

ПРАКТИЧЕСКАЯ РАБОТА 6

Тема УПРАВЛЯЮЩИЕ КОНСТРУКЦИИ. ФУНКЦИИ. ВЕКТОРИЗОВАННЫЕ ВЫЧИСЛЕНИЯ

Управляющие конструкции

Задача 1.

В переменной `exam` сохранена оценка студента за экзамен (в 5 балльной шкале). Напишите код, который делает следующее:

- Если оценка выше 3, выводит сообщение «Молодец!»
- Иначе – выводит сообщение «Плохо»

Задача 2.

Дан поименованный вектор `coffee`: `coffee <- c(Ann = 2, Alice = 3, Rowena = 1, Nick = 0)`

Пояснение. Поименованный вектор в R можно рассматривать как набор пар ключ-значение. Здесь ключами являются имена респондентов, а значениями – количество чашек кофе, которое они выпивают в среднем в день. Элементы поименованного вектора можно вызвать по ключу:

```
coffee["Ann"] # ключ в кавычках
```

```
## Ann
```

```
## 2
```

Вектор с ключами, именами респондентов, можно получить используя функцию `names()`:

```
names(coffee)
```

```
## [1] "Ann", "Alice", "Rowena", "Nick"
```

Интересующее нас имя респондента сохранено в переменной `resp` (может быть любым). Напишите код, который делает следующее:

- Если имя респондента есть в векторе ключей, выводит на экран число чашек кофе, которое он(она) выпивает в среднем в день;
- Если имя респондента отсутствует в векторе, выводит сообщение на экран "No such respondents".

Задача 3.

Для целых чисел от 1 до 15 выведите строки с текстом вида: «Квадрат числа 5 равен 25».

Подсказка. Вспомните про функцию `cat()` для склеивания частей выдачи на экран.

Задача 4.

Список файлов сохранен в вектор `mfiles`:

```
mfiles <- c("file01", "file_01", "file_02", "file03", "file05", "file_05", "file_09", "file10")
```

Известно, что файлы с подчеркиванием `_` в названии более новые, и мы хотим работать только с ними. Напишите код, который позволит отобрать такие файлы в вектор `to.work`.

Подсказка. Для начала можно создать пустой вектор `to.work <- c()`. Проверить, входит ли некоторый символ в строку, можно с помощью функции `grepl()`. Пример:

```
# правда ли, что _ есть в a_b
```

```
grepl("_", "a_b")
```

```
## [1] TRUE
```

Функции

Задача 1

Напишите функцию `bmi()`, которая запрашивает у пользователя с клавиатуры его рост в сантиметрах и вес в килограммах (считайте, что пользователь вводит два числа отдельно, с новой строки) и возвращает индекс массы тела, округленный до целого значения. Индекс массы тела считается так:

$$BMI = \frac{m}{h^2},$$

где m – масса тела в килограммах, h – рост в метрах.

Подсказка. Для запроса значения с клавиатуры в R используется функция `readline()`.

Задача 2

Напишите функцию `hello()`, которая просит пользователя ввести свое имя, и если оно совпадает Вашим, выводит на экран “[Name], welcome to R! I missed you”, если не совпадает, то выводит на экран “Sorry, R is not available now”. Подразумевается, что R знает Ваше имя (оно сохранено в отдельную переменную).

Векторизованные вычисления

Задача 1

Дан вектор `dat`, состоящий из трех числовых векторов с баллами студентов за тесты (вектор с вложенной структурой):

```
dat <- c(c(12.5, 40, 45.5, 48),
        c(25, 45, 30.5, 47),
        c(37.5, 28, 30, 45))
```

1.1. Запустите код `mean(dat)`.

1.2. Запустите код `sapply(dat, mean)`.

1.3. Сравните результаты пункта 1.1 и 1.2. Сделайте вывод о том, как работает функция `mean()` при применении к векторам с вложенной структурой.

Задача 2

Дан список с названиями кинотеатров:

```
kino <- list(c(1, "кинотеатр «Октябрь»"),
            c(2, "кинотеатр «Пионер»"),
            c(3, "кинотеатр «Балтика»"))
```

Используя функции для обработки строк и векторизованные операции, получите следующий вектор из названий кинотеатров:

```
## ОКТЯБРЬ ПИОНЕР БАЛТИКА
```

Подсказка. Чтобы сделать все буквы заглавными, понадобится функция `toupper()` (для обратной операции существует функция `tolower()`).