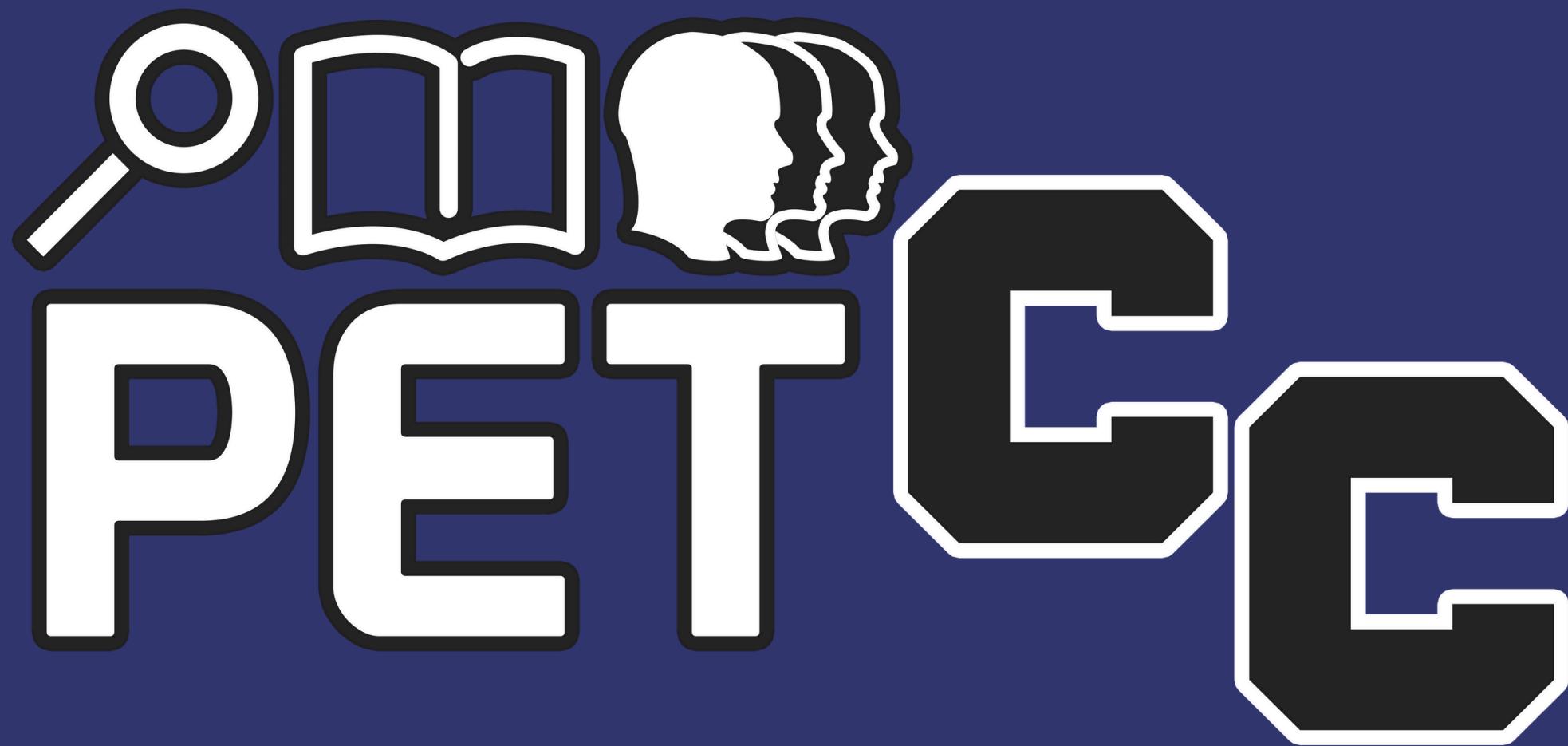


MINICURSO
BÁSICO DE

PETCC

R





Introdução à Linguagem R



Manipulação de dados



Análise e Visualização de Dados



Atividade Final

Código de acesso à turma do Classroom:



Repositório do github do minicurso:

<https://github.com/GiBorelli/minicursoR>



minicursoR

Public

Pin

Watch 0

Fork 0

Star 0

main ▾

1 Branch 0 Tags

Go to file

t

Add file ▾

Code ▾

About



GiBorelli organizacao pastas

39ceb11 · 3 minutes ago 6 Commits

Aulas e Ementa

organizacao pastas

3 minutes ago

datasets

organizacao pastas

3 minutes ago

README.md

Update README.md

14 minutes ago

No description, website, or topics provided.

Readme

Activity

0 stars

0 watching

0 forks

Releases

No releases published

[Create a new release](#)

Packages

No packages published

[Publish your first package](#)

Minicurso Básico de R

PET-CC - UFSM

Repositório de suporte para o Minicurso Básico de R feito pelo PET Ciência da Computação (PET-CC) da Universidade Federal de Santa Maria (UFSM) entre 20 e 23 de outubro de 2025.

Aula 01

Apresentação da linguagem



O QUE É O R?

COMO SURGIU?

ONDE É USADO?

POR QUÊ USAR R?

O que é o R?

- R é uma linguagem de programação focada em programação estatística e visualização de dados.
- É uma linguagem grátis e de código aberto. Grande parte das ferramentas utilizadas são bibliotecas feitas por usuários.
- É uma linguagem interpretada, não compilada (mostraremos o que isso significa).
- Uma das ferramentas mais utilizadas para programar em R é o RStudio.

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function Addins

Project: (None)

R Untitled1 x

Source on Save | Run | Source | Grid | Global Environment

1

1:1 (Top Level) R Script

Console Terminal Background Jobs

R 4.3.3 · ~/

```
R version 4.3.3 (2024-02-29 ucrt) -- "Angel Food Cake"
Copyright (C) 2024 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R é um software livre e vem sem GARANTIA ALGUMA.
Você pode redistribuí-lo sob certas circunstâncias.
Digite 'license()' ou 'licence()' para detalhes de distribuição.

R é um projeto colaborativo com muitos contribuidores.
Digite 'contributors()' para obter mais informações e
'citation()' para saber como citar o R ou pacotes do R em publicações.

Digite 'demo()' para demonstrações, 'help()' para o sistema on-line de ajuda,
ou 'help.start()' para abrir o sistema de ajuda em HTML no seu navegador.
Digite 'q()' para sair do R.
```

> |

Environment History Connections Tutorial

Import Dataset 95 MiB Grid

R Global Environment

Name	Type	Length	Size	Value
Environment is empty				

Files Plots Packages Help Viewer Presentation

Zoom Export

Como surgiu?

Inspirada pela linguagem S, a linguagem R foi criada em 1993 para ensinar estatística introdutória na Universidade de Auckland. Apenas no ano de 2000 a versão 1.0 oficial foi lançada.

Criadores

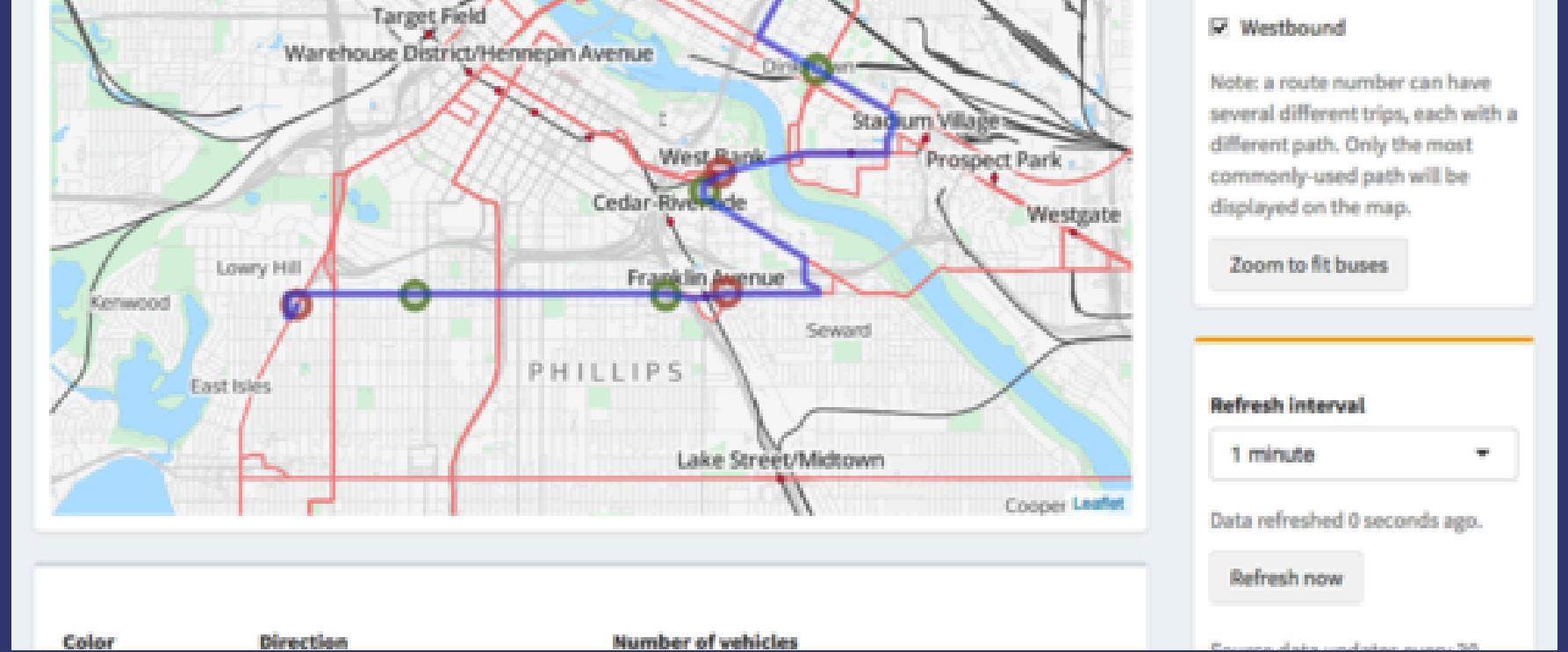
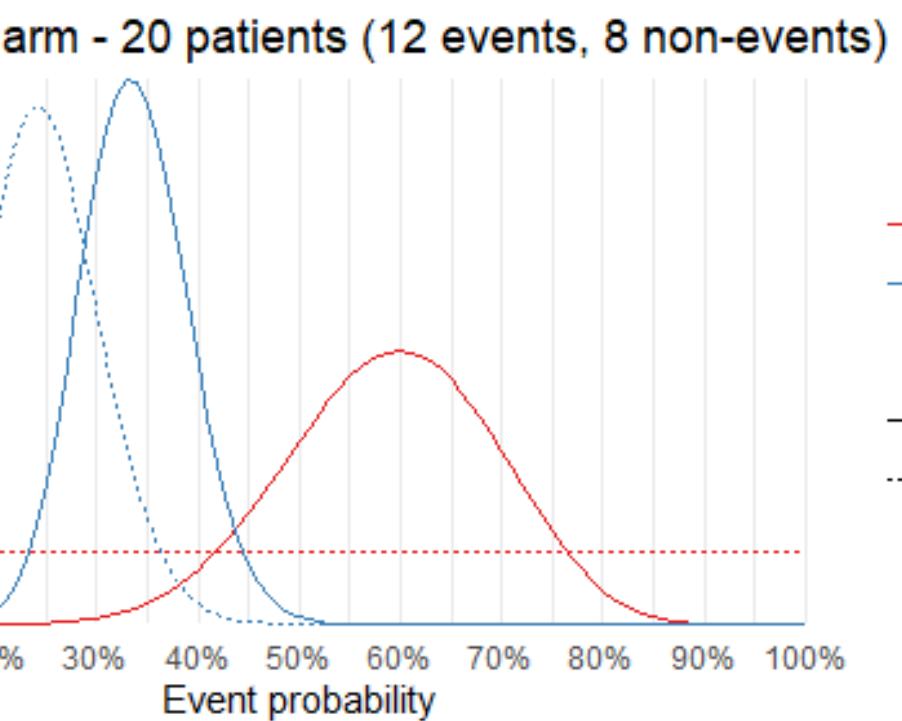
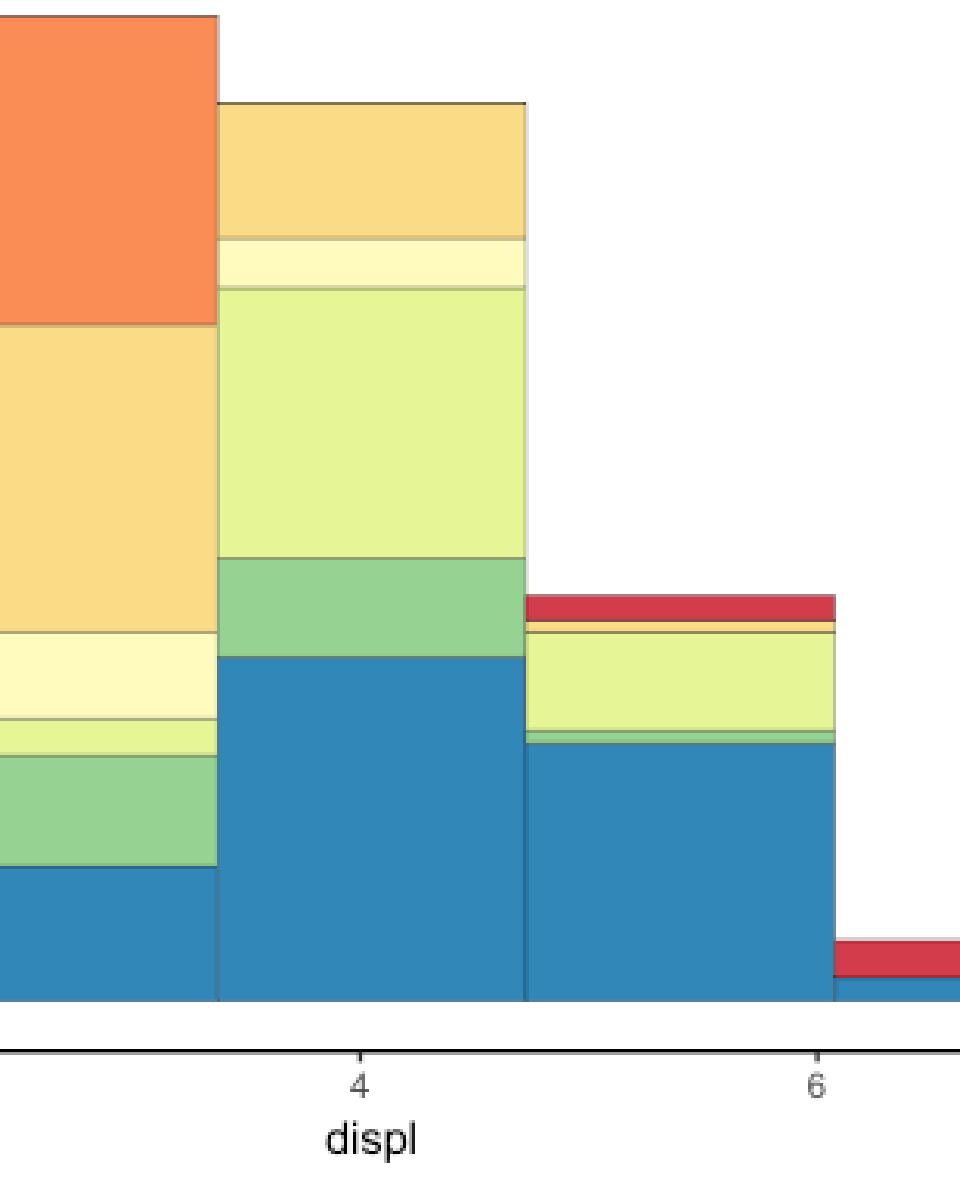


Ross
Ihaka

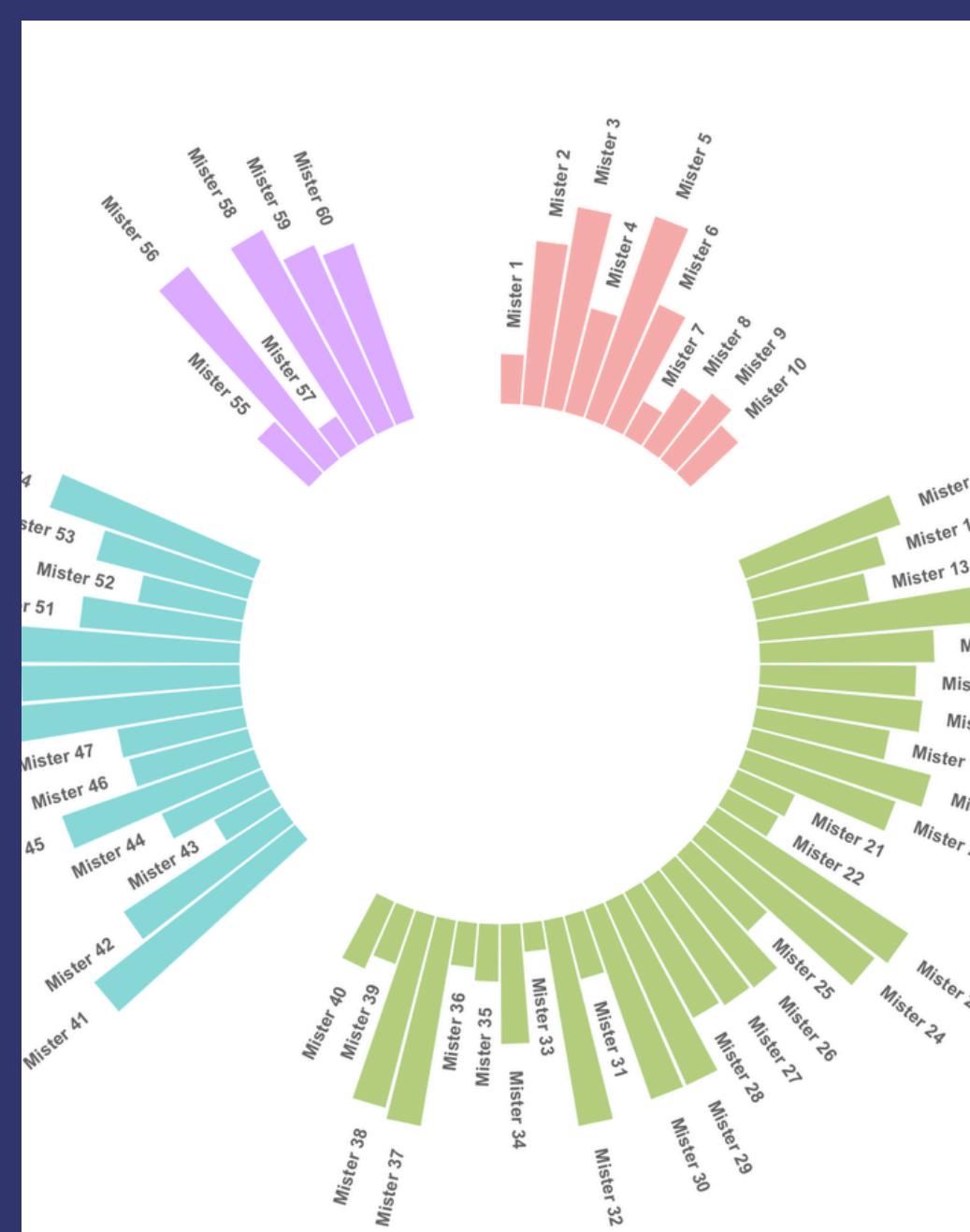
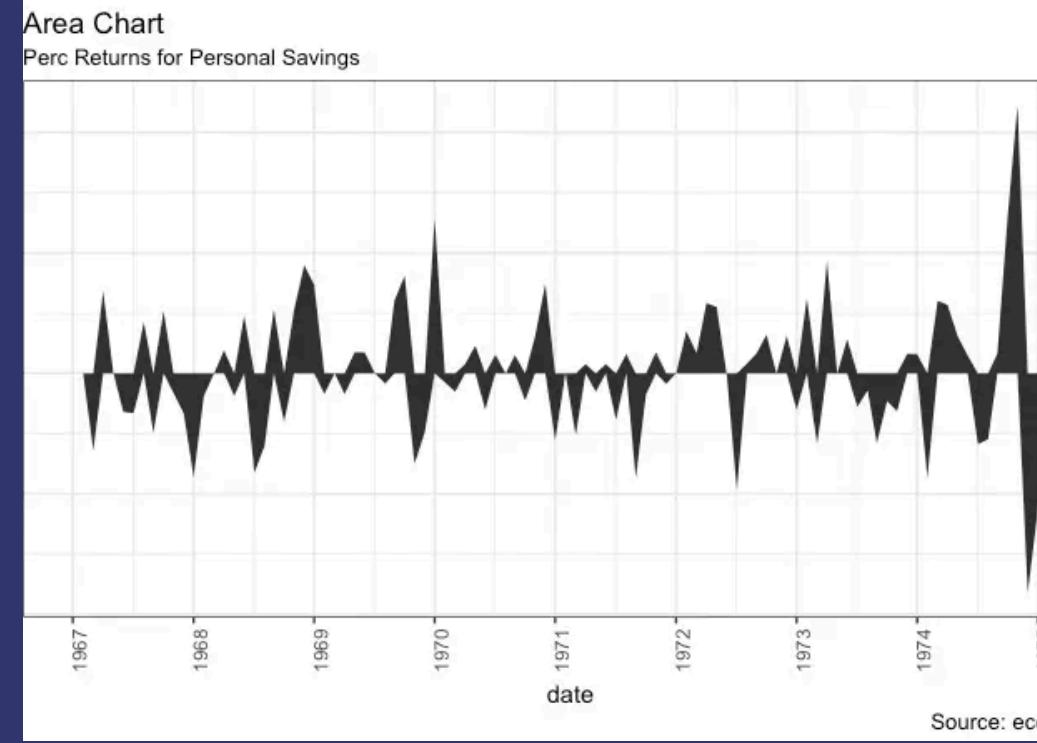
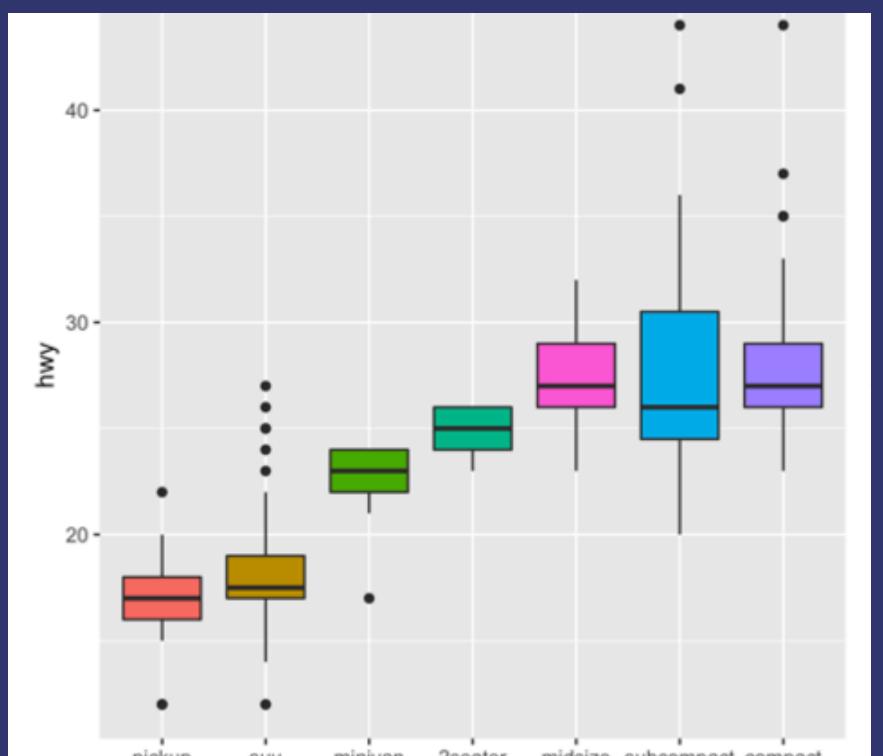
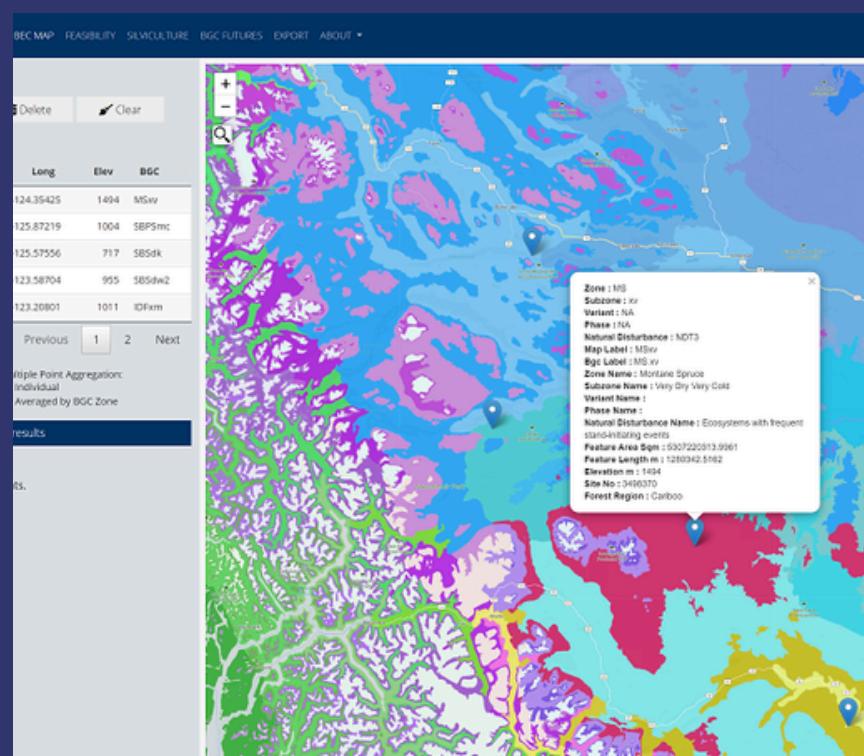
Robert
Gentleman

d Bins

ss Vehicle Classes



Onde o R é utilizado?



Por que usar R?

Linguagem
amigável mesmo
para não
programadores

Grande
comunidade,
várias
ferramentas

Integração com
outras
linguagens e com
WEB

INSTALANDO A FERRAMENTA

<https://posit.co/downloads/>



File Edit Code View Plots Session Build Debug Profile Tools Help

+ R Untitled1 x Go to file/function Addins

1

Prática no RStudio

1 - Abra o RStudio

1:1 (Top Level) R Script

Console Terminal Background Jobs

R 4.3.3 · ~/

```
R version 4.3.3 (2024-02-29 ucrt) -- "Angel Food Cake"
Copyright (C) 2024 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R é um software livre e vem sem GARANTIA ALGUMA.
Você pode redistribuí-lo sob certas circunstâncias.
Digite 'license()' ou 'licence()' para detalhes de distribuição.

R é um projeto colaborativo com muitos contribuidores.
Digite 'contributors()' para obter mais informações e
'citation()' para saber como citar o R ou pacotes do R em publicações.

Digite 'demo()' para demonstrações, 'help()' para o sistema on-line de ajuda,
ou 'help.start()' para abrir o sistema de ajuda em HTML no seu navegador.
Digite 'q()' para sair do R.
```

Environment History Connections Tutorial

Import Dataset 95 MiB Grid

R Global Environment

Name	Type	Length	Size	Value
Environment is empty				

Files Plots Packages Help Viewer Presentation

Zoom Export

Untitled1 x

Source on Save | Run | Source | Grid | Environment | History | Connections | Tutorial | Import Dataset | 95 MiB | Global Environment | Name | Type | Length | Size | Value | Environment is empty |

1

Ambiente de escrita do código (salvável)

1:1 (Top Level) R Script

Console Terminal Background Jobs

R 4.3.3 · ~/

```
R version 4.3.3 (2024-02-29 ucrt) -- "Angel Food Cake"
Copyright (C) 2024 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R é um software livre e vem sem GARANTIA ALGUMA.
Você pode redistribuí-lo sob certas circunstâncias.
Digite 'license()' ou 'licence()' para detalhes de distribuição.

R é um projeto colaborativo com muitos contribuidores.
Digite 'contributors()' para obter mais informações e
'citation()' para saber como citar o R ou pacotes do R em publicações.

Digite 'demo()' para demonstrações, 'help()' para o sistema on-line de ajuda,
ou 'help.start()' para abrir o sistema de ajuda em HTML no seu navegador.
Digite 'q()' para sair do R.
```

Terminal de execução

Variáveis carregadas no sistema

Files Plots Packages Help Viewer Presentation

Zoom Export

Environment is empty

Gráfico, ajuda, arquivos

Operadores

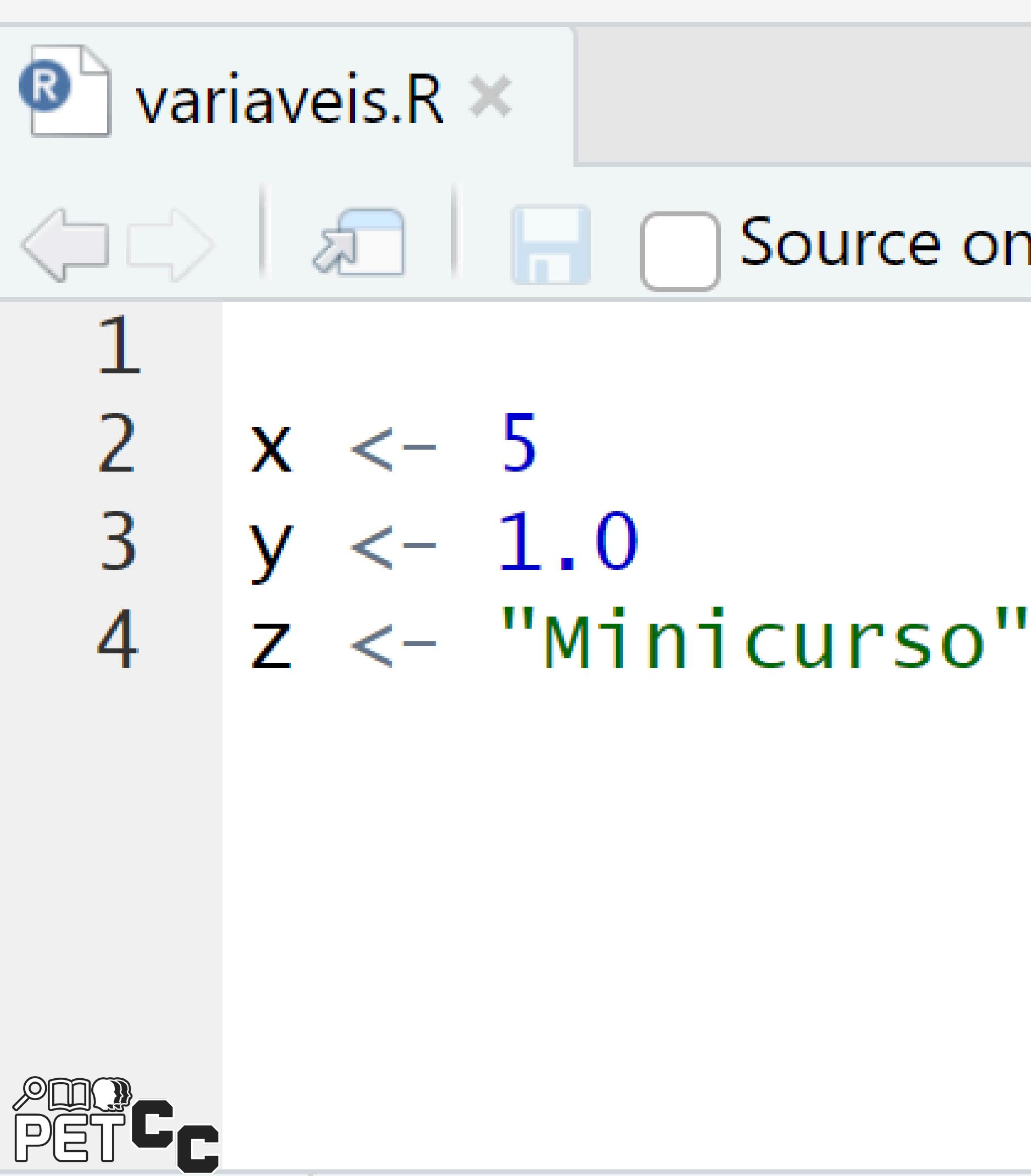
- O operador `<-` serve para designar um valor para uma variável
- Ou seja:
 - `x <- 5`
 - A variável `x` recebe o valor 5

Operadores

- O operador + serve para somar
- Ou seja:
 - $x \leftarrow x + 5$
 - A variável x recebe o valor
dela mesma + 5

Operadores

- Os operadores básicos são
 - + (adição)
 - - (subtração)
 - * (multiplicação)
 - / (divisão)
 - entre outros



```
R variaveis.R ×
← → | ↕ | H | Source on
1
2 x <- 5
3 y <- 1.0
4 z <- "Minicurso"
```

Declaração de variáveis

Diferente, por exemplo, da linguagem C, no R não é necessário declarar uma variável com o tipo.

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function Addins Project: (None)

variaveis.R x

Source on Save Run Source

1
2 x <- 5
3 y <- 1.0
4 z <- "Minicurso"

Environment History Connections Tutorial

Import 138 MiB Grid C

R Global Environment

Name Type Leng... Size Value

Environment is empty

Files Plots Packages Help Viewer Presentation

Zoom Export

4:17 (Top Level) R Script

Console Terminal Background Jobs

R 4.3.3 ~/ ↗

> |

CTRL + Enter para executar uma linha

As variáveis ficam salvas na memória

The screenshot shows the RStudio interface with the following components:

- Code Editor:** Displays the script file "variaveis.R" containing the following code:

```
1 x <- 5
2 y <- 1.0
3 z <- "Minicurso"
```
- Environment View:** Shows the global environment with the following variables:

Name	Type	Length	Size	Value
x	numerical	1	56 B	5
y	numerical	1	56 B	1
z	character	1	120 B	"Minicurso"
- Console View:** Displays the R session history:

```
R 4.3.3 · ~/minicurso> x <- 5
R 4.3.3 · ~/minicurso> y <- 1.0
R 4.3.3 · ~/minicurso> z <- "Minicurso"
R 4.3.3 · ~/minicurso>
```

The screenshot shows the RStudio interface with the following components:

- File**: File Edit Code View Plots Session Build Debug Profile Tools Help
- Addins**: Go to file/function Addins
- Project**: Project: (None)
- Code Editor**: variaveis.R x (containing the R code below)

```
1 x <- 5
2 y <- 1.0
3 z <- "Minicurso"
4 variavel <- x + y
5 print(variavel)
```
- Environment**: Shows the global environment with the following variables:

Name	Type	Length	Size	Value
variavel	numerical	1	56 B	6
x	numerical	1	56 B	5
y	numerical	1	56 B	1
z	character	1	120 B	"Minicurso"
- Console**: R 4.3.3 · ~/ ~
> x <- 5
> y <- 1.0
> z <- "Minicurso"
> variavel <- x + y
> print(variavel)
[1] 6
>

Text Overlay:

Para printar algo usa-se
print(nome_da_variavel)

Tipos de dados

- Character
- Numeric
- Integer
- Complex
- Logical

Outras características

1

Basicamente, todas variáveis
são vetores

2

Os vetores são iniciados em 1

3

Quando um valor não existe,
ele é representado por NA

Estrutura Condicional

Similar a outras linguagens, como o C.

```
if(x > 10){  
    print("O valor é maior que dez")  
}else{  
    print("O valor é menor ou igual a dez")  
}
```

Estruturas de Repetição

Similar a outras linguagens, como o C.

```
i = 1
while(vetor[i] <= 3) {
    print(vetor[i])
    i = i + 1
}
```

Estruturas de Repetição

Diferentemente de outras linguagens, a definição dos valores abordados pelo laço for é feita por um intervalo, sem necessidade de acréscimo visível.

```
for(i in 1:length(vetor)){
  print(vetor[i])
}
```

Sua vez:

Faça com que uma variável x receba o cálculo:

$$12*7$$

faça com que uma variavel y receba o cálculo:

$$x + 15$$

printe o valor da variável y na tela.

Faça com que uma variável texto com:
“y é maior que 15”
apareça na tela somente se y for maior que 15

Faça com que uma variável i receba os valores de 1 a 10 e faça com
que o dobro dela apareça na tela a cada repetição:
`print(i * 2)`