This assignment will be reviewed by peers based upon a given rubric. Make sure to keep your answers clear and concise while demonstrating an understanding of the material. Be sure to give all requested information in markdown cells. It is recommended to utilize Latex.

Problem 1

The Birthday Problem: This is a classic problem that has a nonintuitive answer. Suppose there are N students in a room.

Part a)

What is the probability that at least two of them have the same birthday (month and day)? (Assume that each day is equally likely to be a student's birthday, that there are no sets of twins, and that there are 365 days in the year. Do not include leap years).

Note: Jupyter has two types of cells: Programming and Markdown. Programming is where you will create and run R code. The Markdown cells are where you will type out expalantions and mathematical expressions. [Here (https://www.markdownguide.org/basic-syntax/)](https://www.markdownguide.org/basic-syntax/) is a document on Markdown some basic markdwon syntax. Also feel free to look at the underlying markdown of any of the provided cells to see how we use markdown.

$$\begin{aligned} P(\text{At least two have same birthday}) &= 1 - P(\text{All } N \text{ persons have different birthdays}) \\ &= 1 - \frac{365}{365}\frac{364}{365}\cdots\frac{365-N+1}{365} \\ &= 1 - \frac{365!}{(365-N)!365^N} \end{aligned}$$

Here the probability that the i^{th} person has different brithday from the first $i-1$ persons = $\frac{365-i+1}{365}$. Hence the proability that All N persons have different brthdays = $\prod_{i=1}^{N}\frac{365-i+1}{365}$.

Part b)

How large must N be so that the probability that at least two of them have the same birthday is at least 1/2?

Required $1 - \frac{365!}{(365-N)!365^N} \geq \frac{1}{2}$. Using the following code snippet, In [20]:

```
N = 100
P = c(1)
for (i in 2:N) {
    P = c(P, P[i-1]*(365-i+1)/365)
}
Prob = 1 - P
which(Prob >= 1/2)[1]
```

23

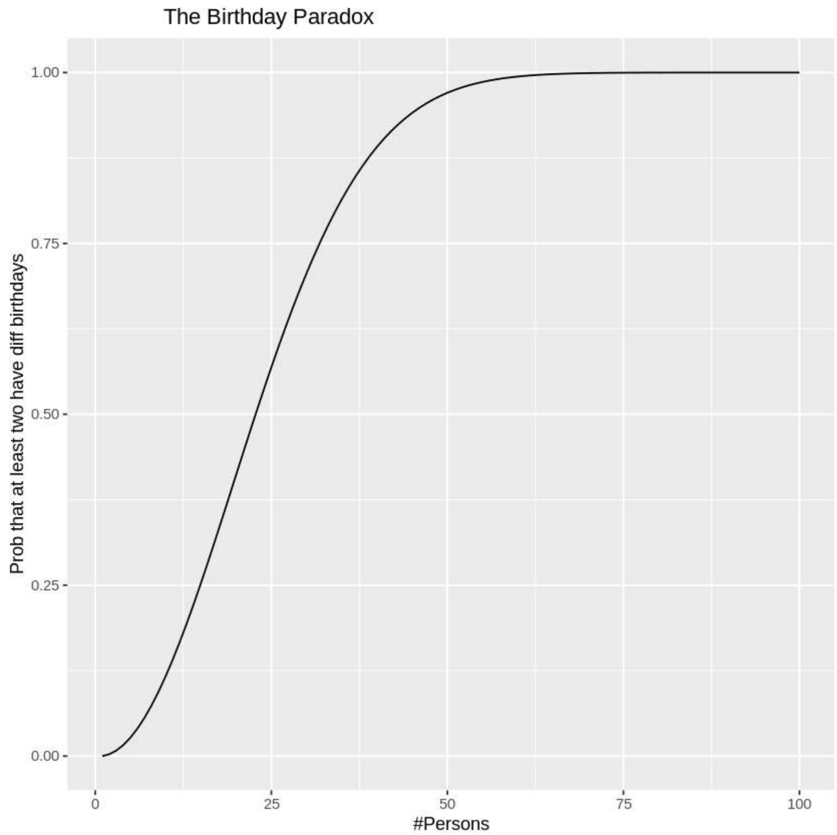
As can be seen from above, N must be at least 23.

Part c)

Plot the number of students on the x-axis versus the probability that at least two of them have the same birthday on the y-axis.

In [10]:

```
library(ggplot2)
ggplot() + geom_line(aes(1:N, Prob)) + xlab('#Persons') + ylab('Prob that at least two
 have diff birthdays') +
                                       ggtitle('\t\tThe Birthday Paradox')
```



Thought Question (Ungraded)

Thought question (Ungraded): Would you be surprised if there were 100 students in the room and no two of them had the same birthday? What would that tell you about that set of students?

YOUR ANSWER HERE

Problem 2

One of the most beneficial aspects of R, when it comes to probability, is that it allows us to simulate data and random events. In the following problem, you are going to become familiar with these simulation functions and techniques.

Part a)

Let X be a random variable for the number rolled on a fair, six-sided die. How would we go about simulating X?

Start by creating a list of numbers [1, 6]. Then use the `sample()` function with our list of numbers to simulate **a single** roll of the die, as in simulate X. We would recommend looking at the documentation for `sample()`, found [here (https://www.rdocumentation.org/packages/base/versions/3.6.2/topics/sample)](https://www.rdocumentation.org/packages/base/versions/3.6.2/topics/sample), or by executing `?sample` in a Jupyter cell.

In [1]:

```
n = 1
sample(1:6, n, replace=TRUE) # random sampling with replacement
```

4

Part b)

In our initial problem, we said that X comes from a fair die, meaning each value is equally likely to be rolled. Because our die has 6 sides, each side should appear about $1/6^{th}$ of the time. How would we confirm that our simulation is fair?

What if we generate multiple instances of X? That way, we could compare if the simulated probabilities match the theoretical probabilities (i.e. are all $1/6$).

Generate 12 instances of X and calculate the proportion of occurances for each face. Do your simulated results appear to come from a fair die? Now generate 120 instances of X and look at the proportion of each face. What do you notice?

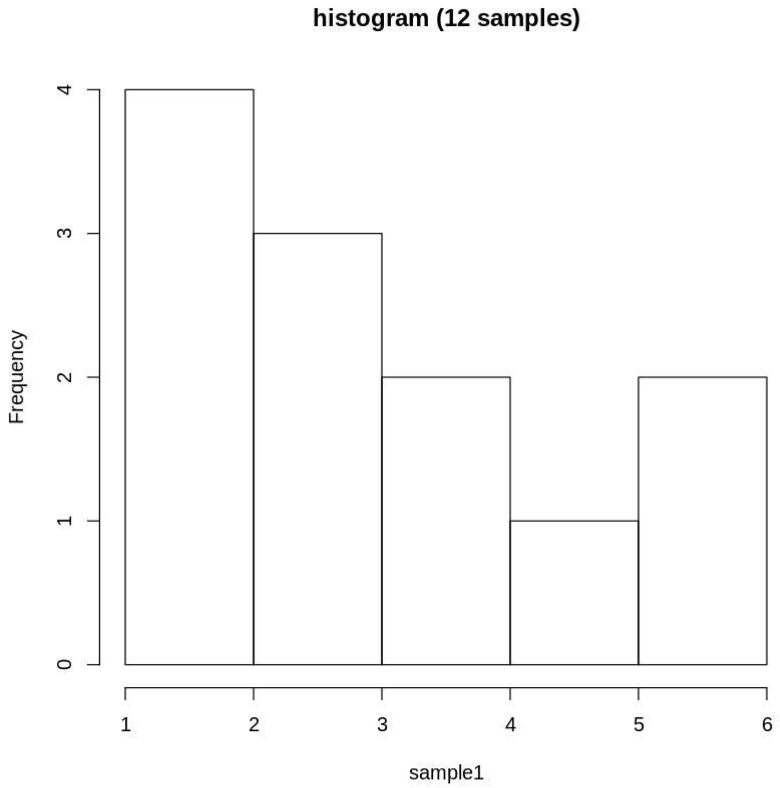
Note: Each time you run your simulations, you will get different values. If you want to guarantee that your simulation will result in the same values each time, use the `set.seed()` function. This function will allow your simulations to be reproducable.

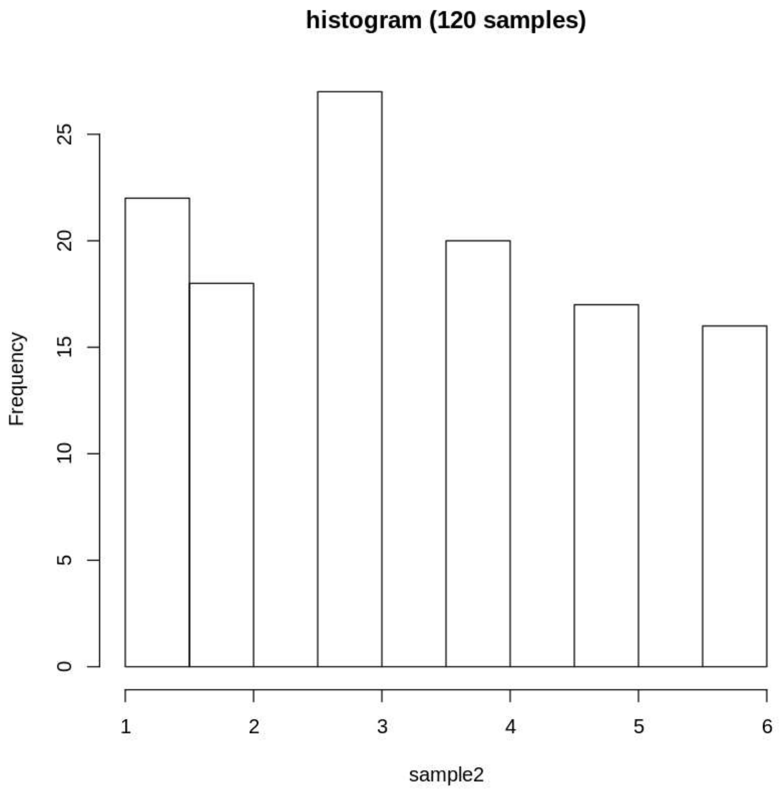
In [29]:

```
set.seed(112358)
sample1 = sample(1:6, 12, replace=TRUE) # random sampling with replacement
table(sample1) / sum(table(sample1))
hist(sample1, main='histogram (12 samples)')
sample2 = sample(1:6, 120, replace=TRUE)
table(sample2) / sum(table(sample2))
hist(sample2, main='histogram (120 samples)')
```

```
sample1
         1          2          3          4          5          6 
0.08333333 0.25000000 0.25000000 0.16666667 0.08333333 0.16666667 
```

```
sample2
        1         2         3         4         5         6 
0.1833333 0.1500000 0.2250000 0.1666667 0.1416667 0.1333333 
```





As can be seen from the above output, the simulation results seem to come from a fair die (becomes more uniform) as we increase the sample size from 12 to 120.

Part c)

What if our die is not fair? How would we simulate that?

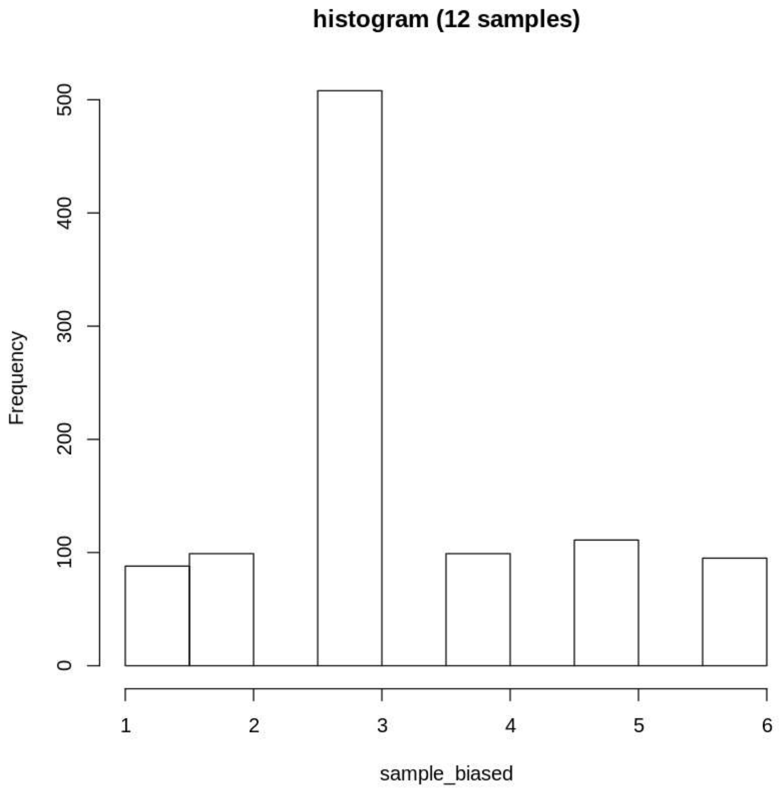
Let's assume that Y comes from an unfair six-sided die, where $P(Y = 3) = 1/2$ and all other face values have an equal probability of occuring. Use the `sample()` function to simulate this situation. Then display the proportion of each face value, to confirm that the faces occur with the desired probabilities. Make sure that n is large enough to be confident in your answer.

In [37]:

```
set.seed(112358)
probs = c(1/10,1/10,1/2,1/10,1/10,1/10)
sum(probs)
sample_biased = sample(1:6, 1000, prob=probs, replace=TRUE)
table(sample_biased) / sum(table(sample_biased))
hist(sample_biased, main='histogram (12 samples)')
```

1

```
sample_biased
    1     2     3     4     5     6 
0.088 0.099 0.508 0.099 0.111 0.095 
```



As can be seen from above, $Y = 3$ occurs with probability 0.508 which is quite close to $\frac{1}{2}$, as desired.