

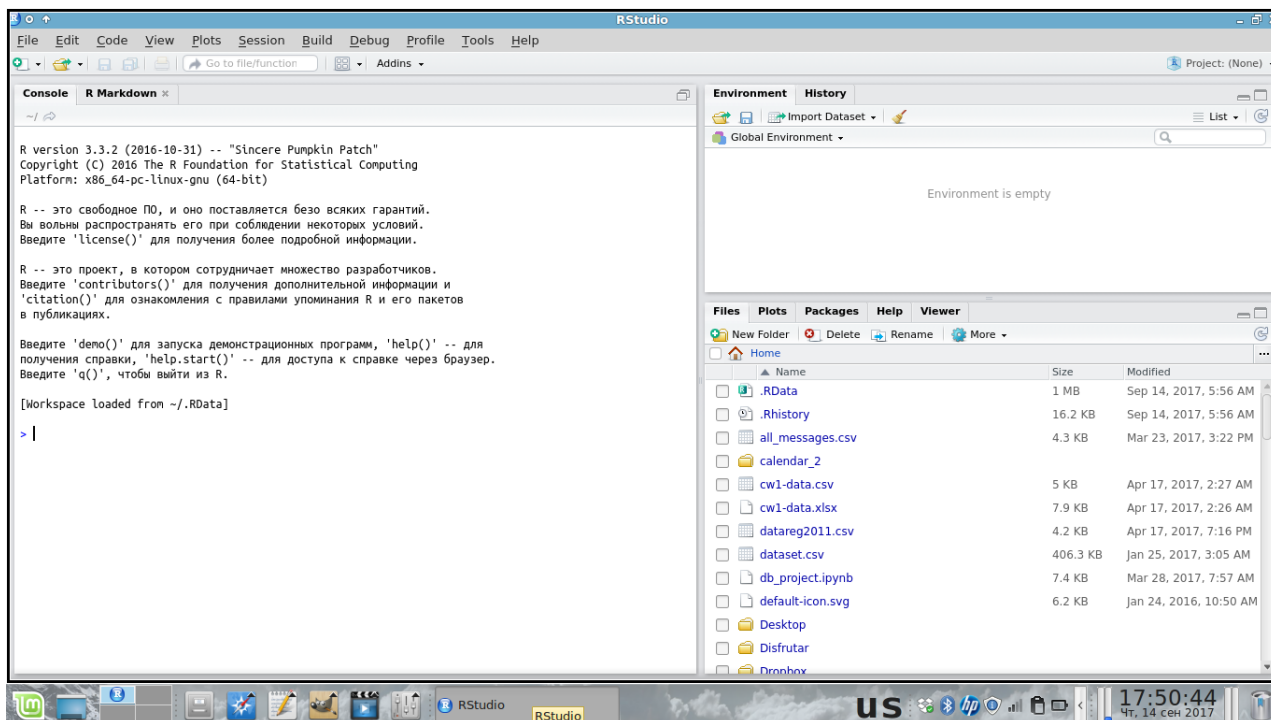
Знакомство с RStudio

Небольшая инструкция в формате вопросов-ответов по работе в RStudio. Предполагается, что R и RStudio уже установлены.

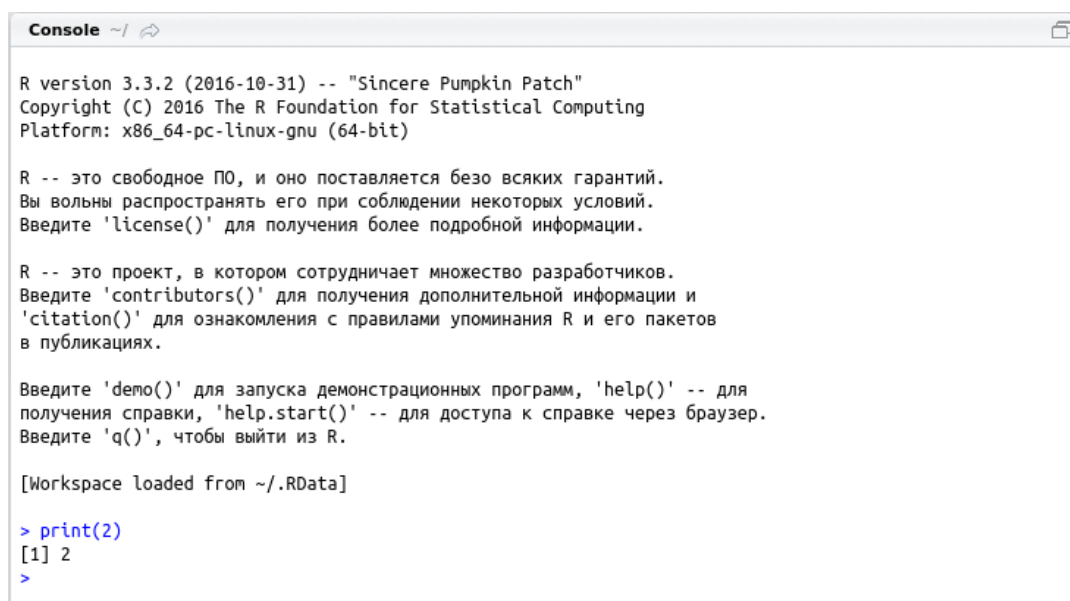
1 Интерфейс RStudio

1.1 Что происходит, когда мы запускаем RStudio?

1. Открывается рабочая область, которая выглядит примерно так:



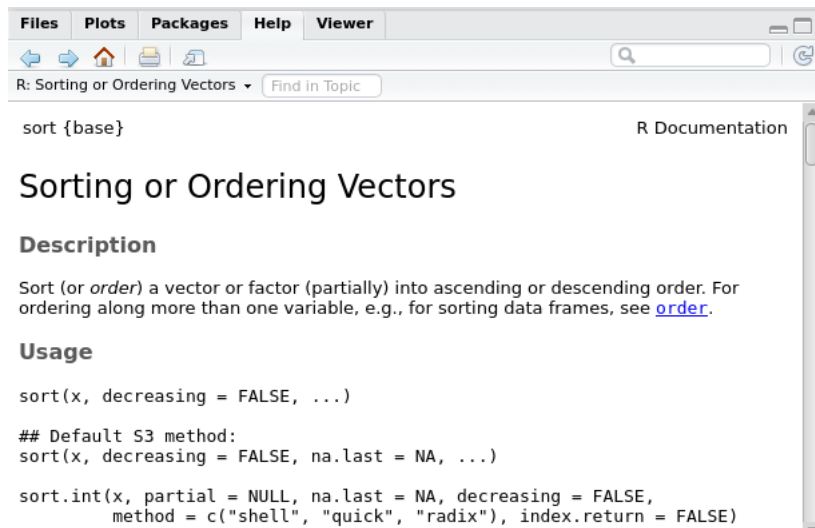
2. В окне консоли (**Console**) мы можем вводить код и сразу получать результат. Для того, чтобы исполнить код, введенный в окне консоли, нужно нажать *Enter*.



3. В правом верхнем окне (**Environment** | **History**) отображаются сохраненные во время работы в R объекты. Например, мы создаем переменную `x` и присваиваем ей значение 2. В окне **Environment** тут же появляется следующая запись:



4. В нижнем правом окне есть много вкладок. Во вкладке **Files** отображаются файлы и папки, которые находятся в рабочей папке (папке, из которой запускается R). Во вкладке **Plots** отображаются графики, которые мы строим в RStudio. Чтобы увеличить график для просмотра, нужно нажать *Zoom*. Графики из этого окна можно сохранять вручную, нажав кнопку *Export* и выбрав формат. Во вкладке **Help** отображается подробная информация по той функции R, которая нас интересует, когда мы обращаемся к R за помощью (например, для функции `sort()` через `?sort`):



2 Работа в RStudio

2.1 Что можно делать с помощью RStudio?

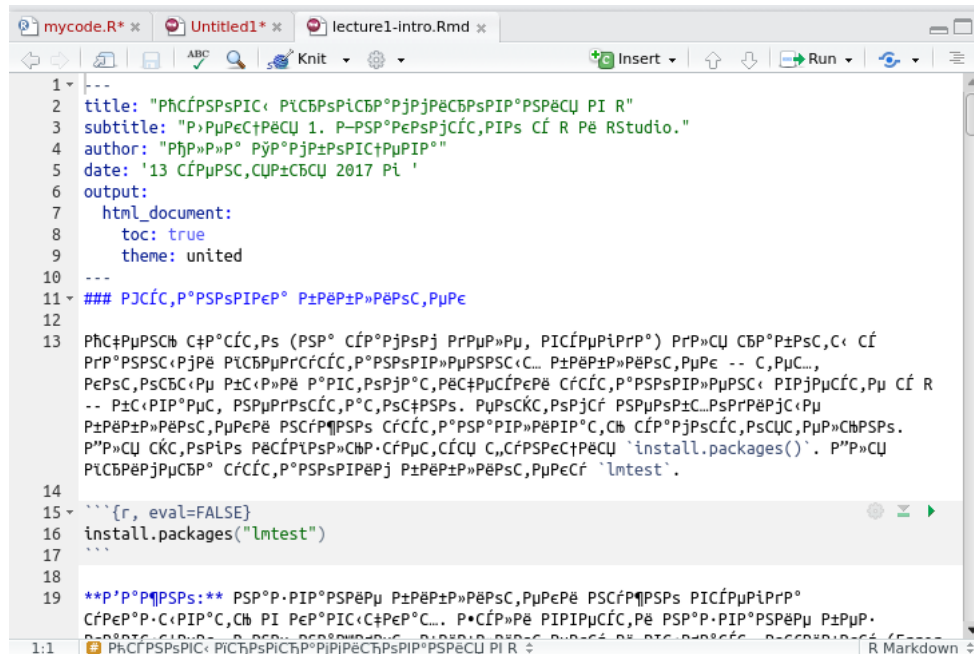
С помощью RStudio можно загружать файлы с кодом на R или создавать свои файлы (R, Rmd и не только). С помощью RStudio можно редактировать и запускать код, а также добавлять к нему текстовые комментарии.

2.2 Как загрузить файл в RStudio (например, файл с материалами лекции/семинара)?

После того как открылся RStudio, выбрать в меню: *File* → *Open*. Выбрать нужный файл и открыть.

2.3 Что делать, если загруженный Rmd-файл выглядит странно?

Например, так:



```
1 ---
2 title: "PñCÍPSPsPIC PíCBPsPíCBP°PjPjPèCBPsPIP°PSPèCJ PI R"
3 subtitle: "P»PµPèC†PèCJ 1. P–PSP°PèPSPjCíC,PiPs Cí R Pè RStudio."
4 author: "PñP»P»P° PÿP°PjP±PsPIC†PµPIP°"
5 date: '13 CíPµPSC,CµP±CèCJ 2017 Pí '
6 output:
7   html_document:
8     toc: true
9     theme: united
10 ---
11 ### PJCíC,P°PSPsPIPeP° P±PèP±P»PèPsC,PµPe
12
13 PñC†PµPSCè C†P°CíC,Ps (PSP° CíP°PjPsPj PrPµP»Pµ, PICíPµPíPrP°) PrP»CJ CèP°P±PsC,Cí Cí
14 PrP°PSPSC†PjPè PíCBPµPrCíCíC,P°PSPsPIP»PµPSPSC†C... P±PèP±P»PèPsC,PµPe -- C,PµC...,
15 PePsC,PsCèC†Pµ P±C†P»Pè P°PIC,PsPjP°C,PeC†PµCíPèPè CíCíC,P°PSPsPIP»PµPSC† PIPjPµCíC,Pµ Cí R
16 -- P±C†PIP°PµC, PSPµPrPpsCíC,P°C,PsC†PSPs. PµPscKc,PsPjCí PSPµPsP±C...PsPrPèPjC†Pµ
17 P±PèP±P»PèPsC,PµPePè PSC†PµPSPs CíCíC,P°PSP°PIP»PèPIP°C,Cè CíP°PjPsCíC,PsCUC,PµP»CèPSPs.
18 P»P»CJ CíC,PsPíPs PèCíPíPsP»CèP·CíPµC,CíCJ C,CíPSPèC†PèCJ `install.packages()`. P»P»CJ
19 PíCBPèPjPµCèP° CíCíC,P°PSPsPIPèPj P±PèP±P»PèPsC,PµPeCí `lmtest`.
20
21 ```{r, eval=FALSE}
22 install.packages("lmtest")
23 ```
24
25 **P°P°PµPSPs:** PSP°P·PIP°PSPèPµ P±PèP±P»PèPsC,PµPePè PSC†PµPSPs PICíPµPíPrP°
26 CíPèP°P·C†PIP°C,Cè PI PèP°PIC†C†PèP°C... P·CíP»Pè PIPiPµCíC,Pè PSP°P·PIP°PSPèPµ P±PµP·
27 PñCÍPSPsPIC PíCBPsPíCBP°PjPjPèCBPsPIP°PSPèCJ PI R
```

Такие странности могут происходить из-за разницы в кодировках. Rmd-файл мог быть создан на компьютере с Windows (кодировка cp1251), а открыт на Mac (кодировка utf8) или наоборот. Чтобы это поправить, нужно, не закрывая файл с крокозябрами, в меню выбрать *File* → *Reopen with Encoding*, выбрать нужную кодировку (обычно UTF-8, если Rmd-файл странно открылся на Windows, и WINDOWS-1251, если файл странно открылся на Mac) и нажать *OK*.

2.4 Как создать новый R-файл в RStudio (файл с расширением .R)?

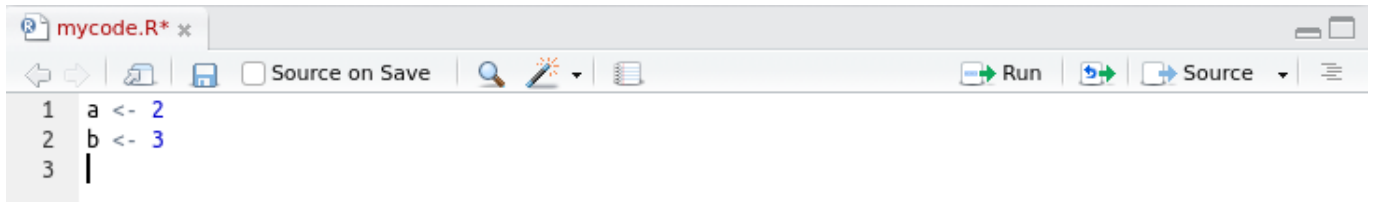
После того как открылся RStudio, выбрать в меню: *File* → *New File* → *R Script*.
Для тех, кто любит использовать быстрые клавиши, можно не обращаться к меню, а просто нажать *Ctrl + Shift + N*.
Над окном консоли откроется файл *Untitled1.R*.

2.5 Создали новый R-файл. Что дальше?

1. Сохраним файл и дадим ему название (*File* → *Save* или *Ctrl + S*).
2. В R-файле можем писать код, а также добавлять комментарии к нему.

Хотим написать код.

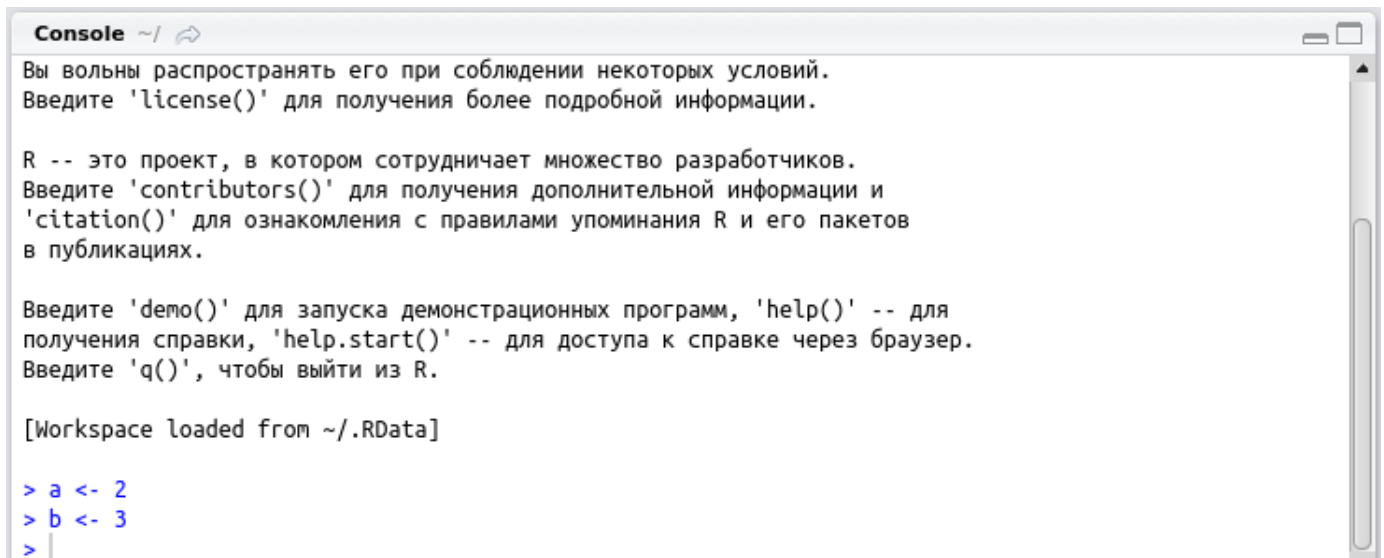
Пишем код (создаем переменные `a` и `b` и присваиваем им значения 2 и 3 соответственно):



```
1 a <- 2
2 b <- 3
3 |
```

Исполняем код: ставим курсор в любое место нужной строки и нажимаем `Ctrl+Enter` или кликаем `Run`.

Когда код исполнен, строки с кодом отображаются в окне консоли:



```
Console ~/ |
Вы вольны распространять его при соблюдении некоторых условий.
Введите 'license()' для получения более подробной информации.

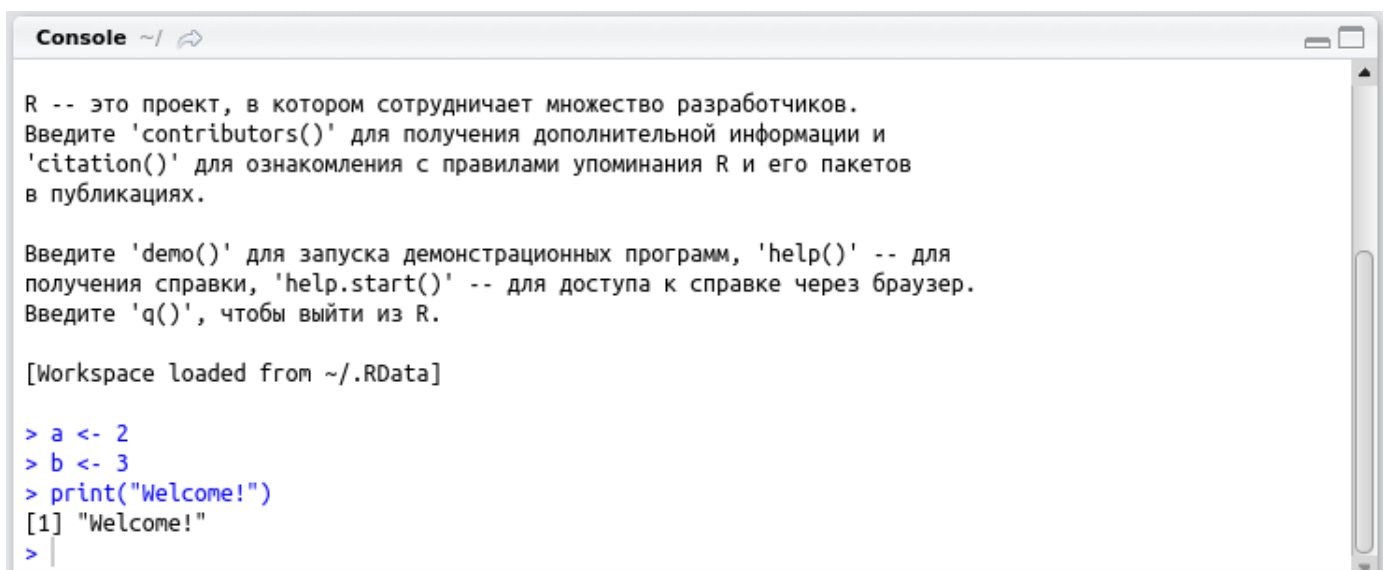
R -- это проект, в котором сотрудничает множество разработчиков.
Введите 'contributors()' для получения дополнительной информации и
'citation()' для ознакомления с правилами упоминания R и его пакетов
в публикациях.

Введите 'demo()' для запуска демонстрационных программ, 'help()' -- для
получения справки, 'help.start()' -- для доступа к справке через браузер.
Введите 'q()', чтобы выйти из R.

[Workspace loaded from ~/.RData]

> a <- 2
> b <- 3
> |
```

Если код подразумевает вывод чего-то на экран, то ожидаемый вывод тоже появится в окне консоли:



```
Console ~/ |
R -- это проект, в котором сотрудничает множество разработчиков.
Введите 'contributors()' для получения дополнительной информации и
'citation()' для ознакомления с правилами упоминания R и его пакетов
в публикациях.

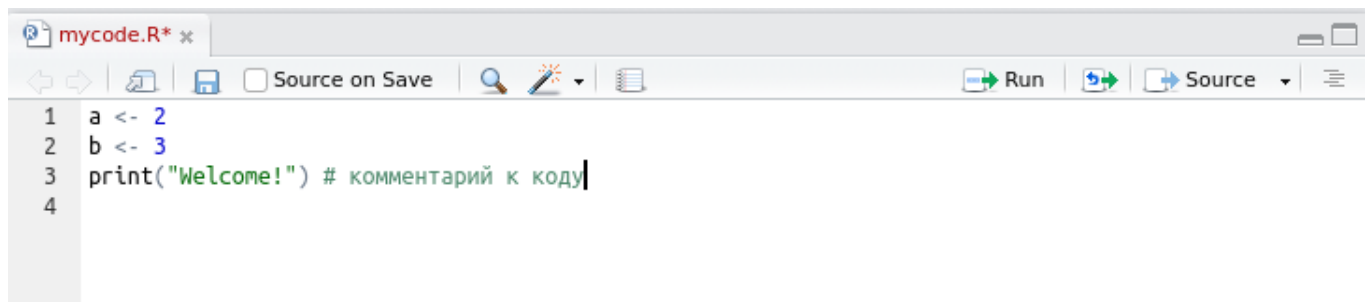
Введите 'demo()' для запуска демонстрационных программ, 'help()' -- для
получения справки, 'help.start()' -- для доступа к справке через браузер.
Введите 'q()', чтобы выйти из R.

[Workspace loaded from ~/.RData]

> a <- 2
> b <- 3
> print("Welcome!")
[1] "Welcome!"
> |
```

Хотим добавить комментарии к коду.

Комментарий к коду начинается со знака `#`. Этот символ дает понять R, что все, что следует в строке после него, не является кодом, и поэтому исполняться не должно.



```
1 a <- 2
2 b <- 3
3 print("Welcome!") # комментарий к коду
4
```

2.6 Как создать новый RMarkdown-файл в RStudio (файл с расширением `.Rmd`)?

После того как открылся RStudio, выбрать в меню: *File* → *New File* → *R Markdown*.

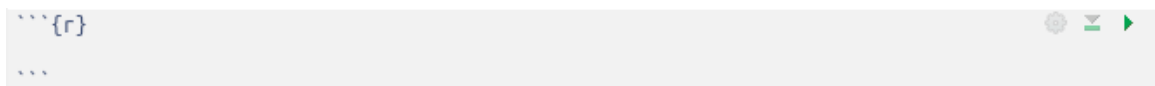
Если мы создаем Rmd-файл в первый раз, то, как только мы выберем соответствующий пункт в меню, R выдаст окно с предложением поставить необходимые библиотеки. Соглашаемся с R и немного ждем.

Дальше R выдает окно *New R Markdown*. Тип документа по умолчанию **Document**, его не изменяем. Дальше вводим заголовок документа **Title** – это не название файла, а заголовок, который будет отображаться в документе, и имя автора **Author**. Оставляем в **output format** HTML и нажимаем ОК.

2.7 Создали новый Rmd-файл. Что дальше?

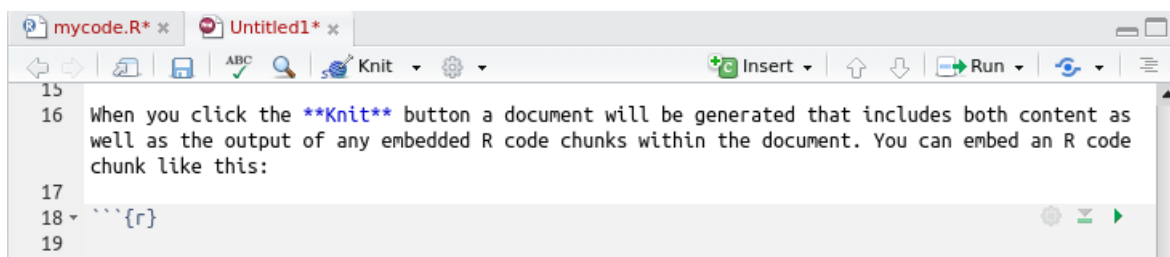
1. Сохраним файл и дадим ему название (*File* → *Save* или *Ctrl + S*).
2. В Rmd-файл мы можем добавлять красиво оформленный текст (подробнее про RMarkdown см. здесь), код, картинки и прочее.

Когда мы создаем Rmd-файл, он создается непустой – R показывает пример того, как в RMarkdown можно оформить текст и код. Блоки с кодом (*code chunks*) выделяются специальными ограничителями:



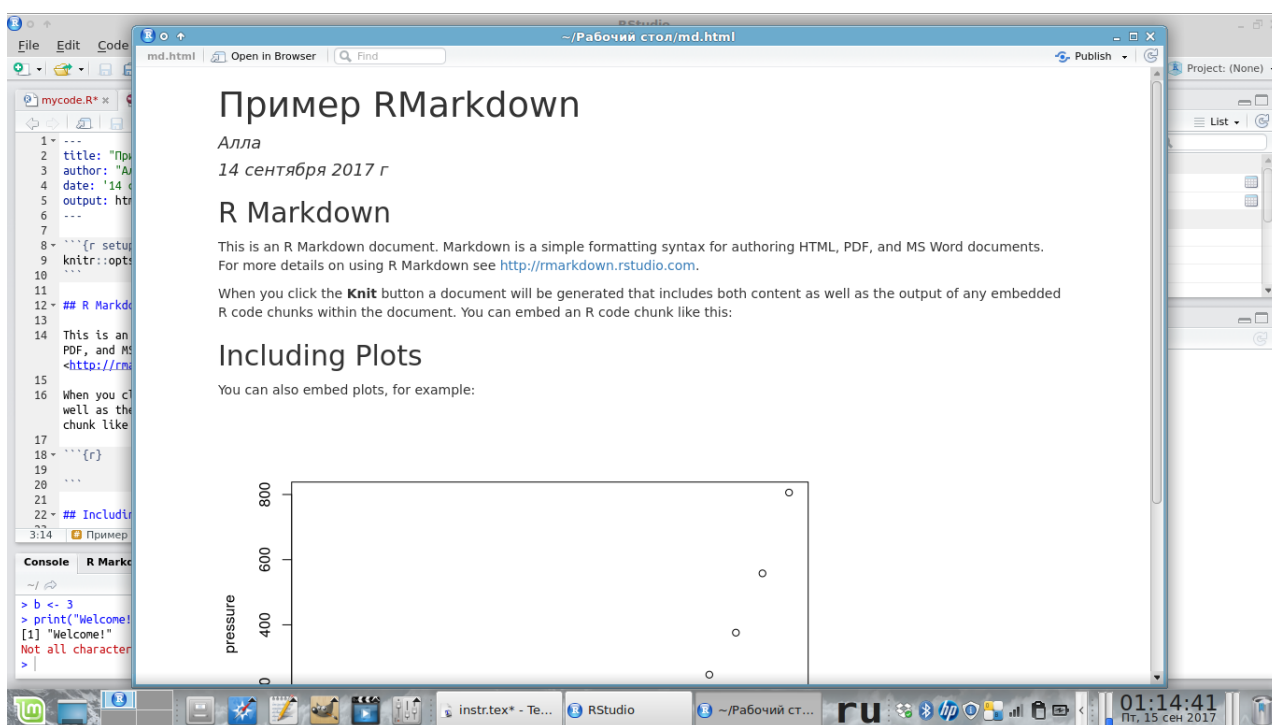
```
```{r}
...
```
```

3. Rmd-файл можно «связать» в html-файл и получить красивую html-страницу. Для этого нужно нажать **Knit**:



```
15
16 When you click the Knit button a document will be generated that includes both content as
17 well as the output of any embedded R code chunks within the document. You can embed an R code
18 chunk like this:
19 ```{r}
20 ...
```

В окне с предлагаемыми кодировками взять первую (по умолчанию), нажать ОК. После этого файл «свяжется» в html-страницу и откроется:



А что если не откроется? Возможно, при компиляции файла возникли ошибки. В окне консоли при работе с Rmd-файлом появляется вкладка **RMarkdown**. Когда html-файл «связывается», в этом окне постоянно бегут строки. Там же фиксируются ошибки: обычно указывается номер строки, в которой допущена ошибка, и пояснение. Важно понимать, что когда «связывается» html-файл, весь код, указанный в размеченных блоках `{r}` исполняется. Поэтому, если есть опечатки или ошибки в коде, html-файл не «свяжется».

2.8 Как опубликовать файл на RPubs?

Опубликовать на RPubs можно html-файл, который открывается в RStudio после того, как мы нажали **Knit**. В правом верхнем углу есть кнопка **Publish**. Когда мы ее нажимаем в первый раз, R предлагает поставить несколько библиотек. Мы с ним соглашаемся (эти библиотеки должны поставиться быстро), в появившемся окне выбираем RPubs и нажимаем **Publish**. После этого открывается браузер, и на странице нам предлагают создать аккаунт. Создаем аккаунт, заходим в него. После этого выполняем завершающий шаг: даем документу название, описание (по желанию) и адрес ссылки. Ссылка на файл будет выглядеть как `http://rpubs.com/user/filename`, где `filename` – придуманный нами адрес ссылки.

2.9 Как остановить R (R завис/запустили не тот код)?

В меню выбрать *Session* и соответствующий пункт: *Interrupt R* (прервать). Если пункт недоступен, можно выбрать *Terminate R* (завершить) или *Restart R* (перезагрузить), но тогда после весь код нужно будет исполнять заново, так как эти операции равносильны тому, что мы просто закроем RStudio и снова откроем.

2.10 Что делать, если ошиблись, и код не хочет исполняться?

Например, если в консоли в начале строки вместо привычной стрелочки стоит плюс, и R ни на что не реагирует? Например, мы записывали значения вектора и забыли поставить закрываю-

щую скобку:

```
Console ~/ ↵

R version 3.3.2 (2016-10-31) -- "Sincere Pumpkin Patch"
Copyright (C) 2016 The R Foundation for Statistical Computing
Platform: x86_64-pc-linux-gnu (64-bit)

R -- это свободное ПО, и оно поставляется безо всяких гарантий.
Вы вольны распространять его при соблюдении некоторых условий.
Введите 'license()' для получения более подробной информации.

R -- это проект, в котором сотрудничает множество разработчиков.
Введите 'contributors()' для получения дополнительной информации и
'citation()' для ознакомления с правилами упоминания R и его пакетов
в публикациях.

Введите 'demo()' для запуска демонстрационных программ, 'help()' -- для
получения справки, 'help.start()' -- для доступа к справке через браузер.
Введите 'q()', чтобы выйти из R.

[Workspace loaded from ~/.RData]

> my_vector <- c(2, 3, 4, 5
+ |
```

Чтобы вернуть R в рабочее состояние, нужно поставить курсор в строчку консоли, где стоит плюс, и нажать *Esc*. Стрелочка в начале строки вернется обратно, и можно запускать код дальше.