**Файл «Variables.R»**

**Задание 1**

В векторе **my\_vector** отберите только те наблюдения, которые отклоняются от среднего больше чем на полтора стандартных отклонения. Сохраните эти наблюдения в новую переменную **my\_vector\_2**.

При этом исходный вектор **my\_vector** оставьте без изменений:

21 18 36 19 25 20 17 17 18 22 17 18 18 54 19 27 21 20 24 17 15 24 24 29 1 14 21 17 19 18 4 20 21 21 19 19 17 21 13 17 47 23 15 23 24 16 17 72 24 22

**Файл «data frames.R»**

**Задание 2**

В датафрэйме **mtcars** создайте новую колонку (переменную) под названием **even\_gear**, в которой будут единицы, если значение переменной (**gear**) нечетное, и нули если количество четное.

**Задание 3**

В датафрэйме **mtcars** создать переменную ­­– вектор **mpg\_6** и сохранить в нее значения расхода топлива (**mpg**) для машин с шестью цилиндрами (**cyl**) и автоматической коробкой передач.

**Задание 4**

Создать новый dataframe под названием **mini\_mtcars**, в котором будут сохранены только вторая, седьмая, девятая, двенадцатая и последняя строчка датафрейма **mtcars**.

**Файл «conditions.R»**

**Задание 5**

Создайте новую числовую переменную **new\_var** в данных mtcars, которая содержит единицы в строчках, если в машине меньше четырёх карбюраторов (переменная "**carb**") или больше шести цилиндров (переменная "**cyl**"). В строчках, в которых условие не выполняется, должны стоять нули.

**Задание 6**

Впеременной **my\_vector** сохраните вектор из 50 чисел:

20.67 23.34 22.65 17.11 22.1 26.32 20.39 21.04 23.78 31.11 21.13 22.44 23.21 27.02 18.64 20.9 20.77 20.0 21.29 23.48 18.47 25.02 17.04 30.97 12.91 23.88 32.95 8.46 23.15 21.05 20.63 19.95 17.38 29.35 24.43 23.66 18.32 30.13 19.36 19.67 24.23 20.82 18.21 9.91 21.45 18.04 18.31 17.18 10.99 10.06

Решите задачу используя конструкцию:

if () {

} else {

}

Если среднее значение вектора **my\_vector** больше 20, в переменную **result** сохраните "My mean is great", если среднее значение **my\_vector** меньше или равно 20 то в переменную **result** сохраните строку "My mean is not so great".

**Задача 7**

В этой задаче от вас потребуется узнать некоторую информацию о типах данных в R самостоятельно! Встроенные в R данные AirPassengers - это новый для нас формат данных типа Time-Series. Изучите структуру этих данных, прежде чем начать решение задачи! Например напишите команды:

> ?AirPassengers # справка о данных

> str(AirPassengers) # структура данных

Во встроенных в R данных AirPassengers хранится 144 значения (количество пассажиров в месяц) с 1949 по 1960 год. Данные Time-Series очень похожи на вектор по своей структуре, например мы можем обратиться к любому из 144 элементов используя уже знакомую нам индексацию AirPassengers[1] или AirPassengers[56].

Можно вообще перевести исходные данные в вектор при помощи команды as.vector(AirPassengers) и продолжить с ними работу как с вектором.

И так ваша задача создать переменную good\_months и сохранить в нее число пассажиров только в тех месяцах, в которых это число больше, чем показатель в предыдущем месяце.

Важный момент! В R оператор : для создания последовательности имеет приоритет над арифметическими действиями. Таким образом, если у вас есть переменная i, равная 10, и вы хотите создать вектор от 1 до i - 1, воспользуйтесь скобками, чтобы указать последовательность действий.

> i <- 10

> 1 : i - 1 # так мы создадим последовательность от 1 до 10, а потом вычтем единицу из каждого элемента

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

> 1 : (i - 1) # а вот так мы создадим последовательность от 1 до i - 1, то есть от 1 до 9.

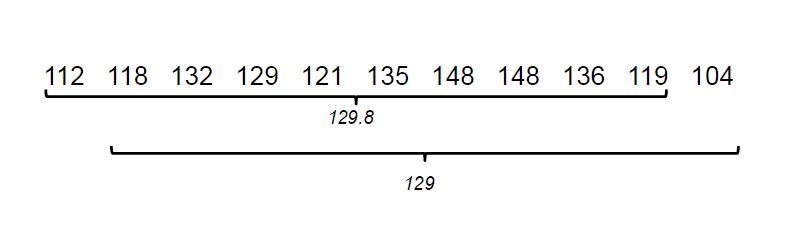
[1] 1 2 3 4 5 6 7 8 9

**Задача 8**

Для встроенных в R данных AirPassengers рассчитайте скользящее среднее с интервалом сглаживания равным 10. Напечатайте получившийся результат (первым значением в выводе должно быть среднее для элементов 1:10, во втором значении - среднее для элементов 2:11 и т.д., в последнем  - среднее для элементов 135 :144)

Все полученные значения средних сохраните в переменную moving\_average.

Пример расчета для вектора из 11 элементов:



Соответственно, для наших данных из 144 наблюдений должно получиться 135 средних (первые и последние 4 средних):

129.8 129.0 129.0 127.3 ... 483.6 489.2 486.5 490.6

Если вам потребуется создать вектор moving\_average заранее, то есть несколько способов сделать это:

1. самый простой, но не очень правильный вариант - создать пустой вектор

moving\_average <- c()

2. можно сразу создать вектор определенной длины и определенного типа:

moving\_average <- numeric(135)

Такой вариант является более предпочтительным. Также можно познакомиться с функцией cumsum. Подсказка: если у нас есть два вектора одинаковой длинны, то если из одного вектора вычесть второй вектор, мы найдем разность для первых элементов векторов, затем для вторых и т.д.

> x <- c(2, 4, 7)

> y <- c(2, 3, 5)

> x - y

[1] 0 1 2