while Loops

CSCI21
Mark Kazakevich
while loops

- A **while** loop is a statement that allows us to repeat code when we don’t know the specific amount of times we want to go through the loop.

- The number of times it repeats depends on a truth condition, which must be **true** for the loop to continue.

- The variables involved in the condition can change in the loop until it is **false** and we leave the loop.

- Let’s take a closer look.
while loop Format

```
while (condition) {
    # loop body
}
```

This block is considered one while loop

Let’s talk about what these words all mean
while loop Format

```
while (condition) {
    # loop body
}
```

while

Indicates that this is a while loop statement
while loop Format

```java
while (condition) {
    # loop body
}
```

**condition**

condition is an expression that evaluates to a **logical value**.

We continue executing the loop as long as condition is true. “While condition is true, keep repeating the loop.”
**while loop Format**

```java
while (condition) {
    # loop body
}
```

**# loop body**
- These lines of code (which are indented in the for loop), will repeat as long as condition is true.
- Unlike a for loop, there is no variable that changes at every iteration of the loop.
- But, we can change variables involved in the condition
Let’s see an example

```r
n <- 3
while (n > 0) {
    cat(n)
    cat("\n")
    n <- n - 1
}
```
Let’s see an example

```r
n <- 3
while (n > 0) {
  cat(n)
  cat("\n")
  n <- n - 1
}
```

**while**

Indicates that this is a while loop statement
Let’s see an example

```r
n <- 3
while (n > 0) {
  cat(n)
  cat("\n")
  n <- n - 1
}
```

$n > 0$

We check if $n > 0$ evaluates to TRUE. If it does, then we run the loop body. Notice that $n$ was defined before the loop.
Let’s see an example

\begin{Verbatim}
\verb+n<-3+
\verb+while (n > 0) {
    cat(n)
    cat("\n")
    n <- n - 1
}\end{Verbatim}

In the loop body, we output the value of \texttt{n} to the console. Notice how we also \textbf{decrement} the value of \texttt{n} by 1. By subtracting \texttt{n} by 1, we are changing a variable used in the \texttt{while} loop condition.

\textbf{Reminder:}\n
“\texttt{\n}” is a string with the newline character. It has nothing to do with the variable \texttt{n}.
Let’s see an example

```r
n <- 3
while (n > 0) {
  cat(n)
  cat("\n")
  n <- n - 1
}
```

In the loop body, we output the value of `n` to the console. Notice how we also **decrement** the value of `n` by 1. By subtracting `n` by 1, we are changing a variable used in the while loop condition.
```r
n <- 3
while (n > 0) {
  cat(n)
  cat("
"")
  n <- n - 1
}
```

**1st iteration of loop:**
- Current value of n: 3
- n > 0 is TRUE
- So we run the loop body

R Console output after running loop body:

```
3
```
Running the example

```r
n <- 3
while (n > 0) {
  cat(n)
  cat("\n")
  n <- n - 1
}
```

**2nd iteration of loop:**
Current value of n: 2

n > 0 is TRUE
So we run the loop body

R Console output after running loop body:

```
3
2
```
Running the example

```
 n <- 3
while (n > 0) {
  cat(n)
  cat("\n")
  n <- n - 1
}
```

**3rd iteration of loop:**
Current value of n: 1
n > 0 is TRUE
So we run the loop body

R Console output after running loop body:

```
 3
 2
 1
```
Running the example

n <- 3
while (n > 0) {
    cat(n)
    cat("\n")
    n <- n - 1
}

4th iteration of loop:
Current value of n: 0
n > 0 is FALSE
The while condition is no longer true - we do not run the loop body

R Console output after running loop body:

3
2
1
Running the example

n <- 3
while (n > 0) {
  cat(n)
  cat("\n")
  n <- n - 1
}

# program continues

The while condition is no longer true. We’re done! We now move on to the statements after the for loop

3
2
1
Something to be careful about

n <- 3
while (n > 0) {
  cat(n)
  cat("\n")
  n <- n + 1
}

- Be careful with what you do to variables involved in your while condition.
- Assigning the wrong thing can lead to an incorrect number of iterations, or..
- Infinite loops

If we always add 1, we will never fail the condition, so the loop will keep going indefinitely.
Examples in RStudio