IntroToR

Vectors

```
# shortcut: option + - gives <-
x <- 1:5
y <- x + 100
z <- x + c(100,1)
```

```
## Warning in x + c(100, 1): longer object length is not a multiple of shorter
## object length
```

```
print(z)
```

```
## [1] 101   3 103   5 105
```

```
y <- c("a","b","c","d")
paste(y,"lo'o'p") #adds things to each of the elements in the vector
```

```
## [1] "a lo'o'p" "b lo'o'p" "c lo'o'p" "d lo'o'p"
```

```
grades <- c(a=1,b=2,c=4,d=0) #names associated with vectors
print(grades[2])
```

```
## b
## 2
```

```
print(grades["c"])
```

```
## c
## 4
```

```
sort(grades)
```

```
## d a b c
## 0 1 2 4
```

```
grades+c(scale=100) #does not change the names, does not add scale to the existing names, only add on t
```

```
##   a   b   c   d
## 101 102 104 100
```

Dataframes

```r
df <- data.frame(nums=1:5,chars=letters[1:5], logical=c(T,T,F,T,F))
df$nums # returns a vector
```

rows first cols second e.g., df[rows,cols]

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

```r
df["nums"] # does the same job
```

```
##   nums
## 1    1
## 2    2
## 3    3
## 4    4
## 5    5
```

```r
df[,1] # return the first col as a vector
```

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

```r
df[1] # return the dataframe (with the header), b/c most of the time observations are in rows and varia
```

```
##   nums
## 1    1
## 2    2
## 3    3
## 4    4
## 5    5
```

```r
df$chars[df$nums>3]
```

```
## [1] "d" "e"
```

```r
df[df$nums>3,]
```

```
##   nums chars logical
## 4    4     d    TRUE
## 5    5     e   FALSE
```