Assignment0

ToDo

```
 ((2018-2014)/(2014-1994))*100
```

```
## [1] 20
```

Todo

```
m <-(2018 - 2014)
x <- (2014 - 1994)
a <- ((m/x)*100)
a
```

```
## [1] 20
```

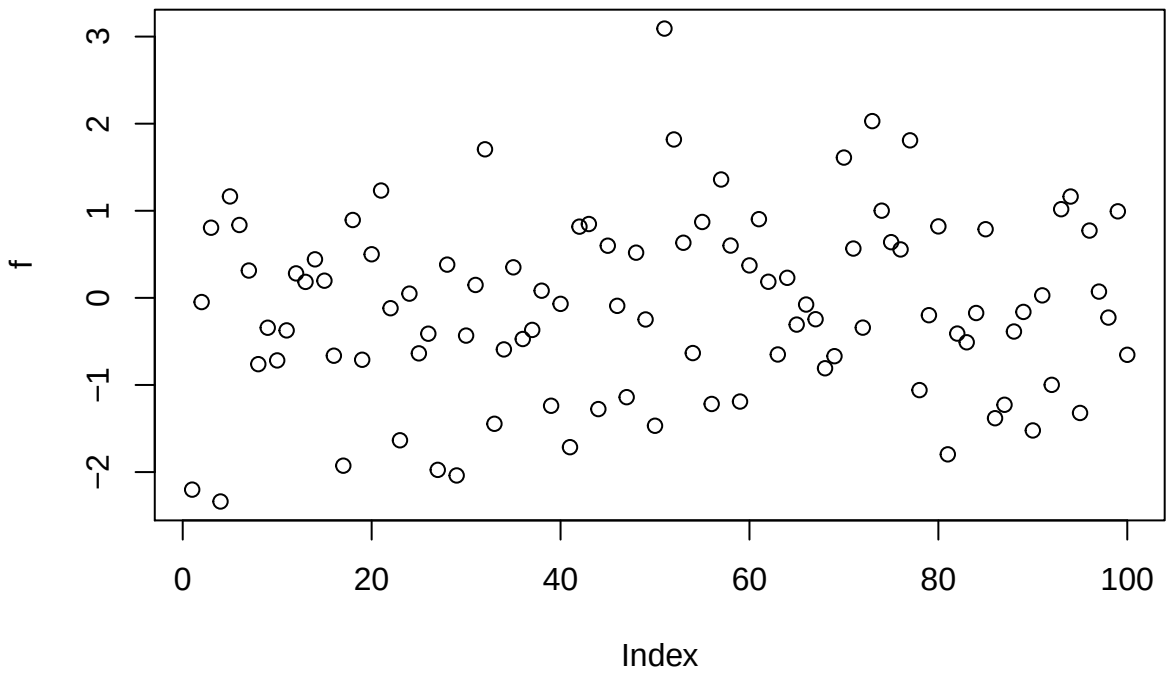
Todo

```
b= c(4,5,8,11)
sum(b)
```

```
## [1] 28
```

Todo

```
f = rnorm(100)
plot(f)
```



R Markdown

```
help (sqrt)
```

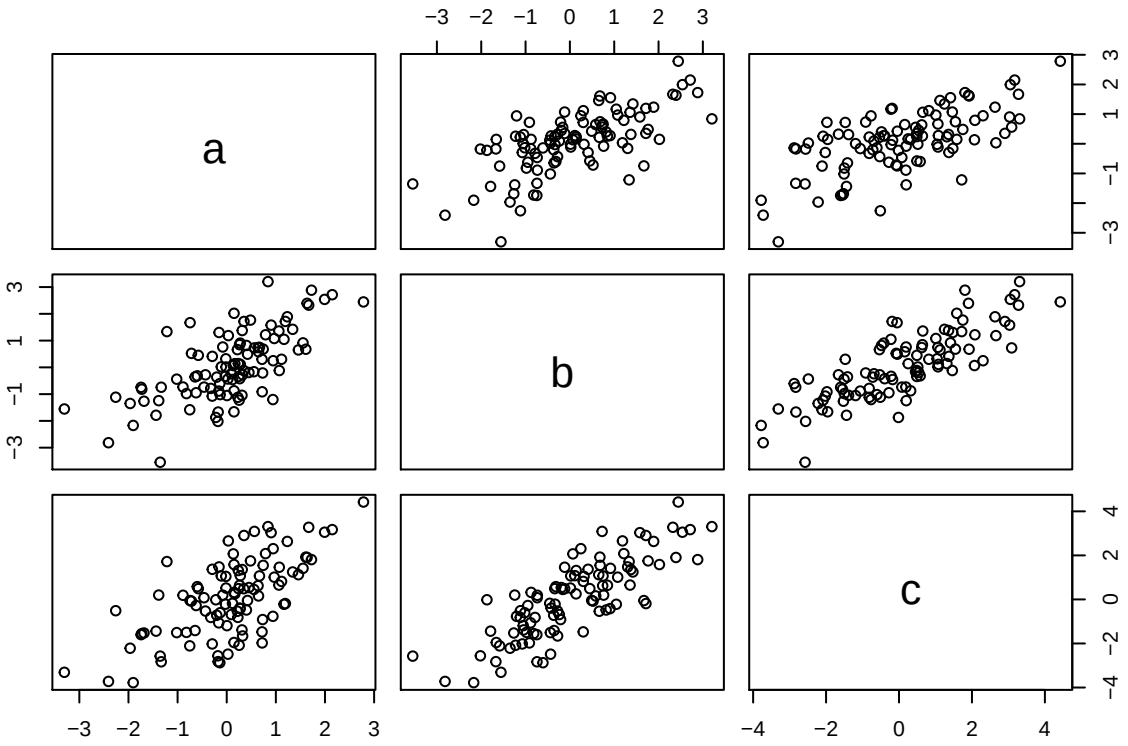
Todo

```
P= seq (from= 31, to= 60)
Q= (matrix(data= P, ncol=5, nrow=6))
Q
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]   31   37   43   49   55
## [2,]   32   38   44   50   56
## [3,]   33   39   45   51   57
## [4,]   34   40   46   52   58
## [5,]   35   41   47   53   59
## [6,]   36   42   48   54   60
```

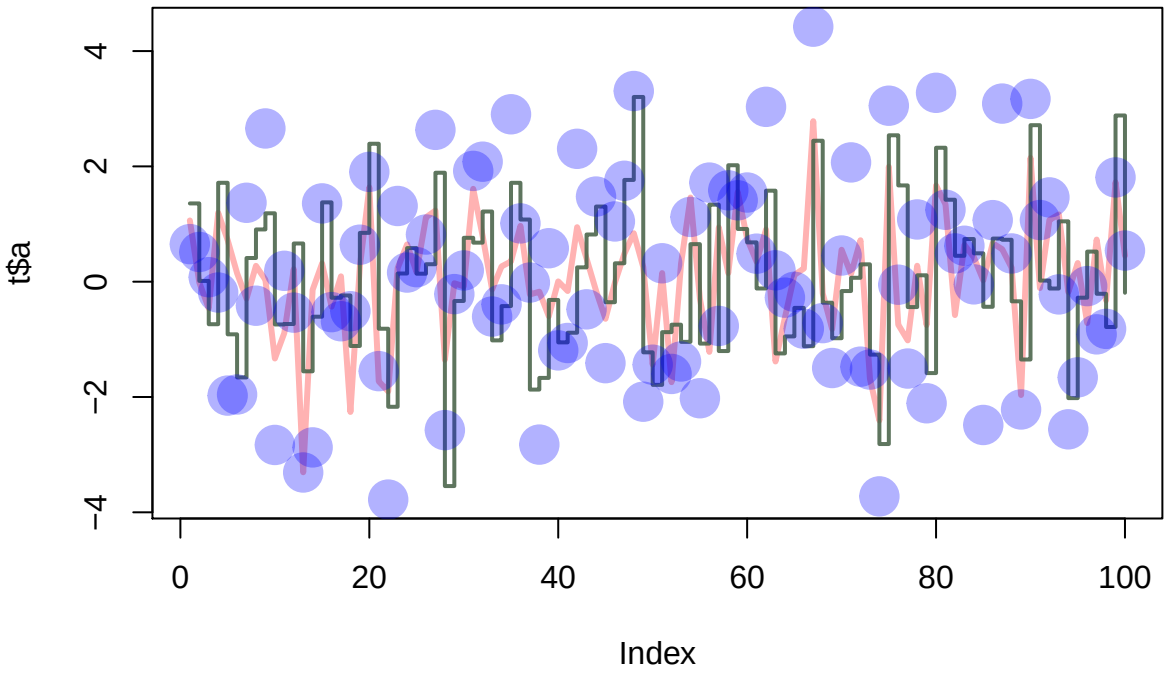
Todo

```
x1= rnorm(100)
x2= rnorm(100)
x3= rnorm(100)
t= data.frame(a = x1, b=(x1+x2), c= (x1+x2+x3))
plot(t)
```



Todo

```
plot(t$a, type="l", ylim=range(t),
     lwd=3, col=rgb(1,0,0,0.3))
lines(t$b, type="s", lwd=2,
      col=rgb(0.3,0.4,0.3,0.9))
points(t$c, pch=20, cex=4, col=rgb(0,0,1,0.3))
```



rgb = red, green and blue. This is used for colours in the group

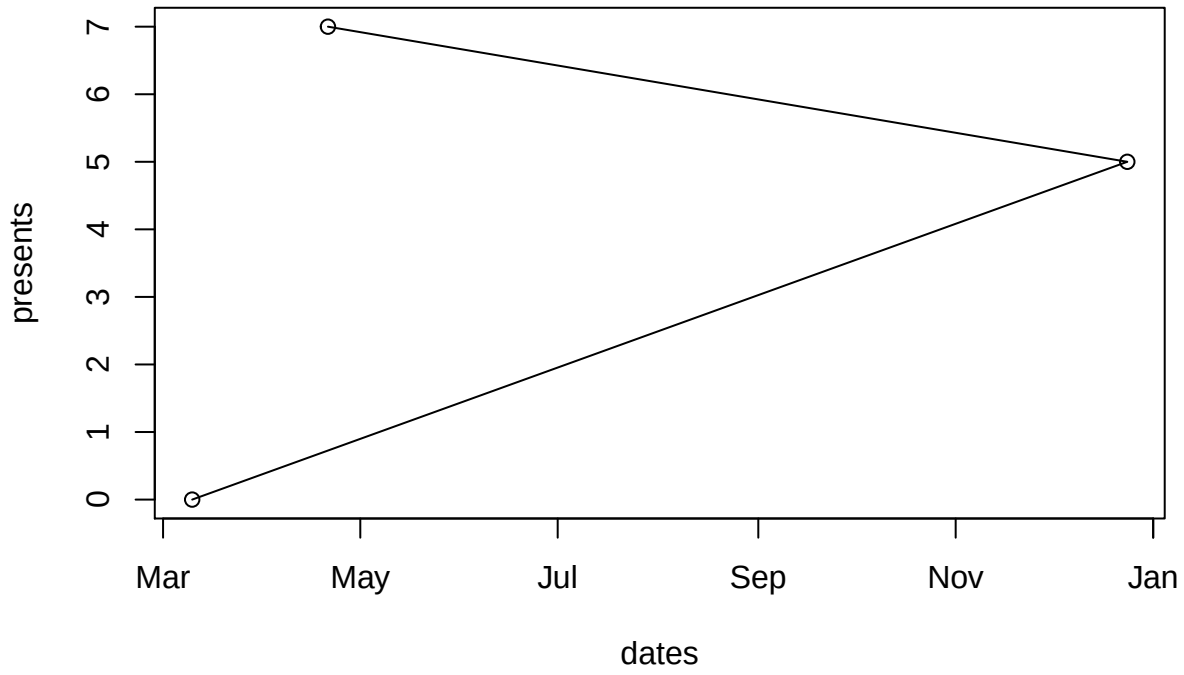
```
sqrt(rnorm(100))
```

```
## Warning in sqrt(rnorm(100)): NaNs produced
```

```
##   [1] 0.1383351       NaN       NaN 0.2672224 0.9075255 0.6317006       NaN
##   [8]       NaN       NaN       NaN 1.0302651       NaN 0.8153013 0.6505423
##  [15] 1.1282256       NaN 0.6186333 0.7963753       NaN 0.1599493 0.7928158
##  [22] 0.8369952 0.9216448       NaN 1.0611453       NaN       NaN 0.6662124
##  [29] 0.5778156       NaN       NaN       NaN       NaN 0.7838322       NaN
##  [36] 0.9184948       NaN 1.7003903       NaN 1.1469011 0.5833183 0.9681401
##  [43] 0.9707189 0.6768148       NaN 0.7280166 0.3880657       NaN 1.1071354
##  [50]       NaN       NaN       NaN 0.8491840       NaN       NaN 1.0709909
##  [57] 0.6621762 0.7897945       NaN 1.0423179       NaN       NaN       NaN
##  [64]       NaN       NaN 0.5071902 1.2097537       NaN       NaN 0.6102791
##  [71] 0.9655182 0.8035064       NaN 1.1590839 1.2474405 1.6158314       NaN
##  [78] 0.6526536 1.1688430       NaN       NaN       NaN 0.3904829       NaN
##  [85] 0.6999051       NaN 1.2618431       NaN 0.6409134       NaN 1.1996168
##  [92] 0.1817956 0.6218692       NaN 0.6739715 1.4068278 1.2047266       NaN
##  [99]       NaN       NaN
```

You get NaNs, not a number.

```
dates= strptime(c("20190310", "20191224","20190421"),format="%Y%m%d")
presents = c(0,5,7)
plot(dates,presents)
lines(dates,presents)
```



Todo

```
myfile<- read.table(file="tst1.txt" , header =TRUE)
myvar <- myfile$g * 5
write.table (myvar, file="tst2.txt")
```

Todo

```
myvec = seq(from=1, to=100)
s = c()

for(i in 1:length(myvec))
{
  if(myvec[i] < 5) {
    s[i] <- (myvec[i] *10)
  }else if (myvec[i] > 90){
    s[i] <- (myvec[i] *10)
  }else{
    s[i] = (myvec[i]*0.1)
  }
}
s
```

```
##   [1]   10.0   20.0   30.0   40.0    0.5    0.6    0.7    0.8    0.9    1.0
##  [11]    1.1    1.2    1.3    1.4    1.5    1.6    1.7    1.8    1.9    2.0
##  [21]    2.1    2.2    2.3    2.4    2.5    2.6    2.7    2.8    2.9    3.0
##  [31]    3.1    3.2    3.3    3.4    3.5    3.6    3.7    3.8    3.9    4.0
##  [41]    4.1    4.2    4.3    4.4    4.5    4.6    4.7    4.8    4.9    5.0
##  [51]    5.1    5.2    5.3    5.4    5.5    5.6    5.7    5.8    5.9    6.0
##  [61]    6.1    6.2    6.3    6.4    6.5    6.6    6.7    6.8    6.9    7.0
##  [71]    7.1    7.2    7.3    7.4    7.5    7.6    7.7    7.8    7.9    8.0
##  [81]    8.1    8.2    8.3    8.4    8.5    8.6    8.7    8.8    8.9    9.0
##  [91]  910.0  920.0  930.0  940.0  950.0  960.0  970.0  980.0  990.0 1000.0
```

Todo

```r
myfunc = function(argv1)
{
  S=c()
  for(i in 1:length(argv1))
  {
    if(argv1[i] < 5) {
      S[i] <- (argv1[i] *10)
    }else if (argv1[i] > 90){
      S[i] <- (argv1[i] *10)
    }else{
      S[i] = (argv1[i]*0.1)
    }
  }
  S
}

myfunc(argv1 = c(2:10))
```

```
## [1] 20.0 30.0 40.0  0.5  0.6  0.7  0.8  0.9  1.0
```