

# Comunicando seus resultados:

Criando apresentações com R

Beatriz Milz e Haydee Svab - R-Ladies São Paulo

20 de maio de 2019

(Atualizado em: 15/04/2019)

# Instrutoras

## Beatriz Milz



- Doutoranda em Ciência Ambiental no PROCAM/IEE/USP
- Co-organizadora do R-Ladies São Paulo
- Assessora Editorial na Revista Ambiente e Sociedade
- Bacharel em Gestão Ambiental (EACH/USP)
- Mestre em Ciências no PPG-Análise Ambiental Integrada (UNIFESP/Diadema)
- Email: [beatriz.milz@hotmail.com](mailto:beatriz.milz@hotmail.com)
- Página: <https://beatrizmilz.github.io/>
- Github: @beatrizmilz
- Twitter: @BeaMilz

# Instrutoras

## Haydee Svab



- Consultora local do Banco Mundial
- CEO da ASK-AR (consultoria em análise de dados)
- Membro do Conselho Deliberativo da AEAMESP (Associação dos Engenheiros e Arquitetos de Metrô)
- Co-fundadora dos grupos RLadies - São Paulo, do PoliGNU e da PoliGen
- Doutoranda em Smart Cities (IME-USP)
- Mestra em Engenharia e Planejamento de Transportes (Poli-USP)
- Especialista em Democracia Participativa, Repúblicas e Movimentos Sociais (UFMG)
- Formação em Eng<sup>a</sup> Civil/Arquitetura pela USP (Programa Poli-FAU USP)
- Email: [hsvab@hsvab.eng.br](mailto:hsvab@hsvab.eng.br)

# Cronograma

- 1 - O que é o R-Ladies?
- 2 - Linguagem de marcação Markdown e arquivos do tipo RMarkdown
- 3 - Como criar uma apresentação utilizando o pacote Xaringan
- 4 - Hospedagem da apresentação utilizando o Github Pages



# O que é o R-Ladies?

R-Ladies é uma organização mundial que promove a diversidade de gênero na comunidade da linguagem R. R-Ladies São Paulo integra, orgulhosamente, a organização R-Ladies Global, em São Paulo.

## Como?

Através de meetups e mentorias em um ambiente seguro e amigável.

Nosso principal objetivo é promover a linguagem computacional estatística R compartilhando conhecimento, assim, quem tiver interesse na linguagem será bem-vinda, independente do nível de conhecimento.

Fonte: [About us - R-Ladies, Meetup R-Ladies São Paulo](#)

# Para quem?

Nosso principal público-alvo são as pessoas que se identificam com o gênero feminino, portanto, mulheres cis, mulheres trans, bem como pessoas não-binárias e queer.

# Missão

Como uma iniciativa de diversidade, a missão das R-Ladies é alcançar uma representação proporcional de pessoas de gêneros atualmente sub-representados na comunidade R, incentivando, inspirando e capacitando-as.

Fonte: [About us - R-Ladies](#), [Meetup R-Ladies São Paulo](#)

# Como o R-Ladies começou?



**Gabriela de Queiroz** fundou o R-Ladies no dia **1 de outubro de 2012**. Ela queria retribuir à comunidade depois de ir a vários encontros e aprender muito de graça. O primeiro encontro R-Ladies foi realizado em **San Francisco, Califórnia (Estados Unidos)**. Nos anos seguintes, mais capítulos do R-Ladies começaram em todo o mundo.

Fonte: [About us - R-Ladies](#)



Fonte: R-Ladies Shiny App (13/04/2019)

# Código de conduta

O R-Ladies dedica-se a proporcionar uma experiência livre de assédio para todas as pessoas participantes, desta forma, não é tolerada nenhuma forma de assédio.

Código de conduta - R-Ladies

# R-Ladies no Brasil

Atualizado em: 13/04/2019. Fonte: [R-Ladies Shiny App](#)

Show  entries

Search:

	city	dt_created	members
1	Florianópolis	2019-04-07	22
2	Salvador	2018-07-23	114
3	Niterói	2018-06-04	289
4	Curitiba	2018-05-29	151
5	Belo Horizonte	2018-04-20	509
6	São Paulo	2018-02-10	343
7	Porto Alegre	2017-10-30	218
8	Natal	2017-09-25	135
9	Americana	2017-06-24	28
10	Rio de Janeiro	2017-02-27	394

Showing 1 to 10 of 10 entries

Previous

1

Next 10 / 74

# Como saber mais?

Website RLadies Global: <https://rladies.org/>

Twitter: [@RLadiesGlobal](#), [@RLadiesSaoPaulo](#)

Instagram: [@RLadiesSaoPaulo](#)

Facebook: [@RLadiesSaoPaulo](#)

MeetUp: <https://www.meetup.com/pt-BR/R-Ladies-Sao-Paulo>

R-Ladies LATAM Blog (Latin America) - Em breve!

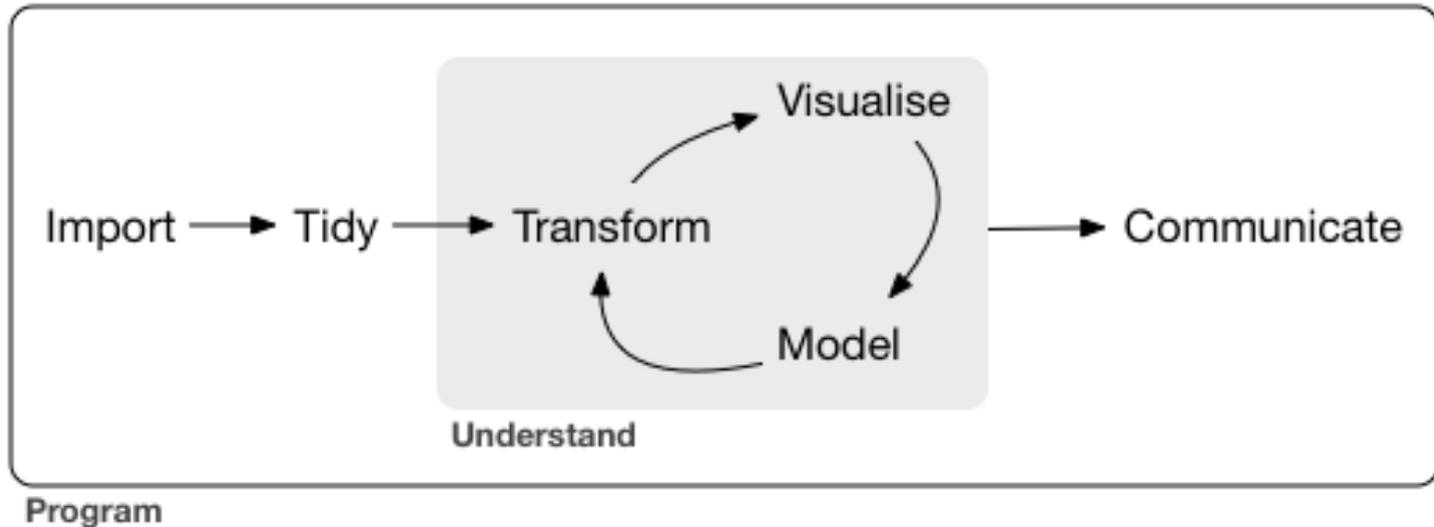
**Não tem capítulo na sua cidade e quer abrir um?**

Saiba como em [R-Ladies - How do get involved](#)

# Linguagem de marcação markdown e arquivos do tipo RMarkdown

# Por que apresentações são importantes?

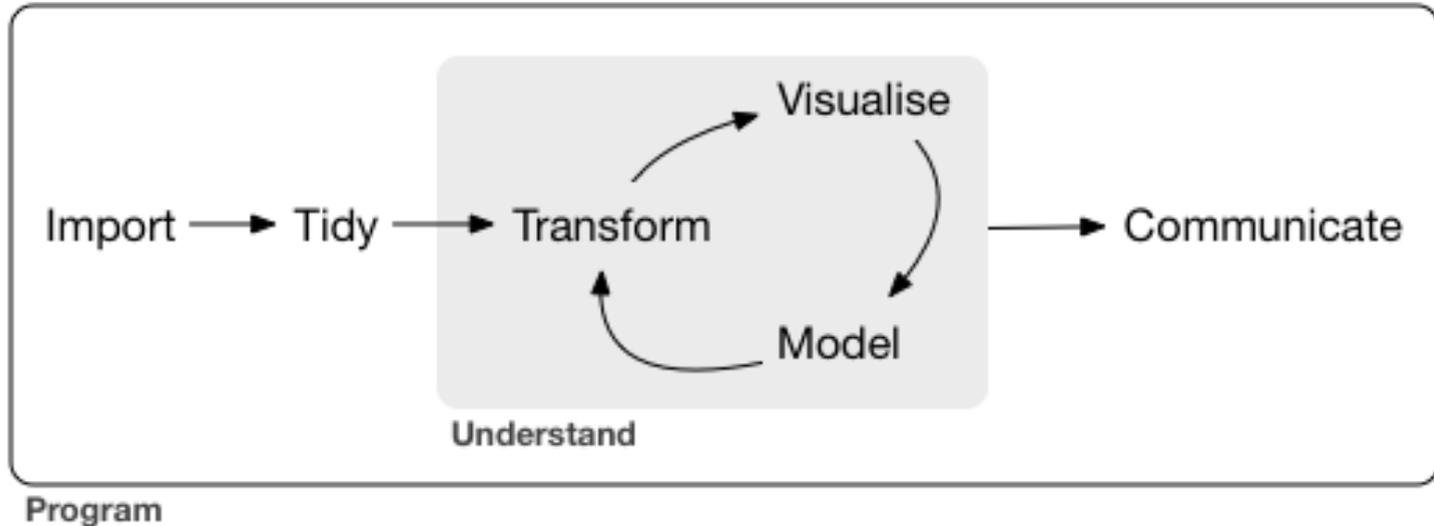
É uma das formas de comunicar os resultados!



Fonte: R for Data Science

# Por que apresentações são importantes?

É uma das formas de comunicar os resultados!



Fonte: R for Data Science

O que vocês utilizam em apresentações?

# Formas de criar apresentações no R

- Pacote **Xaringan** - Utiliza RMarkdown e a biblioteca remark.js do JavaScript para criação de apresentações.
- Outros:
  - output: **beamer\_presentation** -> beamer (pdf)
  - output: **ioslides\_presentation** -> ioslides (HTML)
  - output: **slidy\_presentation** -> slidy (HTML)

Obs.: para separar entre um slide e outro, usar três \*

## Foco de hoje

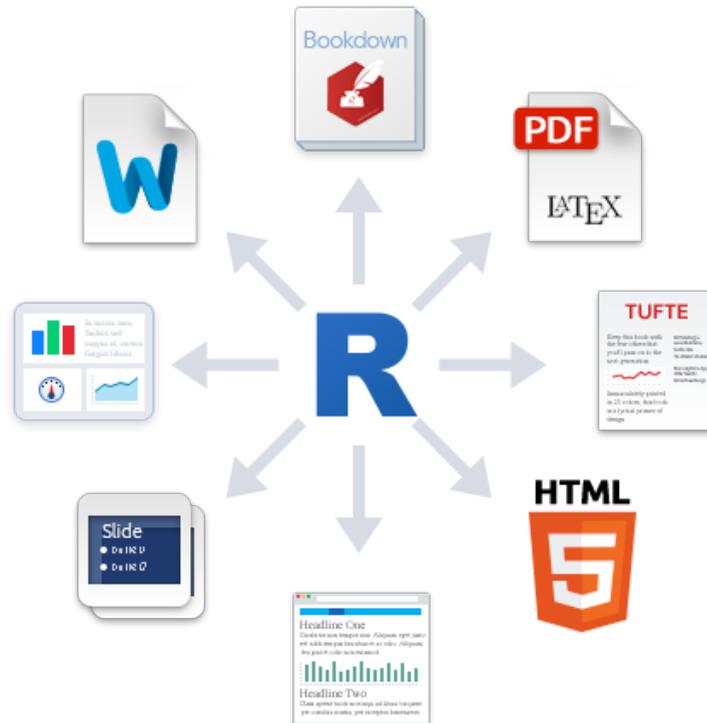
Praticar como criar uma apresentação utilizando o R e o pacote Xaringan

# RMarkdown

- Arquivos **.Rmd** são reprodutíveis.
- É um tipo de arquivo que suporta códigos em R, texto, markdown e outros formatos.
- Apresenta os seus códigos, seus resultados, e também seus comentários.
- São interessantes para trabalhar em equipe, para comunicar seus resultados, e para desenvolver seus códigos em R.
- Apresentação Yihui: [Advanced R Markdown](#)
- Dica de livro: [R Markdown: The Definitive Guide](#)

# RMarkdown

- Possibilita exportar diferentes tipos de arquivos.



# Markdown

- O RMarkdown utiliza a linguagem de marcação Markdown.
- Markdown é uma linguagem de marcação simples.
- Elaborada para ser fácil de ler, fácil de escrever.
- Dica rápida: [Rmarkdown Cheatsheet 2.0](#) e [Rmarkdown Cheatsheet](#).

# Markdown

**Negrito:** **`**bold**`**

*Itálico:* `_italics_`

## Adicionar títulos

`# Título 1`

`## Título 2`

`### Título 3`

# Título 1

## Título 2

### Título 3

# Fazendo uma lista

Você pode fazer uma lista escrevendo com hífens ou asteriscos, como a seguir:

\* **negrito com asteriscos duplos**

\* *itálico com underline*

\* fonte do tipo código com crase

ou desta forma:

- **negrito com asteriscos duplos**

- *itálico com underline*

- fonte do tipo código com crase

Cada um vai aparecer como:

- **negrito com asteriscos duplos**
- *itálico com underline*
- fonte do tipo código com crase

Referência: [Software Carpentry](#)

# Fazendo uma lista enumerada

Você pode fazer uma lista numerada usando somente números. Você pode usar o mesmo número quantas vezes quiser:

1. **negrito com asteriscos duplos**

1. *itálico com underline*

1. `fonte do tipo código com crase`

Isso irá aparecer como:

1. **negrito com asteriscos duplos**
2. *itálico com underline*
3. `fonte do tipo código com crase`

Referência: [Software Carpentry](#)

# Um pouco mais de Markdown

Você pode fazer um hiperlink dessa forma: `[**texto para mostrar**]`  
(<http://the-web-page.com>).

Você pode incluir arquivo de imagem dessa forma: `![**título**]`  
(<http://url/for/file>)

Se você sabe como escrever equações em LaTeX, ficará feliz em saber que pode usar `$$` para inserir equações matemáticas, como:

```
# $$y = \mu + \sum_{i=1}^p \beta_i x_i + \epsilon$$
```

$$y = \mu + \sum_{i=1}^p \beta_i x_i + \epsilon$$

Referência: [Software Carpentry](#)

# Mais sobre Markdown

Não esqueça que você pode consultar:

- [RMarkdown Cheatsheet](#)
- [R Markdown from RStudio](#)
- [R Markdown: The Definitive Guide](#)
- [R for Data Science - Cap 27: R Markdown](#)
- [Software Carpentry - Producing Reports With knitr](#)

# Como criar uma apresentação utilizando o pacote Xaringan

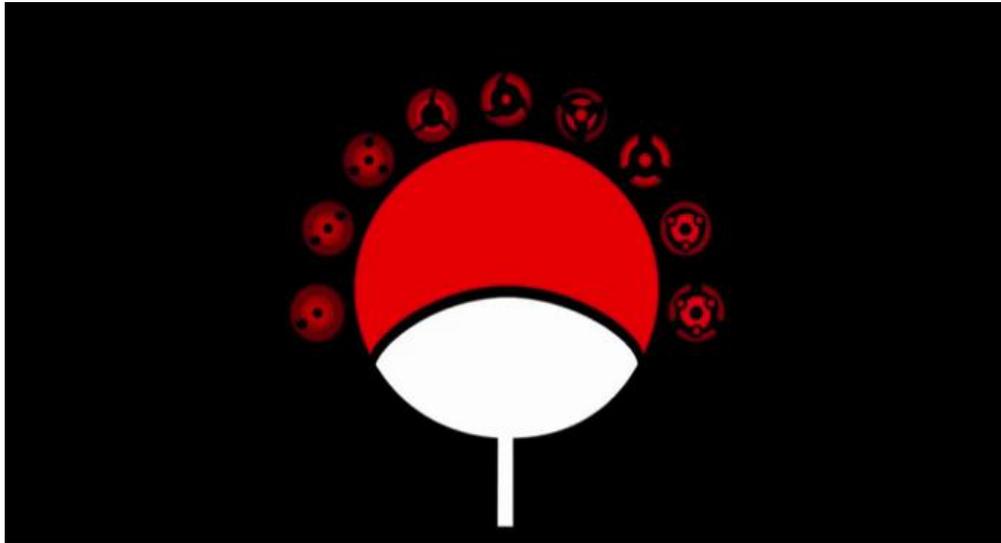
# Introdução sobre o Xaringan

- É um pacote do R para criar apresentações usando o [remark.js](#) através do RMarkdown.



- Mais sobre o xaringan:
  - [Apresentação da documentação](#)
  - [Livro R Markdown \(Capítulo 7\)](#)
  - [Remark.js Wiki](#)
  - [Xaringan Wiki](#)

# Curiosidade sobre o Xaringan



Como pronuncia Xaringan?

- Veja a [resposta do autor do pacote](#)
- [Video](#) (1 minuto e 04 segundos)

# Criador do Xaringan

- O pacote foi criado por **Yihui Xie**, que é desenvolvedor de software na RStudio.
- **Yihui Xie** também desenvolveu os seguintes pacotes: **knitr**, **animation**, **bookdown**, **blogdown**, **pagedown** e **tinytex**.



*Thanks Yihui for this amazing package!*

# Estrutura da apresentação

- No xaringan, utilizamos o arquivo `.Rmd` para escrever a apresentação, e é compilado para um arquivo `.html`
- Esse arquivo `.html` é interpretado pelo navegador (ex. Chrome), utilizando também os arquivos `.css` e `.js`



- **HTML** (HyperText Markup Language - Linguagem de Marcação de Hipertexto) - é uma linguagem de marcação, é usado para estruturar a página.
- **CSS** (Cascading Style Sheets - Folha de Estilo em Cascata) - é usado para estilizar os elementos escritos no HTML.
- **Javascript** - é uma linguagem de programação, roda no navegador e permite que as páginas sejam dinâmicas.
- Se quiser aprender mais sobre, existem muitos materiais gratuitos na internet. Ex: [FreeCodeCamp](#).

# Pré-requisitos

- R
- RStudio
- Pacote RMarkdown

*(O R instala esse pacote automaticamente quando necessário)*

- Pacote Xaringan
- Pacote knitr
- Google Chrome ou Chromium

# Instalar o Xaringan

- Existem duas formas de instalar:

Pelo CRAN:

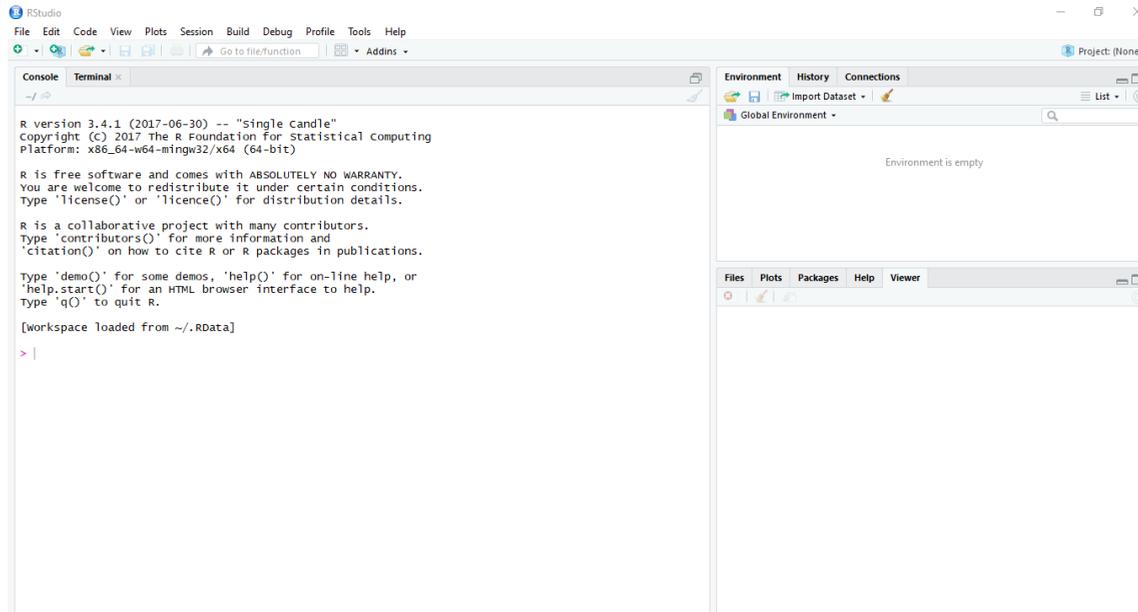
```
install.packages("xaringan")
```

Pelo Github:

```
devtools::install_github("yihui/xaringan")
```

# Primeiros Passos

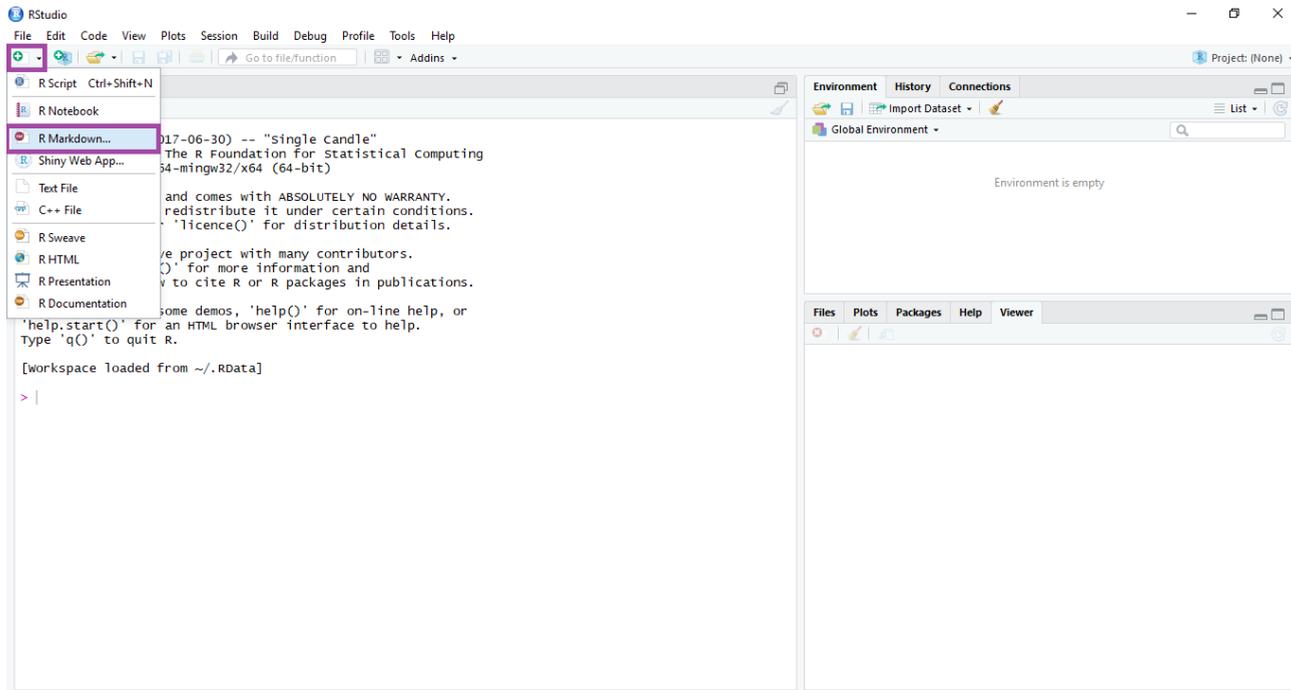
- Inicie o RStudio:



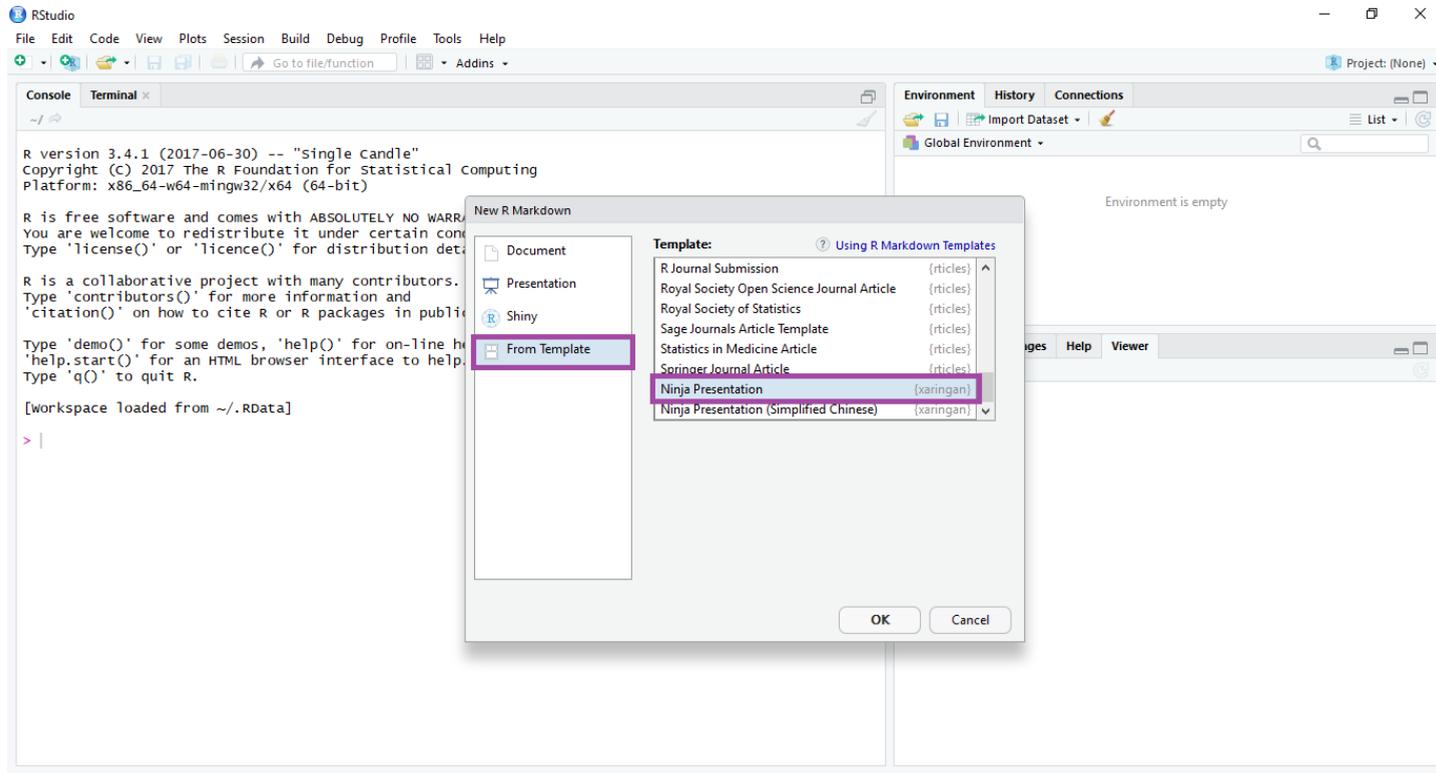
- Instale os pacotes:

```
install.packages("rmarkdown")
install.packages("knitr")
devtools::install_github("yihui/xaringan")
```

# Crie um novo arquivo RMarkdown



# Selecione o template Ninja Presentation

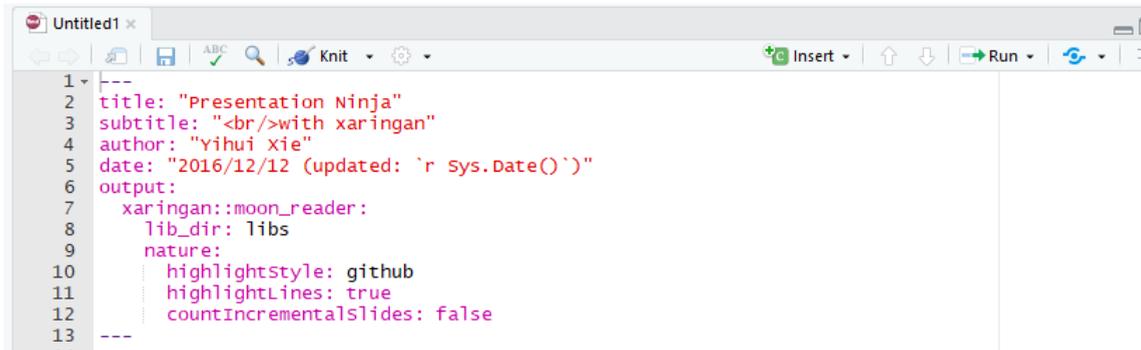


Dica: para facilitar posteriormente, crie a sua apresentação com o nome de `index.Rmd`



# yaml - metadados do documento

- yet another markup language
- Cuidado com a **identação**
- Controla muitas das configurações do documento



```
1 ---
2 title: "Presentation Ninja"
3 subtitle: "<br/>with xaringan"
4 author: "Yihui Xie"
5 date: "2016/12/12 (updated: `r Sys.Date()`)"
6 output:
7   xaringan::moon_reader:
8     lib_dir: libs
9     nature:
10      highlightstyle: github
11      highlightLines: true
12      countIncrementalSlides: false
13 ---
```

- title: título
- subtitle: subtítulo
- author: autor
- date: data
- output: formato (Ex: html\_document, pdf\_document, etc). No caso do Xaringan, o output é **xaringan::moon\_reader**
  - lib\_dir: pasta onde colocará bibliotecas necessárias (ex. remark-latest.min.js)

Leia também: [Output Formats - R Markdown: The Definitive Guide](#)

# Quais são os temas disponíveis?

- Depende da versão do Xaringan que está usando:

```
names(xaringan::list_css())
```

```
## [1] "chocolate-fonts" "chocolate" "default-fonts"
## [4] "default" "duke-blue" "fc-fonts"
## [7] "fc" "hygge-duke" "hygge"
## [10] "kunoichi" "lucy-fonts" "lucy"
## [13] "metropolis-fonts" "metropolis" "middlebury-fonts"
## [16] "middlebury" "ninjutsu" "rladies-fonts"
## [19] "rladies" "robot-fonts" "robot"
## [22] "rutgers-fonts" "rutgers" "shinobi"
## [25] "tamu-fonts" "tamu" "uo-fonts"
## [28] "uo" "uol-fonts" "uol"
```

Alguns exemplos de disponíveis: `xaringan themes`

# Colocar o tema RLadies

```
output:  
  xaringan::moon_reader:  
    css: ["rladies", "rladies-fonts"]
```

- Criado por/Made by: Alison Presmanes Hill



# Outro tema RLadies: Kunoichi

- É esse tema que estamos usando :)



```
output:  
  xaringan::moon_reader:  
    css: [kunoichi, ninjutsu, rladies-fonts]
```

- Criado por/Made by: [Emi Tanaka](#)

# CSS

- Utilizando o CSS, é possível adaptar os temas e também criá-los :)
- No yaml, no atributo: `output: xaringan::moon_reader: css:` (onde definimos o nome dos temas), aponte o arquivo `.css` que deseja utilizar (entre aspas).
- Para manter o diretório organizado, é aconselhável criar uma pasta chamada `css` e colocar os arquivos `.css` nessa pasta.
- Exemplo:

```
output:  
  xaringan::moon_reader:  
    css: ["css/rladies.css", "css/rladies-fonts.css"]
```

# CSS

- Para adaptar algum tema (alterar estilos específicos):

1 - Copie os arquivos originais do tema para a sua pasta `/css` (procure no seu diretório a pasta que contém os arquivos originais, geralmente estão em `libs/remark-css-0.0.1`);

2 - Especifique no `yaml` o nome e pasta dos arquivos novos. Exemplo:

```
output:  
  xaringan::moon_reader:  
    css: ["css/rladies.css", "css/rladies-fonts.css"]
```

3 - Altere o código `.css` dos novos arquivos para personalizar a apresentação.

Dica: No navegador, o atalho `ctrl + shift + c` pode ajudar a inspecionar os elementos.

# JavaScript - RemarkJS

- O Xaringan utiliza o seguinte arquivo javascript: <https://remarkjs.com/downloads/remark-latest.min.js>
- Por padrão, a apresentação só funcionará online!
- Como deixar o RemarkJS offline?

```
xaringan::summon_remark() # o xaringan irá fazer o download  
# da versão mais recente do remark e salvar na pasta /libs
```

O yaml deve conter as seguintes linhas:

```
output:  
  xaringan::moon_reader:  
    chakra: libs/remark-latest.min.js  
    lib_dir: libs
```

# Encoding

Para evitar problemas de codificação, adicione o encoding no yaml:

```
encoding: "UTF-8"
```

Leia o texto do blog do Yihui: [My Biggest Regret in the knitr Package UTF-8, and UTF-8 only, or we cannot be friends](#)

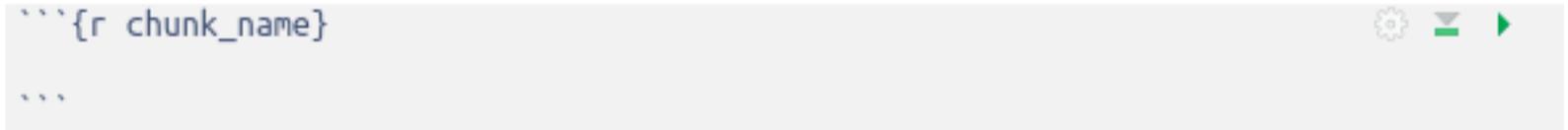
# Exemplo de yaml desta apresentação

```
---
title: "Comunicando seus resultados: "
subtitle: "Criando apresentações com R"
author: "Beatriz Milz e Haydee Svab - R-Ladies São Paulo"
date: "20 de maio de 2019 <br> (Atualizado em: `r format(Sys.Date(), for
encoding: "UTF-8"
output:
  xaringan::moon_reader:
    chakra: libs/remark-latest.min.js
    css: ["css/kunoichi.css", "css/ninjutsu.css",
          "css/rladies-fonts.css"]
    lib_dir: libs
    nature:
      highlightStyle: github
      highlightLines: true
      countIncrementalSlides: false
---
```

# Chunks de Código R

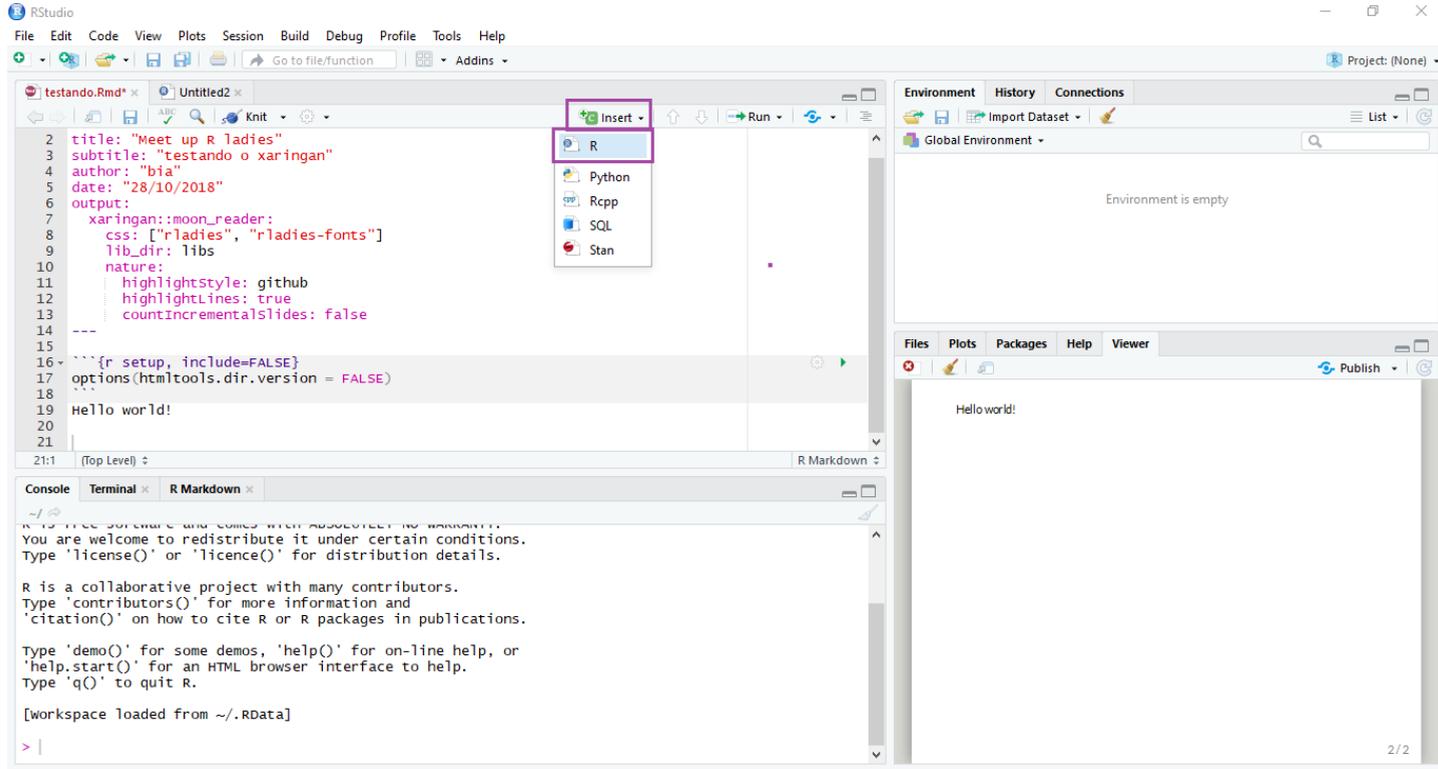
- É um campo onde podemos inserir código de R em um arquivo RMarkdown:

```
```${r chunk_name}
...
````
```



- É uma boa ideia dar a cada chunk um nome, pois eles podem ajudar a corrigir erros, e, se quaisquer gráficos são produzidos, os nomes dos arquivos serão baseados no nome do chunk de código que os erros produzirem.

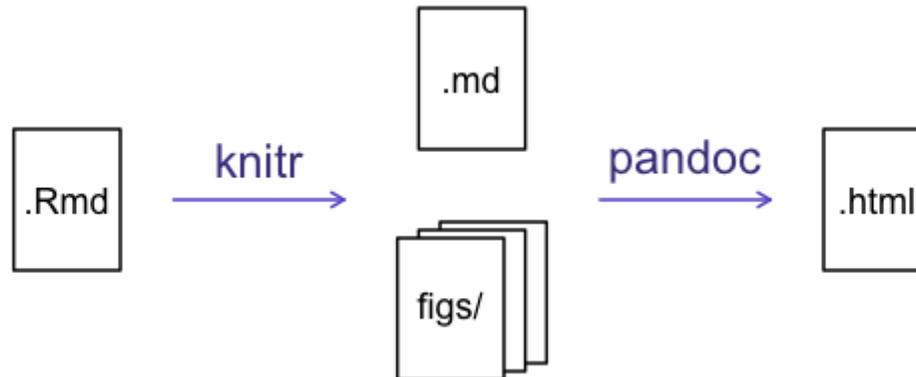
# Como criar um chunk de código?



Dica: Atalho do teclado para criar chunks: **Ctrl + Alt + I**.

# Como as coisas são compiladas?

- Ao pressionar o botão “Knit HTML”, o documento RMarkdown é processado pelo knitr e um documento Markdown é produzido (como, também, um potencial conjunto de arquivos de figuras): o código em R é executado e se figuras são produzidas, links para essas figuras serão incluídos.
- A seguir, os documentos Markdown e de figuras são processados pela ferramenta pandoc, que converte o arquivo .md em um arquivo .html, com as figuras anexadas.



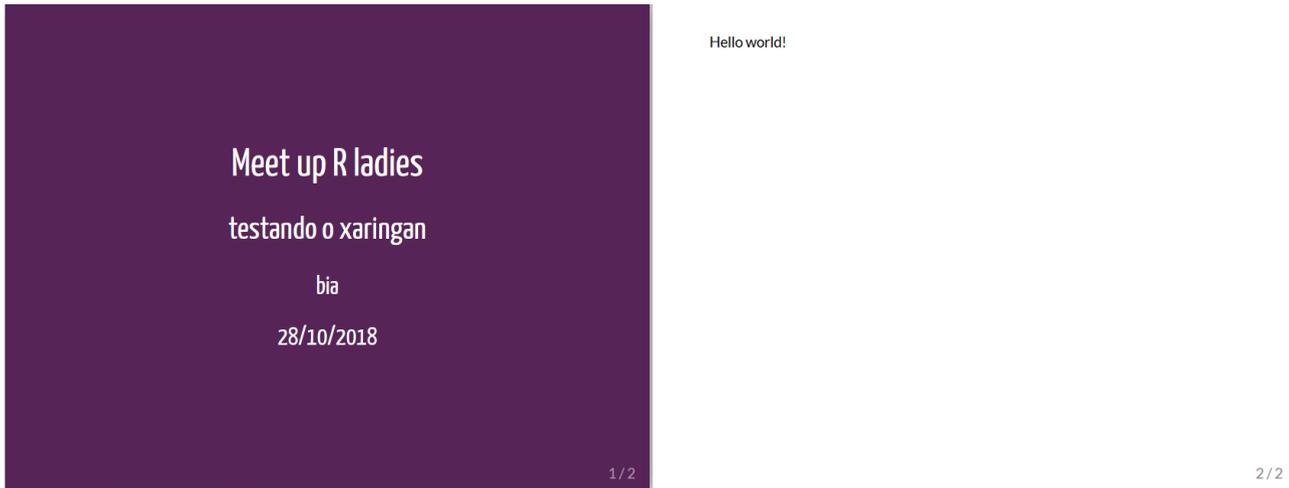
# Exemplo do yaml



```
1 ---
2 title: "Meet up R ladies"
3 subtitle: "testando o xaringan"
4 author: "bia"
5 date: "28/10/2018"
6 output:
7   xaringan::moon_reader:
8     css: ["rladies", "rladies-fonts"]
9     lib_dir: libs
10    nature:
11      highlightstyle: github
12      highlightLines: true
13      countIncrementalSlides: false
14 ---
15
16 {r setup, include=FALSE}
17 options(htmltools.dir.version = FALSE)
18
19 Hello world!
```

- Aperte o Knit e...

# Resultado:



- O knit exportará um arquivo do tipo html
- Agora o tema da apresentação está configurado!
- Aperte H ou ? para conferir a lista de atalhos do Xaringan

# DEU ERRO NO KNITR!

Calma!

É bem comum dar erro ao compilar o arquivo pois **qualquer** código errado causa isso. Algumas coisas pra se observar:

- **Yaml header** - se algo estiver errado, vai dar erro ao compilar. Muito cuidado com a indentação!
- **Código nos chunks** - teste o *Run All Chunks* e veja se dá erro em algum específico. Se descobrir, tente verificar o que há de errado no código (pode ser a falta de um parênteses, vírgula, entre outros).



# Opções de chunk

Há uma variedade de opções que podem afetar como os chunks de código são tratados.

- `echo=FALSE` -> evitar que o próprio código apareça
- `results="hide"` -> evitar que qualquer resultado seja impresso
- `eval=FALSE` -> mostrar o código, mas ele não é avaliado
- `warning=FALSE` e `message=FALSE` -> ocultar qualquer mensagens de avisos produzidas
- `out.width` -> controlar o tamanho de figuras produzidas (Ex: `out.width = "100%"`)

Assim, você pode escrever:

```
```${r load_libraries, echo=FALSE, message=FALSE}
library("dplyr")
library("ggplot2")
```
```

Referência: [Software Carpentry](#)

# Opções globais de chunk

Muitas vezes, haverá algumas opções que você irá querer repetir seu uso.

Para isso, você pode configurar as opções globais de chunk, como:

```
``{r global_options, echo=FALSE}  
knitr::opts_chunk$set(fig.path="Figs/", message=FALSE, warning=FALSE,  
                        echo=FALSE, results="hide", fig.width=11)  
``
```

A opção `fig.path` define onde as figuras serão salvas.

A `/` é importante, pois sem ela, as figuras seriam salvas no local padrão, com nomes começando com `Figs`.

Referência: [Software Carpentry](#)

# Código R no texto

Você pode fazer com que todo número em seu relatório seja reproduzível.

```
# Use `r` e ` ` para um in-line code chunk .  
# O código será executado e  
# substituído com o valor do resultado.
```

```
`r code`
```

Ex:

```
`r round(3.14159265359, 2)`
```

3.14

Não deixe que esse chunk *in-line* seja separado através das linhas.

# Código em R no texto

Outro exemplo:

```
# O dataframe mtcars possui `r nrow(mtcars)` carros.
```

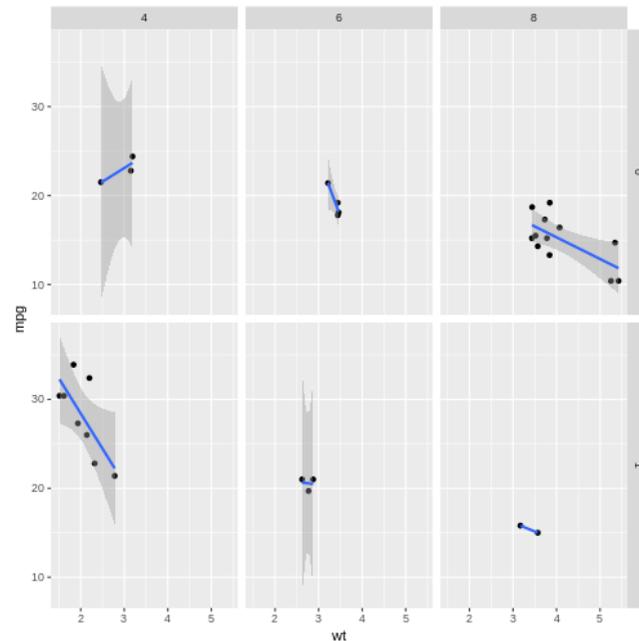
Exemplo: O dataframe mtcars possui 32 carros.

Referência: [Software Carpentry](#)

# Exemplo: adicionando um gráfico

- Crie um chunk e escreva o código que você utiliza para criar um gráfico :)

```
library(tidyverse)
ggplot(data = mtcars, mapping = aes(x = wt, y = mpg)) + geom_point() +
  stat_smooth(method = 'lm') + facet_grid(am ~ cyl)
```



Fonte do exemplo: [http://stcorp.nl/R\\_course/tutorial\\_ggplot2.html](http://stcorp.nl/R_course/tutorial_ggplot2.html)

# Exemplo: adicionando uma tabela

```
DT::datatable(rladies_br)
```

Show  entries

Search:

|    | city           | dt_created | members |
|----|----------------|------------|---------|
| 1  | Florianópolis  | 2019-04-07 | 22      |
| 2  | Salvador       | 2018-07-23 | 114     |
| 3  | Niterói        | 2018-06-04 | 289     |
| 4  | Curitiba       | 2018-05-29 | 151     |
| 5  | Belo Horizonte | 2018-04-20 | 509     |
| 6  | São Paulo      | 2018-02-10 | 343     |
| 7  | Porto Alegre   | 2017-10-30 | 218     |
| 8  | Natal          | 2017-09-25 | 135     |
| 9  | Americana      | 2017-06-24 | 28      |
| 10 | Rio de Janeiro | 2017-02-27 | 394     |

# Adicionando imagens usando o Knitr

Função: `knitr::include_graphics()`

Exemplo:

```
knitr::include_graphics("https://www.r-project.org/Rlogo.png")
```



# Adicionando imagens usando Knitr: opções de Chunk

## Tamanho da imagem

- `out.width = "50%"`



- Opções do KnitR

# Adicionando imagens usando Knitr: opções de Chunk

## Alinhamento da figura

- `fig.align = 'center', 'default', **'left**, 'right'`



- `fig.align = 'center', 'default', 'left', **'right**`



# Adicionando imagens usando Knitr: opções de Chunk

## Legenda de figura

- `fig.cap="Logo R"`



R Logo

# Preview da apresentação

```
xaringan::inf_mr("index.Rmd")
```

- Pode não funcionar com o tema Kunoichi :(
- Post do Yihui sobre isso: <https://yihui.name/en/2019/02/ultimate-inf-mr/>

# Salvar a versão em PDF

- O autor do pacote disse que a forma mais confiável é abrir no Google Chrome ou Chromium, e salvar como PDF - não dá certo com o Mozilla Firefox. Então, usando o navegador Chrome ou no Chromium, imprima para arquivo (Ctrl + P).
- Existe a função **chrome\_print** do pacote `pagedown`:
  - É recente e ainda está em desenvolvimento, portanto, pode apresentar instabilidade
  - Precisa ter o Chrome instalado

```
remotes::install_github('rstudio/pagedown')
```

```
pagedown::chrome_print("index.html")
```

Referência: Apresentação do Yihui na RStudio Conf 2019: `pagedown` - Creating beautiful PDFs with R Markdown and CSS

# Hospedagem da apresentação utilizando o Github Pages

# Compartilhando sua apresentação

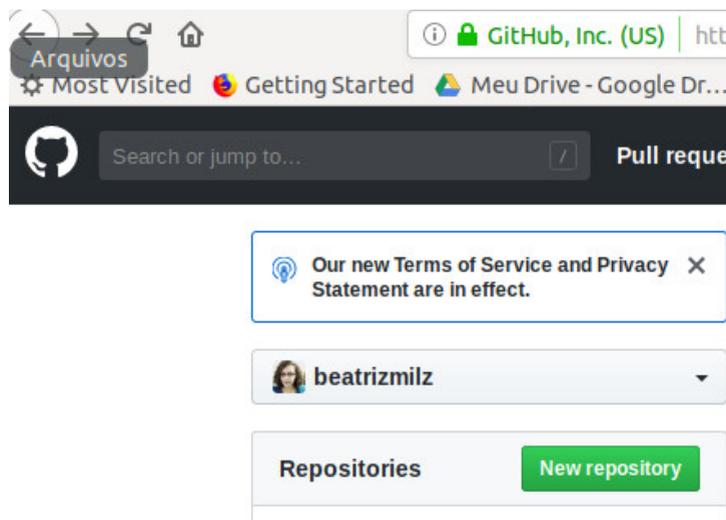
- Já vimos que é possível exportar um PDF



- O formato que utilizamos para exportar uma apresentação elaborada com o Xaringan é **.html** -> é necessário apertar o botão Knit (mostrado anteriormente).
- O knit gera um **arquivo .html** -> para que ele funcione, depende de outros arquivos locais (salvos na pasta do projeto).
- É possível hospedar o seu arquivo de apresentação em sites como o [Github Pages](#) e [Netlify](#).

# Passo a passo: Como hospedar sua apresentação utilizando o Github Pages?

- 1) Crie uma conta gratuita no github: <https://github.com/>
- 2) Na sua conta do Github, crie um novo repositório (New repository)



- Dica: Esse post do @w\_oliveiras é sobre como usar git e github utilizando o terminal/linha de comando e pode ajudar!

# Compartilhando a apresentação via Github Pages

3) Coloque um nome e descrição no novo repositório e deixe-o público

## Create a new repository

A repository contains all the files for your project, including the revision history.

Owner



Repository name



Great repository names are short and memorable. Need inspiration? How about [probable-telegram](#).

Description (optional)

teste

 **Public**  
Anyone can see this repository. You choose who can commit.

 **Private**  
You choose who can see and commit to this repository.

**Initialize this repository with a README**  
This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: **None** ▾

Add a license: **None** ▾ 

Create repository

# Compartilhando a apresentação via Github Pages

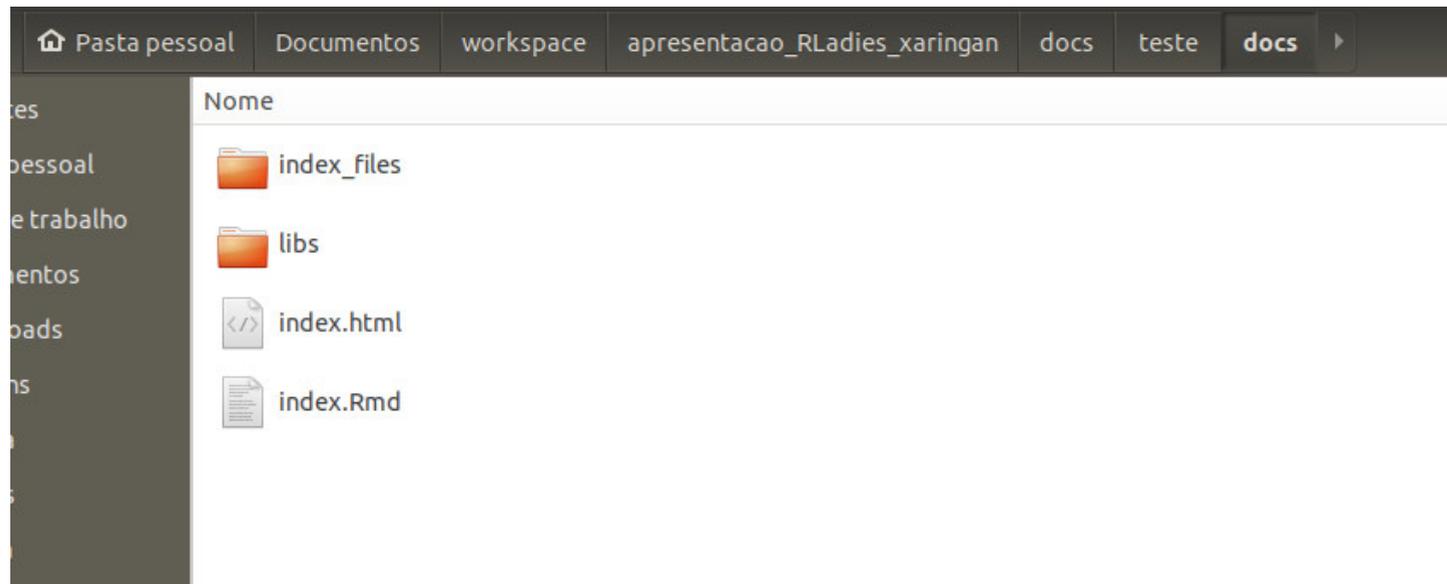
## 4) Repositório criado!

The screenshot shows a GitHub repository page for 'beatrizmilz / teste'. At the top, there are navigation links for 'Code', 'Issues 0', 'Pull requests 0', 'Projects 0', 'Wiki', 'Insights', and 'Settings'. On the right, there are buttons for 'Watch 0', 'Star 0', and 'Fork 0'. Below the repository name, there is an 'Edit' button and a 'Manage topics' link. A summary bar shows '1 commit', '1 branch', '0 releases', and '1 contributor'. Below this, there are buttons for 'Branch: master', 'New pull request', 'Create new file', 'Upload files', 'Find file', and 'Clone or download'. The commit history shows an 'Initial commit' by 'beatrizmilz' 3 minutes ago. The file list shows 'README.md' with an 'Initial commit' 3 minutes ago. The content of 'README.md' is displayed, showing the word 'teste' in a large font, followed by a horizontal line and the word 'teste' in a smaller font.

# Compartilhando a apresentação via Github Pages

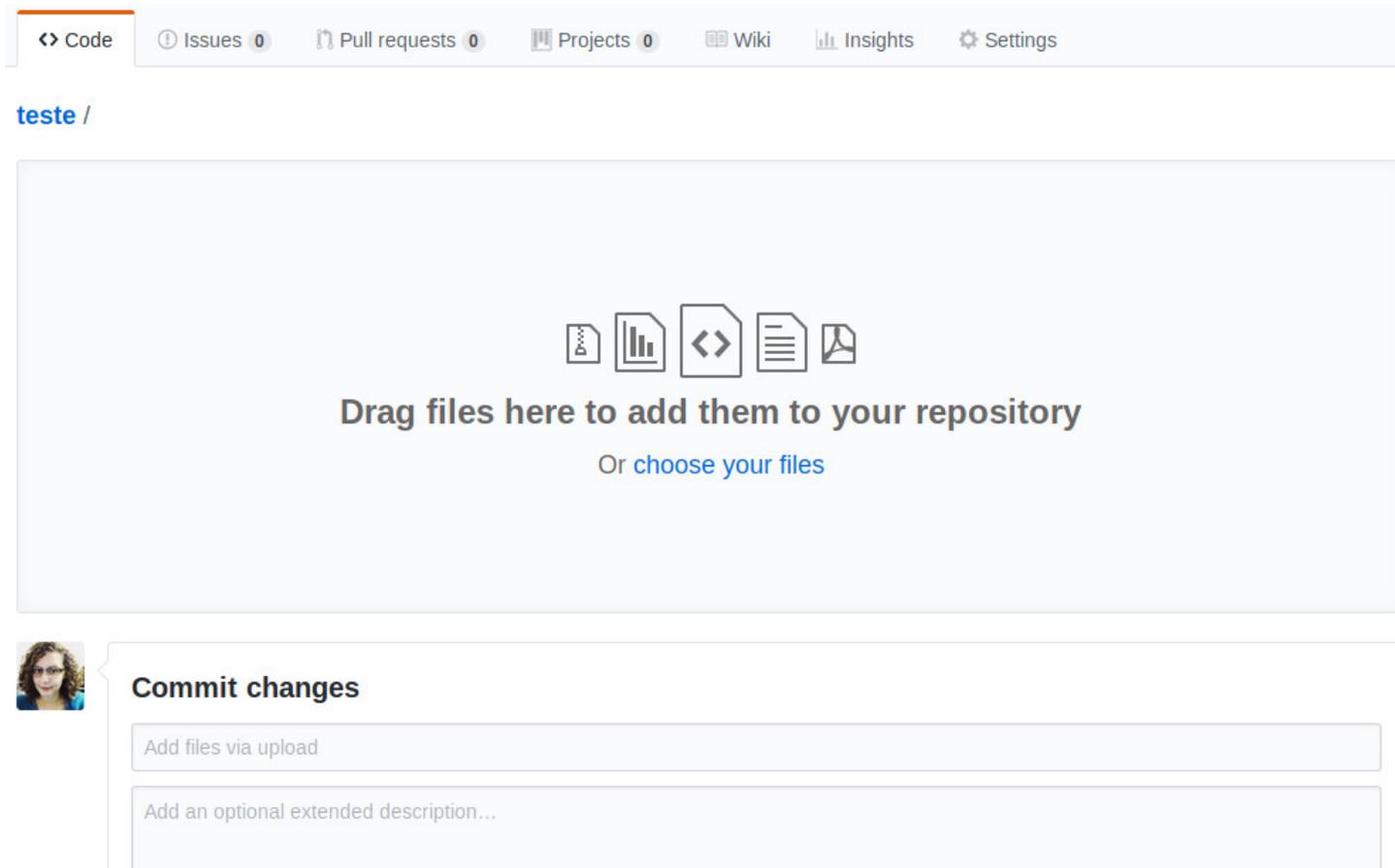
## 5) Githubpages e pasta DOCS

- Para o Github Pages funcionar, é preciso adicionar o conteúdo em uma pasta chamada `docs` -> antes de fazer upload no seu computador/workingdirectory/projeto crie uma pasta com o nome `docs` e coloque todos os arquivos da sua apresentação nessa pasta
- De preferência, a sua apresentação deve se chamar `index.html` (após compilado)



# Compartilhando a apresentação via Github Pages

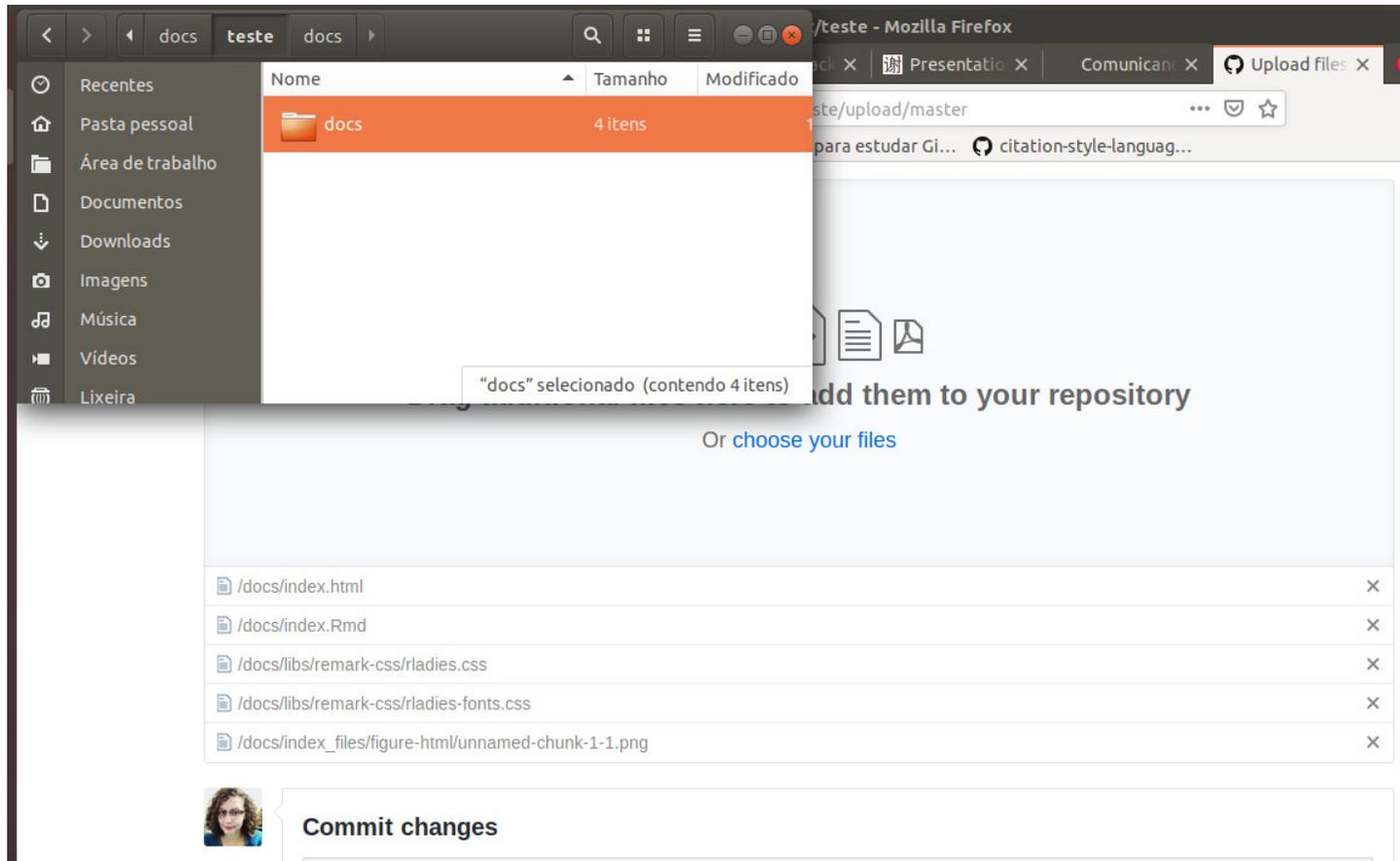
- Área de upload do seu repositório do Github (essa não é a única forma de realizar upload, também é possível através do Github desktop e pelo terminal de comando)



The screenshot displays the GitHub repository interface. At the top, there is a navigation bar with links for Code, Issues (0), Pull requests (0), Projects (0), Wiki, Insights, and Settings. Below the navigation bar, the repository name 'teste /' is visible. The main area is a large light blue box with the text 'Drag files here to add them to your repository' and 'Or choose your files'. Below this, there is a 'Commit changes' section with a user profile picture, a text input field for 'Add files via upload', and another text input field for 'Add an optional extended description...'. The 'Add files via upload' field is currently empty.

# Compartilhando a apresentação via Github Pages

6) Faça o upload da pasta docs INTEIRA no seu repositório do Github



The image shows a file explorer window on the left and a web browser window on the right. The file explorer is displaying the contents of a folder named 'docs', which contains 4 items. The web browser is showing the GitHub 'Upload files' interface, where the selected files are listed and ready to be committed to the repository.

File Explorer (Left):

- Nome
- Tamanho
- Modificado
- docs (4 itens)

Web Browser (Right):

- Upload files
- ste/upload/master
- para estudar Gi... citation-style-languag...
- add them to your repository
- Or [choose your files](#)
- Selected files:
  - /docs/index.html
  - /docs/index.Rmd
  - /docs/libs/remark-css/rladies.css
  - /docs/libs/remark-css/rladies-fonts.css
  - /docs/index\_files/figure-html/unnamed-chunk-1-1.png
- Commit changes

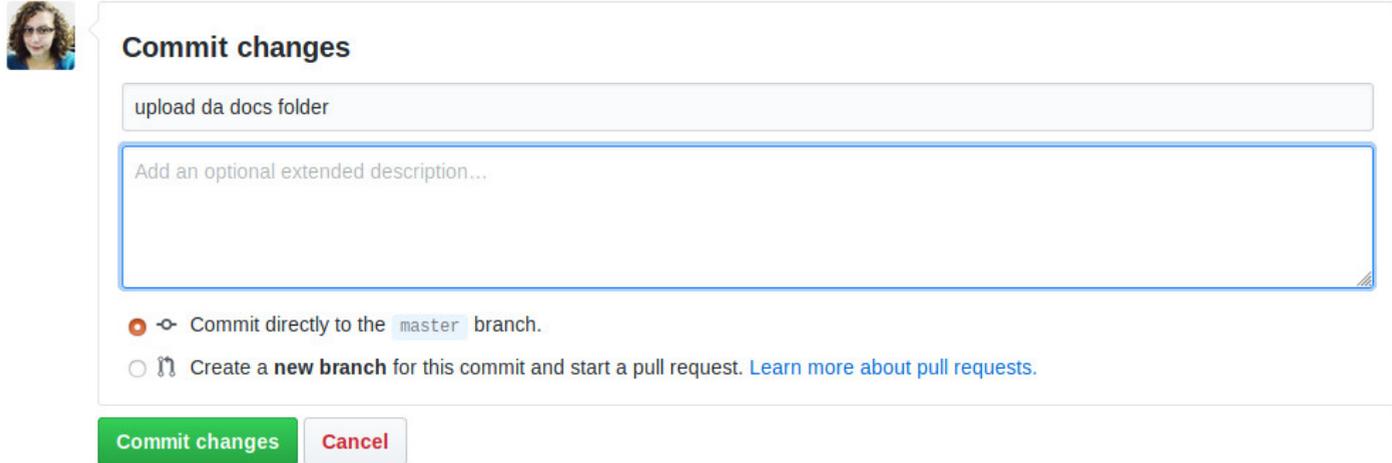
# Compartilhando a apresentação via Github Pages

7) Coloque uma mensagem no *commit*, para deixar registrado que alteração foi feita



File explorer showing the following files:

- /docs/index.Rmd
- /docs/libs/remark-css/rladies.css
- /docs/libs/remark-css/rladies-fonts.css
- /docs/index\_files/figure-html/unnamed-chunk-1-1.png



**Commit changes**

upload da docs folder

Add an optional extended description...

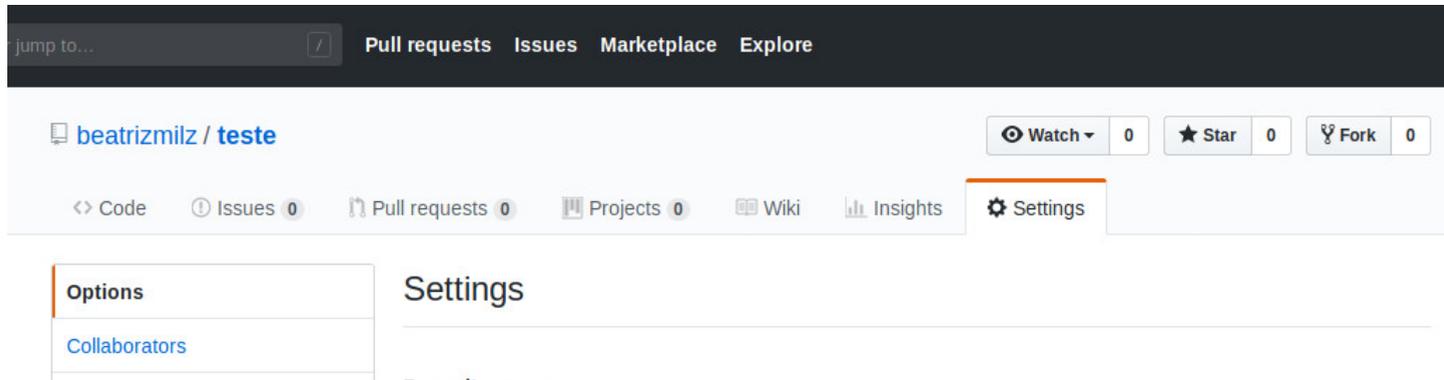
Commit directly to the `master` branch.

Create a **new branch** for this commit and start a pull request. [Learn more about pull requests.](#)

**Commit changes** **Cancel**

# Compartilhando a apresentação via Github Pages

8) No seu repositório, clique em Settings...



# Compartilhando a apresentação via Github Pages

8) ... e na seção Github Pages, selecione **master branch /docs folder**:

## GitHub Pages

GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.

### Source

GitHub Pages is currently disabled. Select a source below to enable GitHub Pages for this repository. [Learn more.](#)

master branch /docs folder ▾

Save

### Theme Chooser

Select a theme to publish your site with a Jekyll theme using the master branch /docs folder. [Learn more.](#)

Choose a theme

**Resultado:** <https://beatrizmilz.github.io/teste/>

# Agora é só enviar o link da sua apresentação!

Última dica: Você pode criar URL mais curtas com o [Bitly](#)



# Obrigada!

Apresentação elaborada com [Xaringan](#) e [Kunoichi](#)

Agradecimentos especiais:

- [Yihui Xie](#) - Pelo pacotes, livros, blogposts, vídeos
- [Bruna Wundervald](#) - Por ajudar na apresentação
- [Julio Trecenti](#), [Curso-R](#) e [13 aMostra de Estatística - IME USP](#) pelo mini-curso de RMarkdown