R_Loop

```
c1 <- c(3,5,2,7,8,3)
c2 <- c(4,5,9,1,5,7)
c3 <- c(1,6,3,2,4,8)
c4 <- c(4,1,8,6,5,2)
# Create a dataframe
df <- data.frame(c1, c2, c3, c4)
# name the rows
row.names(df) <- paste("r", 1:6, sep = '')
df
```

```
##    c1 c2 c3 c4
## r1  3  4  1  4
## r2  5  5  6  1
## r3  2  9  3  8
## r4  7  1  2  6
## r5  8  5  4  5
## r6  3  7  8  2
```

```
# Suppose we want to calculate the mean of col1 and col3 of the dataframe
# for row2, 3, and 6
# e.g., (5+6)/2 =5.5,(2+3)/2=2.5,(3+8)/2=5.5
# then store them in a vector, we want to output a vector and a data frame
id <- c(2, 3, 6)
means<-c(5.5,2.5,5.5)
output.df=data.frame(id,means)
output.df
```

```
##   id means
## 1  2   5.5
## 2  3   2.5
## 3  6   5.5
```

```
meansOfSomeRows <- function(input.df, id, cols.for.mean){
  # an initial vector v contains 3 entries, meaning
   v <- numeric(length(id))
   k <- 1  #  initialize the index for placing entries into v
   for (j in id) {
      row.of.df <- input.df[j, cols.for.mean]  # row j, with requested columns
# row.of.df is a 1 row data frame, not a vector
# the mean function expects a vector as input
# a data frame is a special type of list, so here
# we can use unlist to get a vector from row.of.df:
      row.vec <- unlist(row.of.df)
      row.mean <- mean(row.vec, na.rm = TRUE) # calculate the mean, discarding NA's
```

```
      v[k] <- row.mean
      k <- k + 1  #  increase the running index to the next position in v
   } # end of the for loop
# use id and v to create a 2 column data frame of the results
   result <- data.frame(id, means = v)
# for this function, return the data frame of the results
   return(result)
} # end of the function

id <- c(2, 3, 6)
cols.for.mean <- c(1, 3)
output.df <- meansOfSomeRows(input.df = df, id, cols.for.mean)
output.df
```

```
##   id means
## 1  2   5.5
## 2  3   2.5
## 3  6   5.5
```

```
meansOfSomeRows <- function(input.df, id, cols.for.mean){
   v <- numeric(length(id))
   for (j in 1:length(id)) {#have the loop index run from 1 to length of id
      row.index <- id[j]  #  get the desired row of the data frame from id
      row.of.df <- input.df[row.index, cols.for.mean]  # row, with requested columns
# row.of.df is a 1 row data frame, not a vector
# the mean function expects a vector as input
# a data frame is a special type of list, so here we
# we can use unlist to get a vector from row.of.df:
      row.vec <- unlist(row.of.df)
      row.mean <- mean(row.vec, na.rm = TRUE) # calculate the mean, discarding NA's
      v[j] <- row.mean # now j is the correct index to use in v
   } # end of the for loop
# use id and v to create a 2 column data frame of the results
   result <- data.frame(id, means = v)
# for this function, return the data frame of the results
   return(result)
} # end of the function
meansOfSomeRows(input.df = df, id, cols.for.mean)
```

```
##   id means
## 1  2   5.5
## 2  3   2.5
## 3  6   5.5
```

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

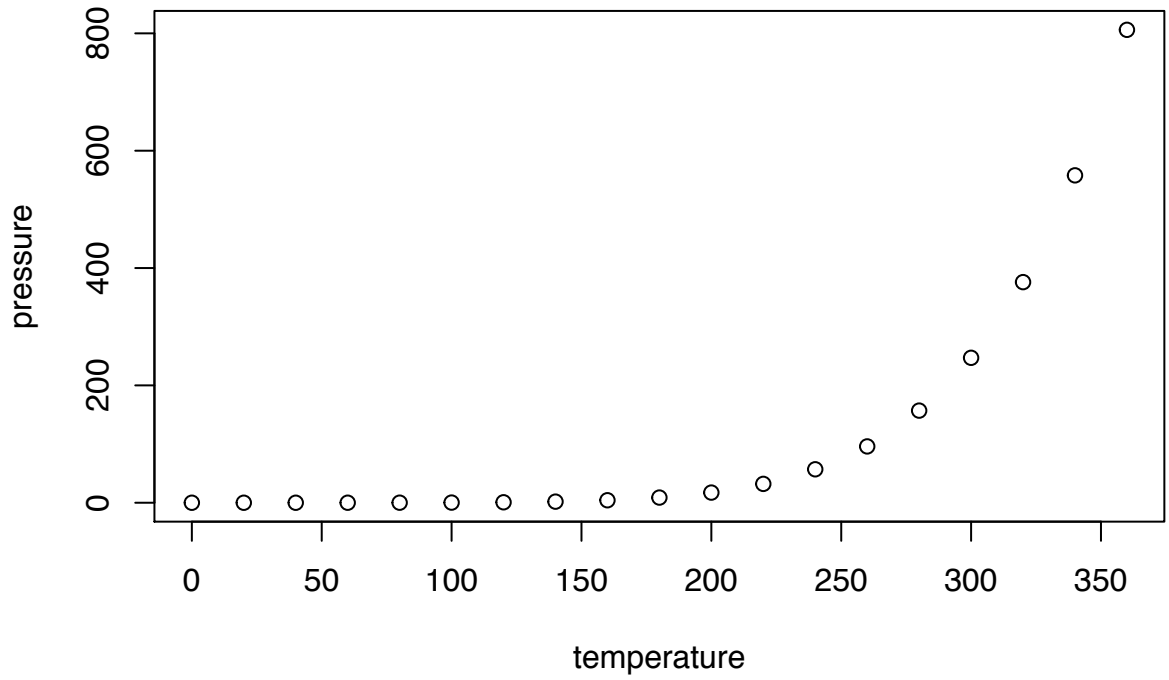
When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```r
summary(cars)
```

```
##      speed           dist       
##  Min.   : 4.0   Min.   :  2.00  
##  1st Qu.:12.0   1st Qu.: 26.00  
##  Median :15.0   Median : 36.00  
##  Mean   :15.4   Mean   : 42.98  
##  3rd Qu.:19.0   3rd Qu.: 56.00  
##  Max.   :25.0   Max.   :120.00
```

Including Plots

You can also embed plots, for example:



Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.