Foundations of R Software

Lecture 19

Basics of Calculations

::::

Conditional Executions – If and If-Else

Shalabh

Department of Mathematics and Statistics

Indian Institute of Technology Kanpur

Control structures in R :

Control statements,

Functions,

Loops.

1. Conditional execution: if()

Syntax

```
if (condition) {execute commands if condition is TRUE}
```

The code in the `{}` is evaluated if the logical test contained in the `( )` is `TRUE`.

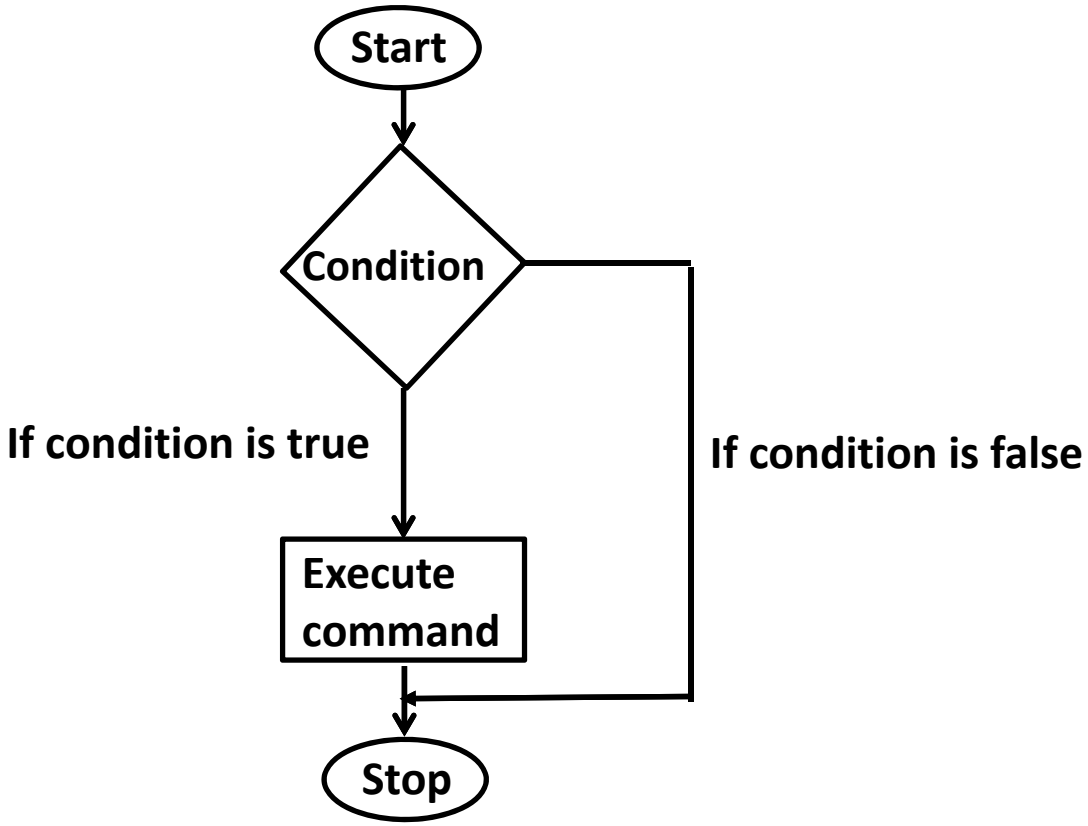
If the logical test is `FALSE`, R will ignore all of the code in the `{}`.

`if()` should not be applied when the `condition` being evaluated is a vector. It is best used only when meeting a single element condition.

1. Conditional execution: if()

Syntax

```
if (condition) {execute commands if condition is
TRUE}
```



1. Conditional execution: if()

Example 1:

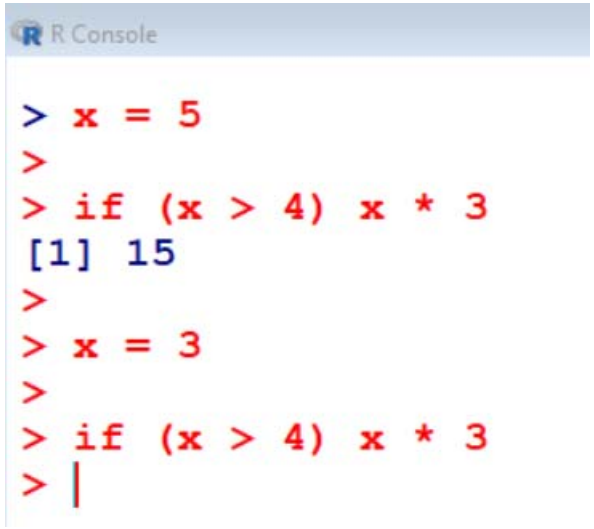
Suppose we want to multiply all the values x > 4 by 3.

```
> x = 5

> if (x > 4) x * 3
[1] 15

> x = 3

> if (x > 4) x * 3
```



No response is obtained.

1. Conditional execution: if()

Example 1: Same with using {}

Suppose we want to multiply all the values x > 4 by 3.

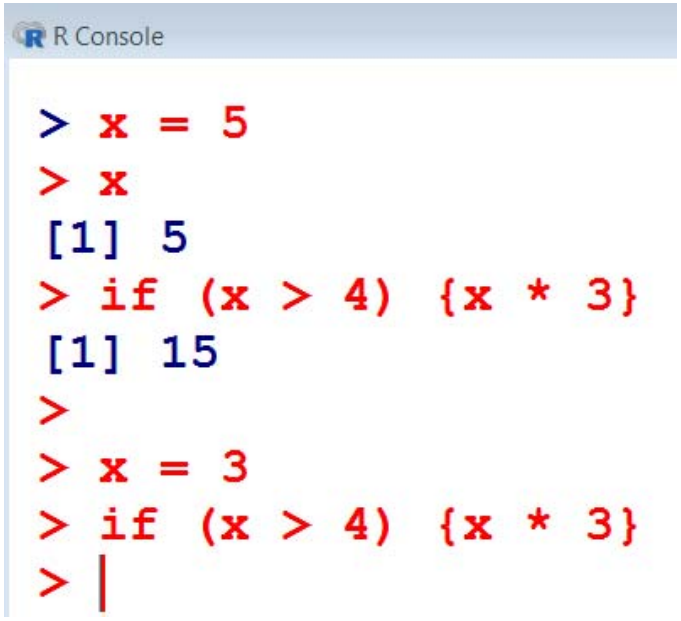
```
> x = 5

> if (x > 4) {x * 3}
[1] 15

> x = 3

> if (x > 4) {x * 3}
```

No response is obtained.

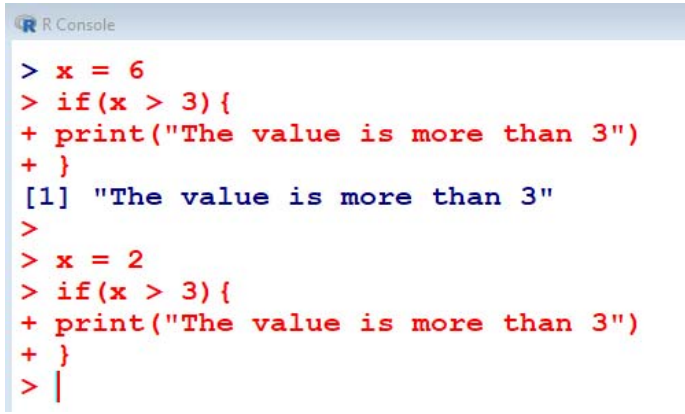


1. Conditional execution: if()

Example 2:

Suppose we want to print if a value is more than 3.

```
x = 6
if(x > 3){
print("The value is more than 3")
}
[1] "The value is more than 3"
```



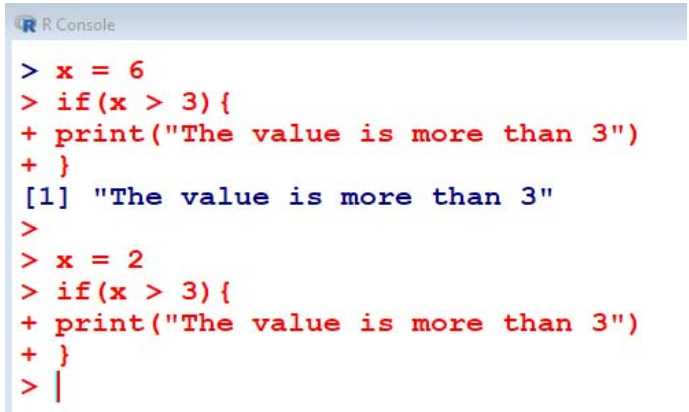
1. Conditional execution: if()

Example 3:

Suppose we want to print if a value is more than 3.

```
x = 2
if(x > 3){
print("The value is more than 3")
}
```

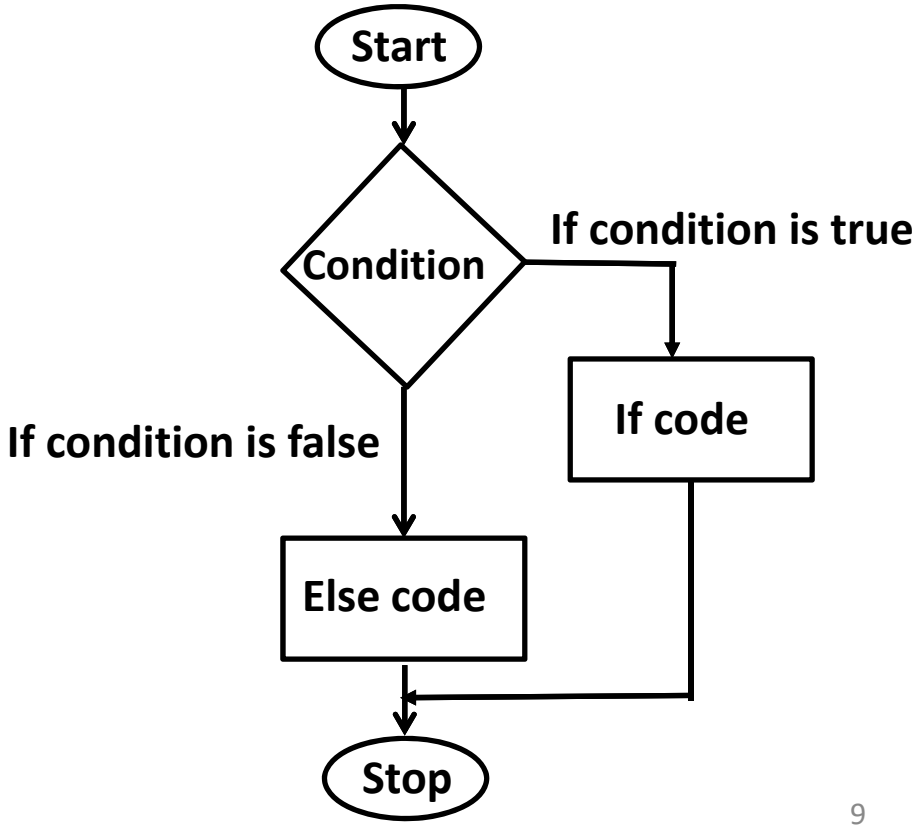
No outcome is obtained.



2. Conditional execution: if else()

Syntax

```
if (condition) {executes commands if condition is TRUE}
else { executes commands if condition is FALSE }
```



2. Conditional execution: if else()

Please note:

- **The condition in this control statement may not be vector valued and if so, only the first element of the vector is used.**

- **`if else()` should not be applied when the `condition` being evaluated is a vector. It is best used only when meeting a single element condition.**

- **The condition may be a complex expression where the logical operators "and" (`&&`) and "or" (`||`) can be used.**

2. Conditional execution: if else()

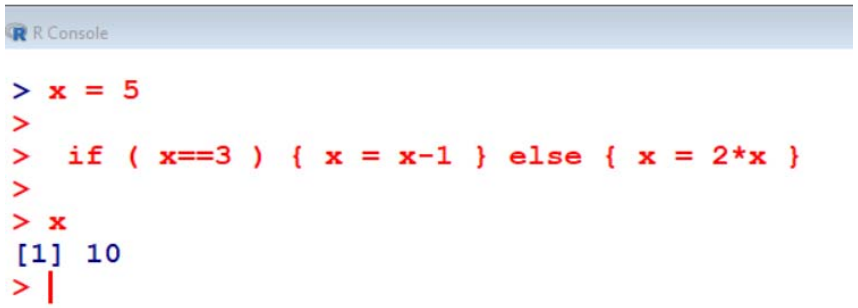
Example 4:

```
x = 5
if ( x==3 ) { x = x-1 } else { x = 2*x }
x
[1] 10
```

Interpretation:

- If x = 3, then execute x = x – 1.
- If x ≠ 3, then execute x = 2*x.

In this case, x = 5, so x ≠ 3. Thus x = 2*5



2. Conditional execution: if else()

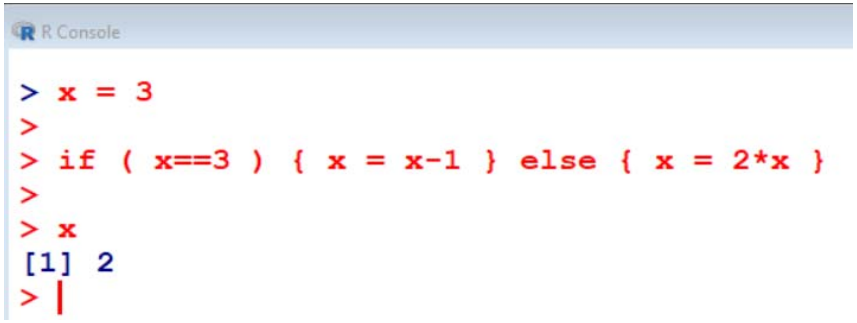
Example 5:

```
x = 3
if ( x==3 ) { x = x-1 } else { x = 2*x }
x
[1] 2
```

Interpretation:

- If x = 3, then execute x = x – 1.
- If x ≠ 3, then execute x = 2*x.

In this case, x = 3. Thus x = 3-1 = 2

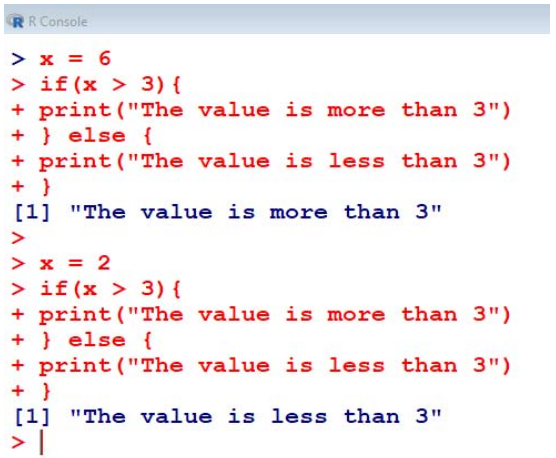


2. Conditional execution: if else()

Example 6:

Suppose we want to print if a value is more than 3 or less than 3.

```
x = 6
if(x > 3){
print("The value is more than 3")
} else {
print("The value is less than 3")
}
[1] "The value is more than 3"
```



2. Conditional execution: if else()

Example 7:

Suppose we want to print if a value is more than 3 or less than 3.

```
x = 2

if(x > 3){

print("The value is more than 3")

} else {

print("The value is less than 3")

}

[1] "The value is less than 3"
```

2. Conditional execution: if else()

Example 6:

R Console

```
> x = 6
> if(x > 3){
+ print("The value is more than 3")
+ } else {
+ print("The value is less than 3")
+ }
[1] "The value is more than 3"
>
> x = 2
> if(x > 3){
+ print("The value is more than 3")
+ } else {
+ print("The value is less than 3")
+ }
[1] "The value is less than 3"
>
```