Consider the following:

```
for(i in 1:3){
  x <- paste0("Number ",i)
  print(x)
}
```

```
## [1] "Number 1"
## [1] "Number 2"
## [1] "Number 3"
```

And...

```
planets <- c("Mercury","Venus","Earth","Mars","Jupiter","Saturn","Uranus","Neptune")
n <- 1
newvector <- c()
for(i in planets){
  newvector[n] <- paste0(i,i)
  n=n+1
  }
newvector
```

```
## [1] "MercuryMercury" "VenusVenus"     "EarthEarth"     "MarsMars"
## [5] "JupiterJupiter" "SaturnSaturn"   "UranusUranus"   "NeptuneNeptune"
```

Exercise 1

Write a for loop that iterates over the numbers 1 to 7 and prints the cube of each number using print().

Exercise 2

Write a for loop that iterates over the column names of the inbuilt iris dataset and print each together with the number of characters in the column name in parenthesis. Example output: Sepal.Length (12). Use the following functions print(), paste0() and nchar().

Exercise 3

```
i <- 1
while (i < 6) {
print(i)
i = i+1
}
```

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

Write a while loop that prints out standard random normal numbers (use rnorm()) but stops (breaks) if you get a number bigger than 1.

Exercise 4

Using a for loop simulate the flip a coin twenty times, keeping track of the individual outcomes (1 = heads, 0 = tails) in a vector that you preallocte.

Exercise 5

Use a while loop to investigate the number of terms required before the product 1*2*3*4... reaches above 10 million.

Exercise 6

Use a while loop to simulate one stock price path starting at 100 and random normally distributed percentage jumps with mean 0 and standard deviation of 0.01 each period. How long does it take to reach above 150 or below 50?