Exercise 1

Create three three vectors with random values with the following code:

```
x <- sample(100)
y <- sample(100)
z <- sample(letters, 100, replace=TRUE) # 100 lowercase letters in random order.
```

Extract from `x` the following:

(a) The fifth element.
(b) All elements except element 10 through 30.
(c) The elements at position 1, 3, 6, 10, 15, 21, 28, 36, 45, 55, 66, 78, 91 (try to create the sequence automatically and not manually).
(d) All elements for which the corresponding element in `y` is smaller.
(e) All elements for which the corresponding element in `z` is either `c`, `r`, or `x`.
(f) The intersection of (d) and (e).
(g) The union of (d) and (e).
(h) The position of the largest element.

Exercise 2

We have learned in the lecture that we can not extract several single elements from a data frame with one call. Why? (Hint: What would the resulting structure be? Take the `iris` data set as an example and extract the the element in the fifth row and fourth column and the element in the fourth row and fifth column separately).
Another sample dataset that comes with R is called `USArrests`, which holds number of crimes per 10,000 people for all states in the US. You can get additional informations about the dataset using the help system and the known functions for data frames! Extract the following informations from the dataset:

(a) All rows with 10 or more murders per 10,000 people.
(b) All rows with rape rates between 20 and 40.
(c) The intersection of (a) and (b).
(d) The urban population in states with more than 200 assaults as a *vector*.

Bonus exercise: `order()` returns positions for its argument that by default arranges that argument in ascending order. So for `c(8, 4, 9)` the function returns `c(2, 1, 3)`. Use this function to sort the dataset by the number of assaults.

*Exercise 3

This exercise is for the upcoming lecture to prepare you for the content or to sensibilize you for a certain aspect.

Starting next week we will start to read data from files. For this, we have to make sure that the working directory is properly set. Create a new folder on your computer called `corona`. Create a new text file called `analysis.R` in this folder and a sub-folder called `data`. Provided that RStudio is closed the working directory will automatically be set when you start RStudio by double-clicking the file. You can see the current working directory by typing `getwd()` and set the working directory with `setwd()`. Try from different locations on your computer to set the working directory correctly. Hint: When you start to type the path in `setwd()` and hit the tabulator key, RStudio will give you all available folders, from which you can choose the correct one.