Discussion 3 - Ishita Dutta

Problem 1

I. Explore the sample function by typing ?sample. *sample takes a sample of the specified size from the elements of x using either with or without replacement.*

```
sample(x, size, replace = FALSE, prob = NULL)

#x:either a vector of one or more elements from which to choose, or a positive integer. See 'Details.'

#n: a positive number, the number of items to choose from. See 'Details.'

#size: a non-negative integer giving the number of items to choose.

#replace:   should sampling be with replacement?

#prob:  a vector of probability weights for obtaining the elements of the vector being sampled.

sample.int(n, size = n, replace = FALSE, prob = NULL,
           useHash = (!replace && is.null(prob) && size <= n/2 && n > 1e7))
```

(a) Create and object named die that contains the numbers 1 through 6. Provide the results.

```
die = 1:6
die
```

```
## [1] 1 2 3 4 5 6
```

(b) Generate one roll of a die of size 2 with R'ssamplefunction. Call this objectdice.Display the results.

```
dice = sample(die, 2)
dice
```

```
## [1] 4 6
```

(c) Repeat part (b) by sampling with replacement. Display the results.

```
dice = sample(die, 20, replace = TRUE)
dice
```

```
##  [1] 3 6 4 4 2 3 5 5 3 4 2 5 4 4 2 6 4 1 6 5
```

(d) Find the sum, mean, and standard deviation for part(c).

```r
df = data.frame(
  sum = sum(dice),
  mean = mean(dice),
  sd = sd(dice)
)
knitr::kable(df, caption = "Summary of Part C")
```

Table 1: Summary of Part C

sum	mean	sd
78	3.9	1.447321

#Problem 2

Every function in R has 3 basic parts: A name, body of code, and set of arguments.Create a function that provides the sum when you generate 10 outcomes of one roll.Call the functionrolland have it run the body of code on the argument die.

Why we use Functions:

-reduces code length -reduces debugging efforts

(a) Display your function.

```r
roll = function(die = 1:6, size = 10, replacement  = TRUE){
  #input -->  die: vector to sample
  #           size: integer, sample size
  #
  #output--> integer, sum of the roll with length size
  #
  #Description: provides the sum when you generate 10 outcomes of one roll
  outcome = sample(die, size, replace = replacement)
  output = sum(outcome)
  return(output)
}
```

(b) Run your function for a 4 sided die and provide the result.

```r
value = roll(die = 1:4,size = 10)
value
```

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

(c) Run your function for an 8 sided die and provide the result.

```r
value = roll(die = 1:8,size = 10)
value
```

```
## [1] 46
```

#Problem 3 Use the lynx dataset to do the following:

(a) Find the range of the lynx data. What did it return?

```
range(lynx)
```

```
## [1]   39 6991
```

(b) Create a function calledmy.rangethat will return the maximum - the minimum. Call the argument used in the functionMyData. Display the function and the result.

```
my.range = function(x){
  min = min(x)
  max = max(x)
  return(max - min)
}

my.range(lynx)
```

```
## [1] 6952
```