| | |
|---|---|
| install.packages(“ape”) | Install a library |
| library(ape) | Load a library |
| ?sum | Find help on functions by typing ? in front |
| setwd("~/myFolder/") | Set the working directory |

| | |
|---|---|
| c(“one”, “two”, “three”) | Concatenate things into a vector |
| myVector[1] | Get the 1st item of a vector |
| myVector[2:5] | Get items 2-5 |
| myDataFrame[1:5, ] | Get rows 1 to 5 and all columns of a data frame |
| myDataFrame[9, 2:3] | Get row 9 and columns 2:3 of a data frame |
| myItems %in% anotherVector | Test if items are in another vector |

**Logical Operators**

| | | | |
|---|---|---|---|
| > | greater than | < | less than |
| >= | greater or equal to | <= | less or equal to |
| == | is equal to (note two equals signs) | != | is not equal to |

**Data Frames**

| | |
|---|---|
| names(dataFrame) | Get the item names (or column names) |
| rownames(dataFrame) | get row names |
| colnames(dataFrame) | get column names |
| str(myDataFrame) | Print a summary of the structure of a variable |
| head(myDataFrame) | See just the first bit of a variable |
| tail(myDataFrame) | See just the last bit of a variable |

| | |
|---|---|
| table(x,y) | Make a contingency table of counts |

| | |
|---|---|
| length(x) | Get the length of a variable |
| dim(x) | Get the dimensions of a data frame (returns number of rows, columns) |

**Summary statistics**

| | |
|---|---|
| sum(x) | Sum of all items in x |
| sum(x, is.na=TRUE) | Sum of all items in x, ignoring NA values |
| mean(x) | Mean of all items in x |
| max(x) | maximum of all items in x |
| min(x) | minimum of all items in x |
| sd(x) | standard deviation of all items in x |
| nchar(x) | For each item, return the number of characters in the string |

| | |
|---|---|
| is.na(x) | Is each member of X an NA value? |
| is.inf() | Is each member of X an infinite value? |

**Apply functions to groups**

tapply(dataPoints,myGroups, myFunction)

*Split the items in dataPoints into groups according to myGroups (same length as dataPoints) and apply myFunction to each group.*

lapply(myList, myFunction) *Apply myFunction to each sub-list of myList*

apply(myDataFrameOrMatrix, 1, myFunction)

*Apply myFunction to each row of myDataFrame (or each column = 2)*

match(x, table) Get the positions of first matches for the elements of x in myTable

merge(dataFrame1, dataFrame2, by.x=”glotto.code”, by.y = “GLOTTO”)

Merge two data frames, using by.x and by.x as the keys to match rows

plot(xvalues, yvalues) Plot data (makes different plots depending on the data)

**Custom functions**

```
myFunction <- function(argument1, argument2){
        # put code here
}
```

**if statements**

```
if(myVariable > 5){
        # code here if true
} else{
        # code here if false
}
```

**For Loops**

```
for(i in myVectorOfNumbers){
        # do something with each element
}
```

**savePDFs**

```
pdf(file="myPdfFile.pdf")
        plot(x,y) # make plot
dev.off()
```

**Reading and writing csv data**

read.csv("myFile.csv", stringsAsFactors=FALSE)
write.csv(myDataFrame, file="myFile.csv")

**Reading and writing R objects**

save(myRObject, file="myFile.Rdat")
load("myFile.Rdat")

**Misc**

| | |
|---|---|
| map() | Draw a map of the world |
| points() | Add points to a plot |
| abline() | Draw a straight line in a plot |
| set.seed(323) | Set the random seed to a particular value |