####VMM 04/09/2015

###Data Incubator Question 1

#### NUMBER OR M

num <- 10000

#### NUMBER OF SIMULATIONS

sim <- 1000

#### Starting data frame values are meaningless testnum=1:sim value=1:sim testnum <- 1:sim value <- 1:sim

test=data.frame(testnum=testnum,value=value)

### FUNCTION TO SUM DICE UNTIL

### SUM IS GREATER THAN OR EQUAL TO "M"

### SAMPLE FOR A NUMBER OF TIMES UP TO "iter"

### PASS BACK THROUGH A DATAFRAME

DiceSum = function(M, iter, dataframe)

{

i <- 1;

testnum <- numeric();

value <- numeric();

while(i < iter+1)

{

sum <- 0;

rolls <- 0;

while(M > sum)

{

sum = sum + sample(1:6,1,rep=T);

rolls = rolls + 1;

}

testnum[i] <- rolls;

value[i] <- sum;

i = i + 1;

}

dataframe$testnum <- testnum;

dataframe$value <- value;

dataframe

}

test <- DiceSum(num, sim, test)

###CHECK TO ENSURE VALUES MAKE SENSE

#test$testnum

#test$value

### MEAN AND STANDARD DEVIATION OF ROLLS MINUS M mean(test$value - num) sd(test$value - num) ### MEAN AND STANDARD DEVIATION OF ROLLS TO GET TO M

mean(test$testnum)

sd(test$testnum)

# M = 20

#> mean(test$value - num)

#[1] 1.66174

#> sd(test$value - num)

#[1] 1.489987

#> mean(test$testnum)

#[1] 6.19175

#> sd(test$testnum)

#[1] 1.217866

#M = 10000

#> mean(test$value - num)

#[1] 1.643

#> sd(test$value - num)

#[1] 1.435832

#> ### MEAN AND STANDARD DEVIATION OF ROLLS TO GET TO M

# > mean(test$testnum)

#[1] 2857.128

#> sd(test$testnum)

#[1] 25.94295