#1. Start RStudio and in the upper left pane type the code below and run it. 
What gets plotted? (You may investigate the `curve()`, `dunif()`, and `punif()` functions by, for 
example, typing `?curve()` in the R pane and reading documentation in the lower right pane.)

```r
curve(dunif(x, min=0, max=1), from=-1, to=2)
```

![Graph showing the dunif function from -1 to 2](image)

```r
curve(punif(x, min=0, max=1), from=-1, to=2)
```
punif(0.3, min=0, max=1)

## [1] 0.3

#2. In the upper left pane type the code below and run it.
curve(dnorm(x), from=-3, to=3)
\texttt{curve(dnorm(x, mean=2, sd=.5), from=.5, to=3.5)}

\texttt{curve(pnorm(x), from=-3, to=3)}
# On the basis of this kind of plotting, what can you say about the "shapes" of normal
# distributions with different means and standard deviations?  
# With less sd the "Bell-shaped" plot is more tighter than that of standard on e.

# 3. Plot the exponential pdfs for means alpha=1 and then alpha=.5 . (In the R parameterization these have "rates" respectively 1 and 2 .) Make these plots for arguments x = 1 to x=10 . Then plot the corresponding cdfs.

```r
curve(dexp(x, rate=1), from=-1, to=10)
curve(dexp(x, rate=2), from=-1, to=10)
```
\texttt{curve(pexp(x, rate=1), from=-1, to=10)}

\texttt{curve(pexp(x, rate=2), from=-1, to=10)}
#4. In the upper left pane type the code below and run it one line at a time and note which curve #is which. (Mark them on your lab before turning it in.) In the R parameterization "shape" is the #text's beta and "scale" is the text's alpha. Locate these curves in Figure 5.16 of the text.

```r
curve(dweibull(x, shape=4, scale=1), from=-.5, to=5)
curve(dweibull(x, shape=1, scale=1), from=-.5, to=5, add=TRUE)
curve(dweibull(x, shape=4, scale=2), from=-.5, to=5, add=TRUE)
```
#5. Type the code below in the upper left pane and run it. It should produce the median (the value $x$ at which $F(x) = P(X \leq x) = .5$) for the Weibull distribution with $\alpha=2$ and $\beta=4$. Use formula 5.30 in

```r
qweibull(.5, shape=4, scale=2)
```

## [1] 1.824889

#6. In the upper left pane type the code below and run it. It will fill a 100 000*5 matrix $M$ with (independent) realizations of $U(0,1)$ random variables and make histograms of the 5 sets of values in the columns of $M$. Are these exactly the same? Roughly, what is their shape?

```r
M<-matrix(runif(50000, min=0, max=1), nrow=10000, byrow=T)
hist(M[,1])
```
Histogram of $M[, 1]$

Histogram of $M[, 2]$

Histogram of $M[, 3]$

\texttt{hist}($M[, 2]$)

\texttt{hist}($M[, 3]$)
\texttt{hist(M[,3])}

\textbf{Histogram of M[, 3]}

\texttt{hist(M[,4])}

\textbf{Histogram of M[, 4]}

\texttt{hist(M[,5])}
Now type the code below and run it. What gets plotted? How does it compare to the second part in question 1 above?

```r
plot(sort(M[,1]),(1:10000)/10000,type="s",ylim=c(0,1),xlim=c(-1,2))
```
#7. Now type the code below into the upper left pane and run it. What do the plots it produces show about the distribution of averages of 5 independent $U(0,1)$ random variables?

```r
av<-1:10000
for (i in 1:10000)
    av[i]<-mean(M[i,])
hist(av,freq=FALSE)
curve(dnorm(x,mean=.5,sd=.1291),add=TRUE)
```

![Histogram of av](attachment:image.png)

```r
plot(sort(av),(1:10000)/10000,type="s",ylim=c(0,1),xlim=c(-1,2))
curve(pnorm(x,mean=.5,sd=.1291),add=TRUE)
```
#8. Redo parts 6 and 7 using exponential random variables with mean 1. (When comparing to normal distributions you will want mean=1, sd=.4472.)

```r
M <- matrix(rexp(50000, rate=1), nrow=10000, byrow=T)
M[342,4]
```

```r
## [1] 0.9457802
```

```r
hist(M[,1])
```
hist(M[,1])

Histogram of M[, 1]

hist(M[,2])

Histogram of M[, 2]

hist(M[,3])
Hist(M[3])

hist(M[, 4])

Hist(M[, 4])

Hist(M[, 5])

hist(M[, 5])
av<-1:10000
for (i in 1:10000){
  av[i]<-mean(M[i,])
}
hist(av,freq=FALSE)
curve(dnorm(x,mean=1,sd=.4472),add=TRUE)
#9. Redo part 8 based on not 5 but rather 10 exponential random variables with mean 1. (Now when comparing to normal distributions you will want mean=1, sd=.3162.) How do your plots in 8 and this part compare?

```r
M <- matrix(rexp(100000, rate=1), nrow=10000, byrow=T)
M[342,4]
## [1] 0.03689345
hist(M[,1])
```
```
hist(M[,1])

hist(M[,2])

hist(M[,3])
```
hist(M[,3])

Histogram of M[, 3]

hist(M[,4])

Histogram of M[, 4]

hist(M[,5])
av <- 1:10000
for (i in 1:10000){
  av[i] <- mean(M[i,])
}

hist(av, freq=FALSE)
curve(dnorm(x, mean=1, sd=.3162), add=TRUE)
#It will be more like a normal.