```
#Solutions to Worksheet 3.
> #Q1a
> normals <- rnorm(1000,3,4)
> #Q1b
> hist(normals)
> #Q1c
> min(normals)
[1] -7.672162
> max(normals)
[1] 14.06067
> #Q1d
> hist(normals, breaks=seq(floor(min(normals)),floor(max(normals)+1),1))
> hist(normals, breaks=seq(floor(min(normals)),floor(max(normals)+1),0.5))
> hist(normals, breaks=seq(floor(min(normals)),floor(max(normals)+1),0.25))
> hist(normals, breaks=seq(floor(min(normals)),floor(max(normals)+1),0.1))
>
> #hist(normals, breaks=seq(floor(min(normals)),floor(max(normals)+1),1.5))
> #This may not work, as the breaks may no longer span the range of values in
> #normals. In that case, expand the base of the histogram slightly:
> hist(normals, breaks=seq(floor(min(normals)-2),floor(max(normals)+2),1.5))
> hist(normals, breaks=seq(floor(min(normals)-4),floor(max(normals)+4),2))
>
> #Q1e
> grid <- seq(floor(min(normals)),floor(max(normals)+1),1)
> really.normal <- dnorm(grid,3,4)
> hist(normals, breaks=seq(floor(min(normals)),floor(max(normals)+1),1))
> lines(grid,really.normal)
```

```
> lines(grid,1000*1*really.normal)
> #Now you can try this out yourself on the other histograms.
> #Q1f
> qqnorm(normals)
> #Looks pretty linear!
> #Q1g
> expo <- rexp(15,1)
> mean(expo)
[1] 0.9065427
> var(expo)
[1] 0.2215546
> qqnorm(expo)
> #The solution to question 2 was given at the end of the worksheet.
> #Excludes Point 1
> betachng <- function(resp,pred,excl){
+   exc <- unique(excl)
+   if(min(excl)<1) {
+   print("Invalid Point to be Excluded - Index too small") }
+   else if(max(excl)>length(pred)) {
+   print("Invalid Point to be Excluded - Index too large") }
+   else {
+   beta <- lsfit(pred,resp)$coef
+   beta.red <-lsfit(pred[-exc],resp[-exc])$coef
+   beta - beta.red }
+   }
>
> betachng(Height,Weight,1)
```

```
  Intercept           X
 0.387648288 -0.007282526
> #Excludes Points 1,2 and 5
> betachng(Height,Weight,c(1,2,5))
  Intercept          X
-2.87756476  0.04095335
>
```