```
#Solutions - functions

#A.
eggs<-c(2,3,4,0,0,5,3,4,1,2,4,3,2,0,2,3,4,3,1,3,2)
where<-c("BigForest","SmallForest","Lake","Meadow","Park","Grassland","Pasture","Roof",
        "BigForest2","SmallForest2","Lake2","Meadow2","Park2","Grassland2","Pasture2","Roof2"
        "BigForest3","SmallForest3","Lake3","Meadow3","Park3")

names(eggs)<-where

length(eggs)
sum(eggs)
min(eggs)
max(eggs)
mean(eggs)
median(eggs)
sum(eggs>3)
sum(eggs==4)
which(eggs==0)
sum(eggs[eggs>=3])
names(eggs)[eggs==0]
eggs[c("Park","Park2","Park3")]
mean(eggs[eggs>0])
eggs<-eggs[eggs>0]
eggs<-eggs[-which(names(eggs)=="Park2")]
eggs[eggs==1]<-6

B.
 size<-rnorm(50,2.2,0.7)
 size
mean(size)
sd(size)

max(size)
min(size)
range(size)
which.min(size)
summary(size)
sum(size)/length(size)


C.
rep(1:3,3)
rep(1:3,each=3)
rep(c("one","two","three"),c(2,1,2))


seq(3,1,-0.2)
seq(from=1,to=3,by=0.2)
 seq(1,3,length.out=20)

D.
ls()
save(size,eggs,where,.....,file="/Users/Ania/herefiles/file.Rdata")
rm(list=ls())
ls()
```

```
load("/Users/Ania/herefiles/file.Rdata")
ls()
```

Solutions - Full

A.

```
> length(eggs)
[1] 21
> sum(eggs)
[1] 51
> min(eggs)
[1] 0
> max(eggs)
[1] 5
> mean(eggs)
[1] 2.428571
> median(eggs)
[1] 3
> sum(eggs>3)
[1] 5
> sum(eggs==4)
[1] 4
> which(eggs==0)
   Meadow      Park Grassland
        4         5        14
> sum(eggs[eggs>=3])
[1] 39
> names(eggs)[eggs==0]
[1] "Meadow"    "Park"      "Grassland"
> eggs[c("Park","Park2","Park3")]
 Park Park2 Park3
    0     2     2
> mean(eggs[eggs>0])
[1] 2.833333
```

B.

```
size<-rnorm(50,2.2,0.7)
 size #this will be of course different
 [1] 1.2450944 3.1908517 2.1563320 3.0638650 2.4062644 2.2016408 3.0092131 2.6427507 2.409286
[13] 2.1802409 2.7080660 2.7076829 1.9349750 1.9802998 2.7580155 2.4072841 2.9223034 2.690642
[25] 2.3158771 2.0046932 2.5750031 1.6893695 3.0953761 2.7605742 2.0254984 2.1579782 1.396794
[37] 2.8152374 0.7800512 3.0016819 2.4028338 2.0573340 2.8072855 3.9578604 2.1710515 3.935481
[49] 2.3952727 2.0196676
> mean(size)
[1] 2.34455
> sd(size)
[1] 0.6653532
> range(size)
[1] 0.7800512 3.9578604
> max(size)
[1] 3.95786
> which.max(size)
[1] 43
> sum(size)/length(size)
[1] 2.34455
```