



Formation publications reproductibles avec  
RMarkdown

# 1 - Introduction

Maël THEULIERE - Marouane ZELLOU

# Déroulé de l'atelier

- **Markdown**: un outil de balisage pour écrire des documents
- **Rmarkdown**: un outil pour mixer du markdown avec du code R

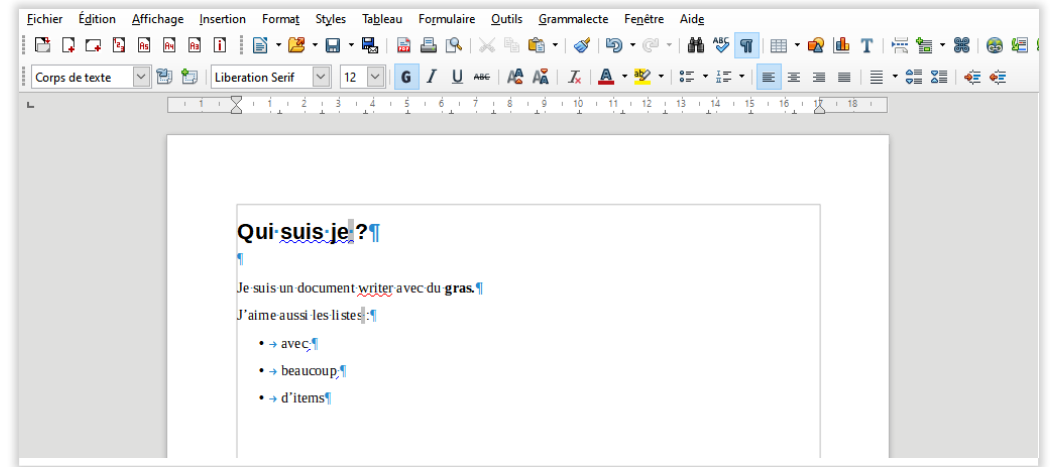
# Le problème

# Le problème

Un document word ou writer est un document lourd au format complexe, qui vous demande d'interagir avec une interface dédiée pour la mise en page.

Cette lourdeur entraine :

- des risques sur la mise en page (empilement de formattage)
- des risques lorsque vous travaillez à plusieurs (je n'ai pas la même mise en page que toi)
- ne permet pas d'historiser proprement ce qu'on fait



**La solution**

**HTML ?**

# HTML

## Qui suis je ?

Je suis un document writer avec du **gras**.

**J'aime aussi les listes :**

- **avec**
- **beaucoup**
- **d'items**

```
<html>
<head>
  <meta http-equiv="content-type" content="text/html; charset=utf-8">
</head>
<body>
  <h1>Qui suis je ?</h1>
  <p>Je suis un document writer avec du <b>gras.<b></p>
  <br>
  <p>J'aime aussi les listes :</p>
  <br>
  <ul>
    <li>avec</li>
    <li>beaucoup</li>
    <li>d'items</li>
  </ul>
</body>
</html>
```

# HTML

Le document html est complexe, le balisage rend compliqué la lecture du document source.

La solution ?  
markdown



# markdown

markdown est un langage de balisage

- **éditable** en fichier texte
  - **compréhensible** par l'être humain
  - permettant la mise en page **rapide** du texte
-

# Exemple

**# qui suis je ?**

Je suis un document writer avec du **\*\*gras\*\***.

J'aime aussi les listes :

- avec
- beaucoup
- d'items

## qui suis je ?

Je suis un document writer avec du **gras**.

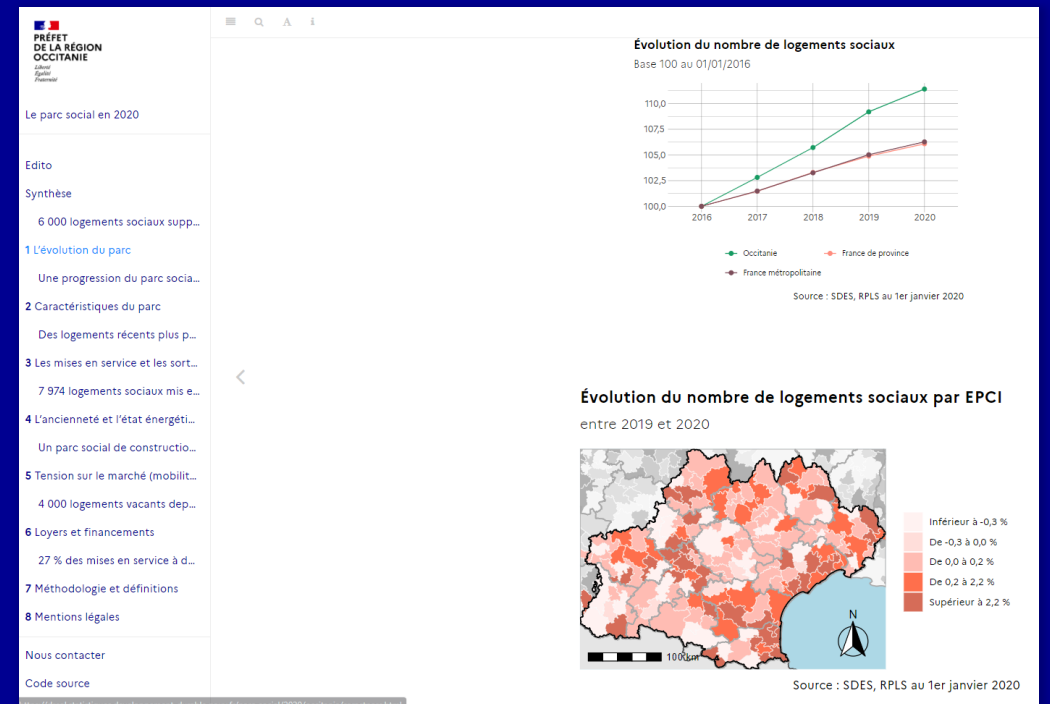
J'aime aussi les listes :

- avec
- beaucoup
- d'items

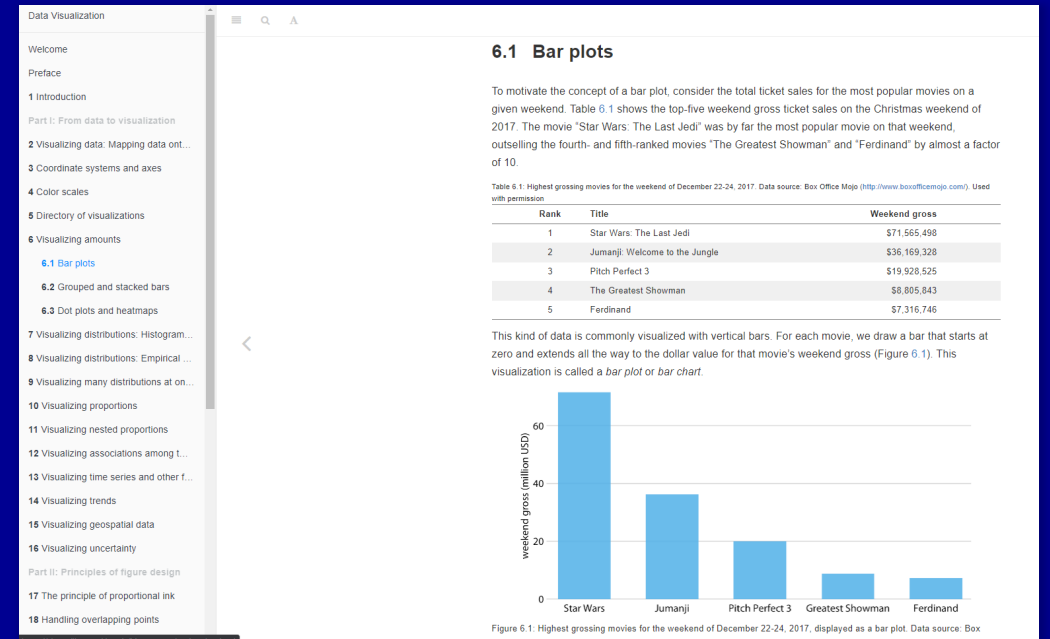
**Et bientôt...**

**Vous saurez faire...**

# Des rapports



# Des livres



# Des diaporama

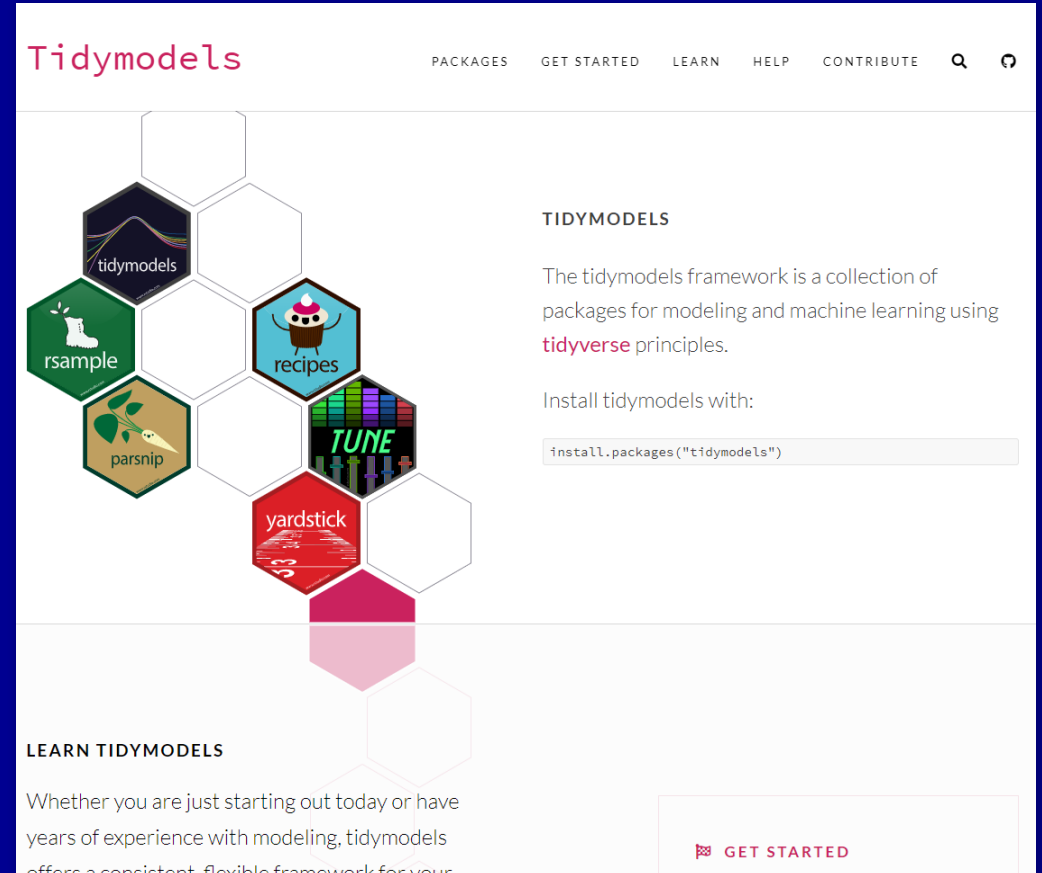


Formation publications reproductibles  
avec **RMarkdown**

## 1 - Introduction

Maël THEULIERE - Marouane ZELLOU

# Des sites web



The screenshot shows the homepage of the Tidymodels website. The header features the 'Tidymodels' logo in pink, followed by navigation links: 'PACKAGES', 'GET STARTED', 'LEARN', 'HELP', and 'CONTRIBUTE'. Search and GitHub icons are on the right. A central graphic displays a cluster of hexagons, each containing a package logo: 'tidymodels' (black with a line graph), 'rsample' (green with a boot), 'parsnip' (brown with a leaf), 'recipes' (blue with a robot), 'TUNE' (black with colorful bars), and 'yardstick' (red with a target). To the right, the 'TIDYMODELS' section describes the framework as a collection of packages for modeling and machine learning using 'tidyverse' principles. It includes the installation command: 

```
install.packages("tidymodels")
```

. The 'LEARN TIDYMODELS' section begins with the text: 'Whether you are just starting out today or have years of experience with modeling, tidymodels offers a consistent, flexible framework for your...'. A 'GET STARTED' button with a flag icon is located in the bottom right corner.

Tidymodels

PACKAGES GET STARTED LEARN HELP CONTRIBUTE 🔍

**TIDYMODELS**

The tidymodels framework is a collection of packages for modeling and machine learning using **tidyverse** principles.

Install tidymodels with:

```
install.packages("tidymodels")
```

**LEARN TIDYMODELS**

Whether you are just starting out today or have years of experience with modeling, tidymodels offers a consistent, flexible framework for your...

🚩 GET STARTED

Juste  
avec  
markdown



# Les bases du markdown

# Formater du texte

de *\*l'italique\**

du **\*\*gras\*\***

de **\*\*\*l'italique gras\*\*\***

de *l'italique*

du **gras**

de ***l'italique gras***

# Mettre des titres

```
# un titre 1
```

```
## un titre 2
```

```
### un titre 3
```

un titre 1

un titre 2

un titre 3

# Des listes non numérotées

Une liste avec :

- un item
- un autre item
- un dernier item

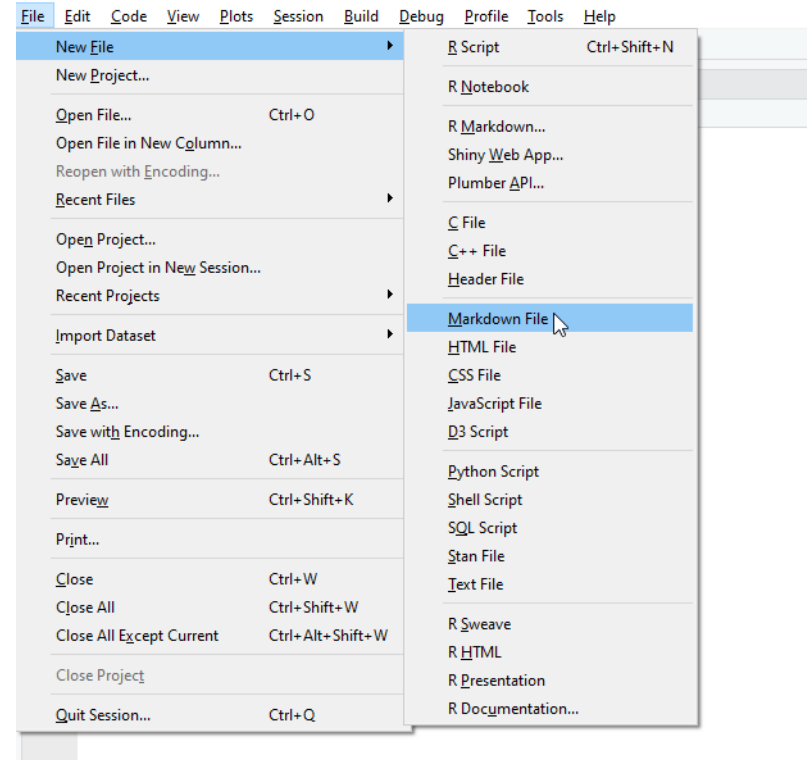
Une liste avec :

- un item
- un autre item
- un dernier item

A vous de jouer ✍️

# A vous de jouer

- Ouvrir Rstudio
- Créer un fichier markdown
- Créer un texte avec un titre, un paragraphe, une liste d'item
- Appuyer sur preview



**Aller plus loin sur le formatage markdown**

# Des listes numérotées

Une liste avec :

1. un item
2. un autre item
3. un dernier item

Une liste avec :

1. un item
2. un autre item
3. un dernier item



# Des listes imbriquées

Une liste avec :

1. un item
2. un autre item
  - avec un sous item
  - et un autre sous item
3. un dernier item

Une liste avec :

1. un item
2. un autre item
  - avec un sous item
  - et un autre sous item
3. un dernier item

# Des liens hypertexte

```
[Ceci](https://fr.wikipedia.org/)  
est un lien vers wikipedia.
```

[Ceci](https://fr.wikipedia.org/) est un lien vers wikipedia.

# Des images

`

Crédits : Kevin Pluck, Creative Commons Attribution 2.0 Générique

A vous de jouer ✍️

# A vous de jouer

- Reprendre votre texte précédent
  - Rajouter une image en dessous du titre principal
  - Rajouter à la fin un lien en savoir plus vers le site de votre choix.
-

# Comment utiliser markdown avec R ?

## Rmarkdown



# Rmarkdown

- Ecrire du texte en **markdown**
- Intégrer des tableaux et dataviz fait en **R**
- Pour produire toute sorte de **documents** :
  - diaporama
  - publications
  - sites web

## Le mapping

### Les paramètres du mapping

Dans l'exemple qui suit, la représentation géographique utilisée est le nuage de points

**geom\_point()**.

D'autres types de représentations géographiques sont présentés dans la partie suivante.

L'aesthetic sert à identifier les variables que l'on souhaite représenter.

Par exemple, si l'on souhaite représenter la longueur de la nageoire en fonction du poids des pingouins :

```
```{r, fig.height=3.5}
ggplot(data = penguins) +
  geom_point(aes(x = flipper_length_mm, y = body_mass_g))
```
```

De plus, la fonction **aes()** admet d'autres arguments qui permettent de modifier l'apparence du graphique selon une 3ème variable du jeu de données. Par exemple :

- **colour** : la couleur,
- **shape** : la forme,
- **size** : la taille,
- **alpha** : la transparence,
- **fill** : le remplissage ;

```
```{r, fig.height=3.5}
ggplot(data = penguins) +
  geom_point(aes(x = flipper_length_mm,
                 y = body_mass_g,
                 color = species,
                 shape = species))
```
```



# Decripter un fichier Rmarkdown



```
1 ---
2 title: "Introduction à ggplot2"
3 author: "Maël THEULIERE"
4 date: "22/03/2021"
5 output: html_document
6 ---
7
8
9 ## Le mapping
10
11 ### Les paramètres du mapping
12
13 Dans l'exemple qui suit, la représentation géographique utilisée est le nuage de points geom_point().
14 D'autres types de représentations géographiques sont présentés dans la partie suivante.
15
16 L'aesthetic sert à identifier les variables que l'on souhaite représenter.
17 Par exemple, si l'on souhaite représenter la longueur de la nageoire en fonction du poids des pingouins :
18
19 ```{r, fig.height=3.5}
20 library(ggplot2)
21 library(palmerpenguins)
22 ggplot(data = penguins) +
23   geom_point(aes(x = flipper_length_mm, y = body_mass_g))
24 ```
25
```



# Decripter un fichier Rmarkdown

```
first_rmd.Rmd x
1 ---
2 title: "Introduction à ggplot2"
3 author: "Maël THEULIERE"
4 date: "22/03/2021"
5 output: html_document
6 ---
7
8
9 ## Le mapping
10
11 ### Les paramètres du mapping
12
13 Dans l'exemple qui suit, la représentation géographique utilisée est le nuage de points geom_point().
14 D'autres types de représentations géographiques sont présentés dans la partie suivante.
15
16 L'aesthetic sert à identifier les variables que l'on souhaite représenter.
17 Par exemple, si l'on souhaite représenter la longueur de la nageoire en fonction du poids des pingouins :
18
19 ```{r, fig.height=3.5}
20 library(ggplot2)
21 library(palmerpenguins)
22 ggplot(data = penguins) +
23   geom_point(aes(x = flipper_length_mm, y = body_mass_g))
24 ```
25
```

Code R

# Decripter un fichier Rmarkdown



```
1 ---
2 title: "Introduction à ggplot2"
3 author: "Maël THEULIERE"
4 date: "22/03/2021"
5 output: html_document
6 ---
7
8
9 ## Le mapping
10
11 ### Les paramètres du mapping
12
13 Dans l'exemple qui suit, la représentation géographique utilisée est le nuage de points geom_point().
14 D'autres types de représentations géographiques sont présentés dans la partie suivante.
15
16 L'aesthetic sert à identifier les variables que l'on souhaite représenter.
17 Par exemple, si l'on souhaite représenter la longueur de la nageoire en fonction du poids des pingouins :
18
19 ```{r, fig.height=3.5}
20 library(ggplot2)
21 library(palmerpenguins)
22 ggplot(data = penguins) +
23   geom_point(aes(x = flipper_length_mm, y = body_mass_g))
24 ```
25
```

markdown

Code R

# Decripter un fichier Rmarkdown



```
1 ---
2 title: "Introduction à ggplot2"
3 author: "Maël THEULIERE"
4 date: "22/03/2021"
5 output: html_document
6 ---
7
8
9 ## Le mapping
10
11 ### Les paramètres du mapping
12
13 Dans l'exemple qui suit, la représentation géographique utilisée est le nuage de points geom_point().
14 D'autres types de représentations géographiques sont présentés dans la partie suivante.
15
16 L'aesthetic sert à identifier les variables que l'on souhaite représenter.
17 Par exemple, si l'on souhaite représenter la longueur de la nageoire en fonction du poids des pingouins :
18
19 ```{r, fig.height=3.5}
20 library(ggplot2)
21 library(palmerpenguins)
22 ggplot(data = penguins) +
23   geom_point(aes(x = flipper_length_mm, y = body_mass_g))
24 ```
25
```

YAML

markdown

Code R

# Decripter un fichier Rmarkdown

Le bouton knit pour compiler votre document



```
1 ---
2 title: "Introduction à ggplot2"
3 author: "Maël THEULIERE"
4 date: "22/03/2021"
5 output: html_document
6 ---
7
8
9 ## Le mapping
10
11 ### Les paramètres du mapping
12
13 Dans l'exemple qui suit, la représentation géographique utilisée est le nuage de points geom_point().
14 D'autres types de représentations géographiques sont présentés dans la partie suivante.
15
16 L'aesthetic sert à identifier les variables que l'on souhaite représenter.
17 Par exemple, si l'on souhaite représenter la longueur de la nageoire en fonction du poids des pingouins :
18
19 ```{r, fig.height=3.5}
20 library(ggplot2)
21 library(palmerpenguins)
22 ggplot(data = penguins) +
23   geom_point(aes(x = flipper_length_mm, y = body_mass_g))
24 ```
25
```

# Le YAML sert à définir les paramètres de votre document

C'est dans le YAML que vous aller définir :

- les auteurs
- la date de votre document
- le format de sortie.



```
1 ---
2 title: "Introduction à ggplot2"
3 author: "Maël THEULIERE"
4 date: "22/03/2021"
5 output: html_document
6 ---
7
8
9 ## Le mapping
10
11 ### Les paramètres du mapping
12
13 Dans l'exemple qui suit, la représentation géographique utilisée est le nuage de points geom_point().
14 D'autres types de représentations géographiques sont présentés dans la partie suivante.
15
16 L'aesthetic sert à identifier les variables que l'on souhaite représenter.
17 Par exemple, si l'on souhaite représenter la longueur de la nageoire en fonction du poids des pingouins :
18
19 ```{r, fig.height=3.5}
20 library(ggplot2)
21 library(palmerpenguins)
22 ggplot(data = penguins) +
23   geom_point(aes(x = flipper_length_mm, y = body_mass_g))
24
25
```

# Le YAML sert à définir les paramètres de votre document

```
---  
title: "Introduction à rmarkdown"  
author: "Marouane ZELLOU"  
date: "22/03/2021"  
output: html_document  
---
```

# Le YAML sert à définir les paramètres de votre document

On peut rajouter plusieurs auteurs :

```
---  
title: "Introduction à rmarkdown"  
author:  
  - "Maël THEULIERE"  
  - "Maël THEULIERE"  
date: "22/03/2021"  
output: html_document  
---
```

# Le YAML sert à définir les paramètres de votre document

On peut rajouter des options pour un output :

```
---
title: "Introduction à rmarkdown"
author:
  - "Maël THEULIERE"
  - "Marouane ZELLOU"
date: "22/03/2021"
output:
  html_document:
    toc: true
    theme: flatly
---
```

- **toc: true** permet de rajouter un sommaire à notre document.
- **theme: flatly** permet de changer le thème du document.



# Le YAML sert à définir les paramètres de votre document

On peut rajouter plusieurs outputs :

```
---
title: "Introduction à rmarkdown"
author:
  - "Maël THEULIERE"
  - "Marouane ZELLOU"
date: "22/03/2021"
output:
  pdf_document: default
  html_document:
    toc: true
    theme: flatly
---
```

A vous de jouer ✍️

# A vous de jouer

- Créer un document rmarkdown html
  - Mettez vous en auteur du document
  - Rajouter un theme spécifique
  - Chercher dans l'aide comment obtenir des titres numérotés
  - Cliquer sur *knit* pour compiler le document
-

# Le code R s'insère dans des chunks

C'est dans les chunks que vous pouvez insérer le code R que vous voulez voir exécuter. Chaque chunk doit être nommé, et peut recevoir des options qui permettent par exemple :

- d'afficher ou non le code (**echo**)
- de lancer ou non la commande rentrée (**eval**)
- d'afficher ou non le résultat du code (**include**)
- de modifier la largeur ou la hauteur d'une figure (**fig.width** et **fig.height**)

Les chunks sont gérés dans `{Rmarkdown}` par `{knitr}`. Il y a plus de 50 options possibles pour un chunk, vous pouvez trouver l'ensemble de la documentation [ici](#).



```
1 ---
2 title: "Introduction à ggplot2"
3 author: "Maël THEULIERE"
4 date: "22/03/2021"
5 output: html_document
6 ---
7
8 ## Le mapping
9
10 ### Les paramètres du mapping
11
12 Dans l'exemple qui suit, la représentation géographique utilisée est le nuage de points geom_point().
13 D'autres types de représentations géographiques sont présentés dans la partie suivante.
14
15 L'aesthetic sert à identifier les variables que l'on souhaite représenter.
16 Par exemple, si l'on souhaite représenter la longueur de la nageoire en fonction du poids des pingouins :
17
18 ```{r, fig.height=3.5}
19 library(ggplot2)
20 library(palmerpenguins)
21 ggplot(data = penguins) +
22   geom_point(aes(x = flipper_length_mm, y = body_mass_g))
23 ```
24
25
```

Code R

# Le code R s'insère dans des chunks

Vous pouvez aussi affecter des valeurs par défaut à l'ensemble de vos chunks avec la fonction `knitr::opts_chunk$set()` à intégrer dans votre premier chunk :

```
```${r, include=FALSE}
knitr::opts_chunk$set(
  echo = FALSE,
  eval=TRUE,
  include=TRUE,
  fig.width = 6
)
```
```



```
1 ---
2 title: "Introduction à ggplot2"
3 author: "Maël THEULIERE"
4 date: "22/03/2021"
5 output: html_document
6 ---
7
8 ## Le mapping
9
10
11 ### Les paramètres du mapping
12
13 Dans l'exemple qui suit, la représentation géographique utilisée est le nuage de points geom_point().
14 D'autres types de représentations géographiques sont présentés dans la partie suivante.
15
16 L'aesthetic sert à identifier les variables que l'on souhaite représenter.
17 Par exemple, si l'on souhaite représenter la longueur de la nageoire en fonction du poids des pingouins :
18
19 ```{r, fig.height=3.5}
20 library(ggplot2)
21 library(palmerpenguins)
22 ggplot(data = penguins) +
23   geom_point(aes(x = flipper_length_mm, y = body_mass_g))
24 ```
25
```

Code R

# Le texte s'écrit partout ailleurs

Partout ailleurs, le texte va pouvoir s'écrire avec la syntaxe *markdown*.



```
1 ---
2 title: "Introduction à ggplot2"
3 author: "Maël THEULIERE"
4 date: "22/03/2021"
5 output: html_document
6 ---
7
8
9 # Le mapping
10
11 ## Les paramètres du mapping
12
13 Dans l'exemple qui suit, la représentation géographique utilisée est le nuage de points geom_point().
14 D'autres types de représentations géographiques sont présentés dans la partie suivante.
15
16 L'aesthetic sert à identifier les variables que l'on souhaite représenter.
17 Par exemple, si l'on souhaite représenter la longueur de la nageoire en fonction du poids des pingouins :
18
19 ```{r, fig.height=3.5}
20 library(ggplot2)
21 library(palmerpenguins)
22 ggplot(data = penguins) +
23   geom_point(aes(x = flipper_length_mm, y = body_mass_g))
24 ```
25
```

markdown

# Formater un document en plusieurs colonnes

Vous aurez besoin souvent de pouvoir par exemple définir un document en plusieurs colonne, avec par exemple une illustration sur une moitié de la page, un texte sur l'autre.

Pour cela vous allez devoir séparer vos deux blocs dans des "div".

Dans rmarkdown, des div commencent par `::: {}` et finissent par `:::`. Pour ensuite que le document aligne ces blocs l'un à côté de l'autre, vous devrez utiliser la propriété css `display: flex;` dans une `div` englobante.

## # Le mapping

### ## Les paramètres du mapping

```
:::: {style="display: flex;"}
```

```
::: {}
```

Dans l'exemple qui suit, la représentation géographique  
D'autres types de représentations géographiques

L'aesthetic sert à identifier les variables qu  
Par exemple, si l'on souhaite représenter la l

```
:::
```

```
::: {}
```

```
```{r, fig.height=3.5}
```

```
library(ggplot2)
```

```
library(palmerpenguins)
```

```
ggplot(data = penguins) +
```

```
  geom_point(aes(x = flipper_length_mm, y = bo
```

```
```\n:::
```

A vous de jouer ✍️



A vous de jouer 

Insérer un exercice