

# Class 6: R Functions

Garrett Cole

## Function basics

- A **name** (we can pick this but it must start with a character)
- Input **arguments** (there can be multiple comma separated inputs)
- The **body** (where work actually happens)

```
# Example input vectors to start with
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)
```

## Q1 Create Function grades()

Removing the minimum value and using mean function

```
mean(student1[-which.min(student1)])
```

```
[1] 100
```

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

```
[1] NA
```

```
mean(student3[-which.min(student3)])
```

```
[1] NA
```

Converting NA to 0

```
student1[is.na(student1)] <- 0  
student1
```

```
[1] 100 100 100 100 100 100 100 90
```

```
student2[is.na(student2)] <- 0  
student2
```

```
[1] 100 0 90 90 90 90 97 80
```

```
student3[is.na(student3)] <- 0  
student3
```

```
[1] 90 0 0 0 0 0 0 0
```

Find Grade By Combine NA <- 0 & Remove Min

```
student1[is.na(student1)] <- 0  
mean(student1[-which.min(student1)])
```

```
[1] 100
```

```
student2[is.na(student2)] <- 0  
mean(student2[-which.min(student2)])
```

```
[1] 91
```

```
student3[is.na(student3)] <- 0  
mean(student3[-which.min(student3)])
```

```
[1] 12.85714
```

I now have my working snippet of code that I have simplified to work with any student 'x'

```

grade <- function(x) {
  #Convert all NA to 0
  x[is.na(x)] <- 0
  #Calculate average with lowest score dropped
  mean(x[-which.min(x)])
}

```

```
grade(student1)
```

```
[1] 100
```

```
grade(student2)
```

```
[1] 91
```

```
grade(student3)
```

```
[1] 12.85714
```

## Q2

Using your grade() function and the supplied gradebook, Who is the top scoring student overall in the gradebook?

```

url <- "https://tinyurl.com/gradeinput"
gradebook <- read.csv(url, row.names = 1)

```

Grade all the students

```

avgStudent <- apply(gradebook, 1, grade)
avgStudent

```

student-1	student-2	student-3	student-4	student-5	student-6	student-7
91.75	82.50	84.25	84.25	88.25	89.00	94.00
student-8	student-9	student-10	student-11	student-12	student-13	student-14
93.75	87.75	79.00	86.00	91.75	92.25	87.75
student-15	student-16	student-17	student-18	student-19	student-20	
78.75	89.50	88.00	94.50	82.75	82.75	

Which student has the highest grade?

```
which.max(avgStudent)
```

```
student-18  
18
```

Student 18 has the highest overall grade at 94.50

### Q3

From your analysis of the gradebook, which homework was toughest on students (i.e. obtained the lowest scores overall)?

Find the average of each hw (each column)

```
avgHW <- apply(gradebook, 2, grade)  
which.min(avgHW)
```

```
hw2  
2
```

HW 2 was the toughest on students

### Q4

From your analysis of the gradebook, which homework was most predictive of overall score(i.e. highest correlation with average grade score)?

```
mask <- gradebook  
mask[ is.na(mask) ] <- 0  
  
cor(mask$hw1 , avgStudent)
```

```
[1] 0.4250204
```

```
cor(mask$hw2 , avgStudent)
```

```
[1] 0.176778
```

```
cor(mask$hw3 , avgStudent)
```

```
[1] 0.3042561
```

```
cor(mask$hw4 , avgStudent)
```

```
[1] 0.3810884
```

```
cor(mask$hw5 , avgStudent)
```

```
[1] 0.6325982
```

```
apply(mask, 2, cor, y = avgStudent)
```

	hw1	hw2	hw3	hw4	hw5
	0.4250204	0.1767780	0.3042561	0.3810884	0.6325982

Homework 5 had the highest correlation with the average grade score making it the most predictive of overall score