Data types

You have seen that a variable can store a *number*. However, several other data types are available. Usually, you do not need to declare these; they will be assigned automatically.

Character data A character object is represented by a collection of characters between double (`"`) or single (`'`) quotes. For example: `"x"`, `'test character'` and `"Mike"`. One way to create character objects is as follows:

```
> name <- "Mike" # Character data (strings) need quotes like "this"
> name
[1] "Mike"
```

Note that double quotes in the output indicate that we are dealing with an object of type “character”.

Logical data An object of data type “logical” can have the value `TRUE` or `FALSE` and is used to indicate if a condition is true or false. Such objects are usually the result of logical expressions.

```
> q1 <- TRUE # Logical data; possible values are TRUE or FALSE
> q1
[1] TRUE
```

Note that `TRUE` and `FALSE` can be abbreviated as `T` and `F`:

```
> q2 <- F
> q2
[1] FALSE
```

However, to avoid confusion with possible variables called `F` or `T`, we recommend that you stick with `TRUE` and `FALSE`.

We can assign 9 to a variable `x` and ask whether `x` is greater or less than 10:

```
> x <- 9
> x > 10
[1] FALSE
```

The result is the logical value `FALSE` because 9 is *not* greater than 10.

```
> x < 10
[1] TRUE
```

The result is `TRUE` because 9 is less than 10.

```
> x == 9
[1] TRUE
```

Note that we used a double equality sign `==` to compare two objects. If we had used a single equality sign, we would have assigned the number 9 to the variable `x` instead. The result is `TRUE` because `x` equals 9.

We can also ask whether `x` is *unequal* to some number or expression.

```
> x != 10 #Is x unequal to 10?
```

```
[1] TRUE
```

If you wish to know whether two logical expressions are both true, you use logical operator `&`, called the AND operator:

```
> x == 10 & x > 2 #Is x equal to 10 and larger than 2?
```

```
[1] FALSE
```

This combined expression is false because the first expression (`x == 10`) is false.

If you wish to know whether at least one of the logical expressions is true, you use logical operator `|`, called the OR operator:

```
> x == 10 | x > 2 #Is x equal to 10 or larger than 2?
```

```
[1] TRUE
```

The combined expression is true because the second expression (`x > 2`) is true.

Factor data To represent categorical data (*i.e.*, data of which the value range is a collection of code names), R uses the *factor* data type. Factor data will be treated in more detail later on after you learned about vectors.