: From your analysis of the gradebook, which homework was most
predictive of overall score (i.e. highest correlation with average grade score)? [1pt]
here we use the **cor()** function
```{r}
ans = apply(scores, 1, grade)
score_without_na = scores
score_without_na[is.na(score_without_na)] = 0
cor(score_without_na$hw5, ans)
apply(score_without_na,2, cor, ans)
```
```{r}
#make a box plot
boxplot(scores)
```