4. Control structure of R

```r
# Reference book: "Beginning R: The Statistical Programming Language"
# Author: Dr. Mark Gardener
```

control Structure

```r
#1. if and else: testing a condition and acting on it
#2. for: excecute a loop a fixed number of time
#3. while: excecute a loop while a condition is true
#4. repeat: execute an infinite loop (must break out of it to stop)
#5. break: break the execution of a loop
#6. next: skip an interation of a loop
```

if structure

```r
#Ex1:
x=4
if (x<5){
  cat("Value of x is less than 5")
}
```

```
Value of x is less than 5
```

```r
#Ex2:
x=1
if (x>3){
  y=10
} else {
  y=0
}
cat("Value of y", y)
```

```
Value of y 0
```

```r
#Ex3:
x=12
if (x<3){
  cat("Value of x is less than 3")
} else if(x>=3 & x<10) {
  cat("Value of x is between 3 and 10")
} else {
  cat("Values of x is greater than 9")
}
```

```
Values of x is greater than 9
```

for loop

```r
#Ex1:
for (i in 1:10){
  print(i)
}
```

```
[1] 1
[1] 2
[1] 3
[1] 4
[1] 5
[1] 6
[1] 7
[1] 8
[1] 9
[1] 10
```

```r
#Ex2:
x=c("a","b","c","d")
for (i in 1:4) {
  print(x[i])
}
```

```
[1] "a"
[1] "b"
[1] "c"
[1] "d"
```

```r
#OR
for (i in 1:4) print(x[i])
```

```
[1] "a"
[1] "b"
[1] "c"
[1] "d"
```

```r
#Ex3:
#seq_along generates an integer sequence based on length of an object
x=c("a","b","c","d","e")
for (i in seq_along(x)) {
  print(x[i])
}
```

```
[1] "a"
[1] "b"
[1] "c"
[1] "d"
[1] "e"
```

```
#Ex4:
x=matrix(1:6,2,3)
for (i in seq_len(nrow(x))){
  for (j in seq_len(ncol(x))) {
    print(x[i,j])
  }
}
```

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

while loop: testing condition

```
count=0
while (count<10) {
  print(count)
  count=count+1
}
```

```
[1] 0
[1] 1
[1] 2
[1] 3
[1] 4
[1] 5
[1] 6
[1] 7
[1] 8
[1] 9
```

repeat loops

```
#Ex1:
x=1
repeat {
  print(x)
  x=x+1
  if (x==6){
    break
  }
}
```

```
[1] 1
[1] 2
[1] 3
```

```
[1] 4
[1] 5
```

```r
#Ex2:
v=c("Hello","loop")
cnt=2
repeat{
  print(v)
  cnt=cnt+1
  if (cnt>5){
    break
  }
}
```

```
[1] "Hello" "loop"
[1] "Hello" "loop"
[1] "Hello" "loop"
[1] "Hello" "loop"
```

break: A break statement is used inside a loop (repeat, for, while) to stop the iterations and flow the control outside of the loop.

```r
x = 1:5
for (val in x) {
if (val == 3){
break
}
print(val)
}
```

```
[1] 1
[1] 2
```

next: A next statement is useful when we want to skip the current iteration of a loop without terminating it.

```r
x = 1:5
for (val in x) {
if (val == 3){
next
}
print(val)
}
```

```
[1] 1
[1] 2
[1] 4
[1] 5
```

Functions

```
#Ex1: Empty function
f=function(){
  }
class(f)
```

```
[1] "function"
```

```
#Ex2: function without argument
f=function(){
  cat("Hello World!\n")
}
f()
```

```
Hello World!
```

```
#Ex3: function with argument
f=function(num){
  for (i in seq_len(num)){
    cat("Hello world!\n")
  }
}
f(3)
```

```
Hello world!
Hello world!
Hello world!
```

```
#Ex4: function with one argument
f=function(x){
  x^2+3*x-2
}
f(2)
```

```
[1] 8
```

```
#Ex5: function with two arguments
sec=function(a,b) {
  a^2+b*3
}
sec(2,3)
```

```
[1] 13
```