1) Generate random samples
Most of the distributions were discussed in Lab 3 or are similar to ones that are included in Lab 3. Please use online help for details. To generate a die roll, a discrete uniform distribution can be simulated from continuous uniform distribution with the following procedure:

The following code should replace the data.vec statement in the code below.

dice <- sample(1:6,n*average,replace=T) #simulates the die role
data.vec <- rep(0,n*average) #initializes the vector to be 0
#a success only occurs when the roll is a 2
for (i in 1:n*average) {
  if (dice[i]==2) data.vec[i] <- 1
}

2) Generate sampling distributions
The usefulness of the Central Limit Theorem (CLT) is through the averaging of random samples to simulate what is happening when many samples are taken from the same distribution.

Even though you will just be running the provided code (with minor modifications), I will outline what is happening below:

1) Generate the appropriate number of SRSs of each of the distributions.

2) Average (by row) the appropriate number of SRSs. For example, let’s assume that we are using 5 SRS of size 6.

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-0.30</td>
<td>-1.28</td>
<td>0.24</td>
<td>1.28</td>
<td>1.20</td>
<td>1.73</td>
</tr>
<tr>
<td>2</td>
<td>-2.18</td>
<td>-0.23</td>
<td>1.10</td>
<td>-1.09</td>
<td>-0.69</td>
<td>-1.69</td>
</tr>
<tr>
<td>3</td>
<td>-1.85</td>
<td>-0.98</td>
<td>-0.77</td>
<td>-2.12</td>
<td>-0.57</td>
<td>-0.40</td>
</tr>
<tr>
<td>4</td>
<td>0.13</td>
<td>-0.37</td>
<td>-0.33</td>
<td>-0.37</td>
<td>1.34</td>
<td>-0.09</td>
</tr>
<tr>
<td>5</td>
<td>-0.19</td>
<td>-0.51</td>
<td>1.97</td>
<td>0.87</td>
<td>2.38</td>
<td>-0.65</td>
</tr>
</tbody>
</table>

Each of the SRS are in the columns (letters). Our sample size (what we are averaging) are the rows (numbers) and the result is placed in column G.
3) From column G, generate histogram, QQ plot, mean and standard deviation.

code:

```r
n <- 1000 # the number of repeats (not to be changed)

# average: the number of columns that are being averaged over
average <- 1
# I strongly suggest that you change any titles to display the number
# of columns that you use are averaging and the type of distribution.

# calculates the average data
data.vec <- rnorm(n*average, mean=0, sd=1) # creates the random data
data.mat <- matrix(data.vec, ncol = average) # separates the data into columns
# apply(matrix, c(1, 2) == c("row", "column"), function)
avg <- apply(data.mat, 1, mean) # performs the averaging

. . .
```

The data that you want to analyze is in the variable ‘avg’. You will need to include the rest of
the code as appropriate to produce what is required in the assignment.