if else

```
if (condition) {
        ## do something
} else {
        ## do something else
}
if (x > 3) {
        y <- 10
} else {
        y <- 0
}
```

for loop

```
for (i in 1:10) {
        print(i)
}
## The following loops have the
same behavior
x <- c("a","b","c","d")
for (i in 1:4) {
        print(x[i])
}
for (i in seq_along(x)) {
        print(x[i])
}
for (letter in x) {
        print(letter)
}
## If a for loop has a single
expression
for (i in 1:4) print(x[i])
```

while loop

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

repeat, next, break

```
x0 <- 1
tol <- 1e-8
repeat {
        x1 <- somefunction()
        if (condition) {
                break
        } else {
                (another
condition)
        }
}
for (i in 1:100) {
        if (i <= 20) {
                # Skip the
first 20 iterations
                next
        }
        ## Do something here
}
```

function

```
add2 <- function(x + y) {
        x + y
}
## A function always returns the
last expression
above <- function(x, n) {
        use <- x > n
        x[use]
}
```