

Slides con Quarto

(Web del curso [aquí](#))

26 de junio de 2023

Intro

0. [ChatGPT](#) puede hacer slides, pero Quarto también

1. Quarto puede hacer slides de varios tipos (ppt, slidy, beamer)

2. Nos centraremos en **Revealjs**

Revealjs is highly recommended unless you have specific requirements for Office or LaTeX output

Presentaciones con Quarto (genérico)

- Las slides se escriben en QMD ... así que ya sabemos 😊
- Hay algunas especificidades que vamos a ver
- Veremos primero aspectos comunes a todos los formatos de slides
- [Documentación oficial](#)

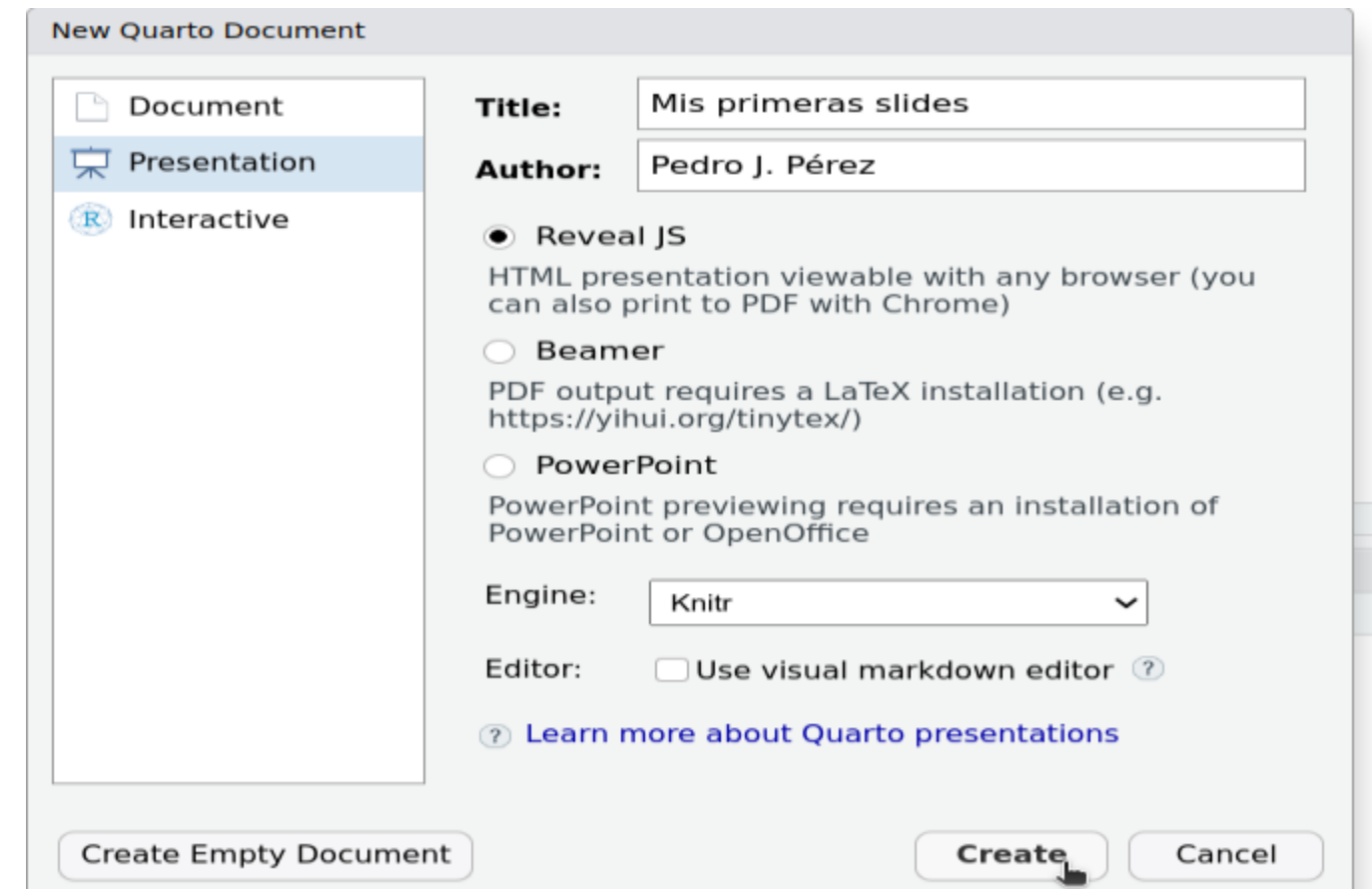
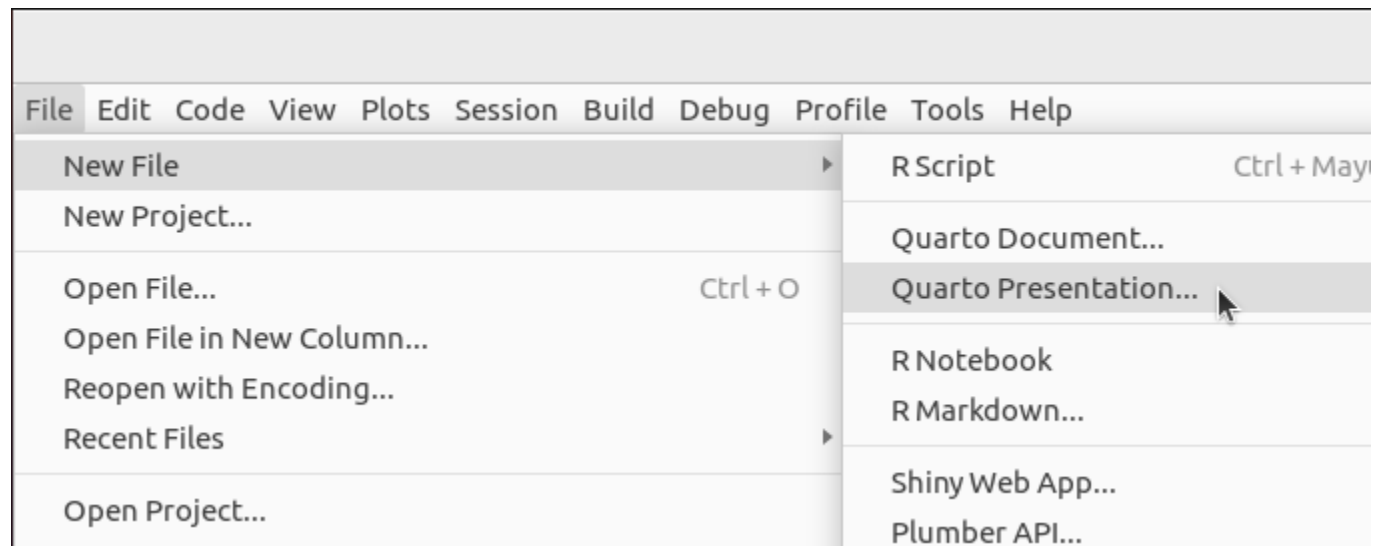
Creación de unas nuevas transparencias

1. Has de seguir la siguiente ruta de menús:

File > New File > Quarto Presentation.

2. Se abrirá una interfaz gráfica donde podrás elegir entre **3 formatos de slides** (Revealjs, Beamer y ppt). También podrás poner título y autor.

En imágenes:



Creación de una nueva slide

- Los títulos de **primer nivel** crean una nueva slide (y tb una nueva sección)
- Los títulos de **segundo nivel** crean una nueva slide.
- También se puede crear una nueva slide usando 3 o mas **guiones cortos**: - - -

Extensión: modificar la generación de nuevas slides

Hay que usar `slide-level:` en el YAML. Por ejemplo:

```
1 ---
2 title: "Mis slides"
3 format: revealjs
4 ---
```

Por defecto, los títulos de primer(#) y segundo nivel (##) crean nuevas slides

```
1 ---
2 title: "Mis slides"
3 format:
4   revealjs:
5     slide-level: 3
6 ---
```

Con este yaml, los títulos de tercer nivel (con `###`) también crearían una nueva slide

Pausas en la presentación

- Por defecto, los elementos de una slide se muestran todos de golpe



Para crear pausas en la presentación has de usar las siguientes marcas:

. . .

(es decir, tres puntos separados por espacios)

- Por ejemplo:

```
1  ## Una nueva slide
2
3  Contenido nº 1
4
5  . . .
6
7  Contenido nº 2
```

(para ver el Contenido nº 2 habrá que dar al avance página)

Listas incrementales

- Por defecto las listas se muestran de golpe
- Si quieres que se muestren elemento a elemento, puedes controlarlo **desde el YAML:**

```
1 title: "My Presentation"
2 format:
3   revealjs:
4     incremental: true
```

- Se puede ir **intercalando** listas incrementales, y no-incrementales, con los siguientes **fenced div's**:

Lista incremental :

```
1 ::: {.incremental}
2 - First we take Manhattan,
3 - then we take Berlin
4 :::
```

Lista no-incremental :

```
1 ::: {.nonincremental}
2 - First we take Manhattan,
3 - then we take Berlin
4 :::
```

Columnas multiples

- Como puedes ver, se puede generar contenido en **varias columnas**. Para ello hay que usar “un **fenced div** con clase `columns`”.

To put material in side by side columns, you can use a native div container with class `.columns`, containing two or more div containers with class `.column` and a width attribute. – [Documentacion oficial](#) de Quarto

- Por ejemplo:

```
1  :::: {.columns}
2
3  ::: {.column width="40%"}
4  Mi contenido en la primera columna
5  :::
6
7  ::: {.column width="60%"}
8  Contenido en la segunda columna
9  :::
10
11 ::::
```


Revealjs slides

- [reveal.js](#) es un open source HTML presentation framework
- A mi me gustan: en mi opinión tienen unas cuantas ventajas
- Documentación oficial [aquí](#)
- Listado (extenso) de opciones [aquí](#)

Revealjs slides (elementos básicos)

- Elementos “básicos”, que mejoran bastante las slides

Revealjs: Slide background

- Las slides mejoran bastante estéticamente si en alguna de ellas cambiamos el fondo. La documentación está [aquí](#)
- Se pueden usar los siguientes elementos: **color**, **imagen**, **vídeo** e **iframe**. Por ejemplo:

```
1
2 ## Título de la slide {background-color="aquamarine"}
3
4 ## Título de la slide {background-color="#A9339A"}
5
```

Extensión: Opciones más complejas

```
1 ## Slide Title {background-video="video.mp4" background-video-loop="true" background-video-muted="true"}
2
3 ## Slide Title {background-color="black" background-image="https://placekitten.com/100/100" background-size="100px" background-repeat="repeat"}
4
5 ## Slide Title {background-iframe="https://www.uv.es"}
```

Slide Backgrounds: ejemplo 1 (color)

Para poner un color de fondo:

```
1  
2 ## Slide Backgrounds: ejemplo 1 (color) {background-color="aquamarine"}
```

Slide Backgrounds: ejemplo 2 (imagen)

Para poner una imagen de fondo:

```
1  
2 ## Slide Backgrounds: ejemplo 2 (imagen) {background-image="imagenes/perro_blanco.jpg" background-size="cover"}
```



Se pueden hacer cosas muy chulas como:

- [Slides Emil Hvitfeldt](#)
- Parallax background. Documentació
- [Quarto iFrame Examples](#). El [repo](#)

Revealjs: Tabsets

- Por ejemplo este tabset

Adivinanza

Solución

Oro parece, plátano es ...

- se ha creado con este código:

```
1 ::: {.panel-tabset}
2 ## Adivinanza
3
4 Oro parece, platano es ...
5
6 ## Solución
7
8 😊
9 :::
```

Revealjs: Numeración de las slides

- Por defecto las slides están numeradas

```
1 ---
2 format:
3   revealjs:
4     slide-number: true #- {true, false}
5 ---
```

- Puedes hacer que una slide **no se cuente**:

```
1 ## Título de la slide {visibility="uncounted"}
```

- Puedes hacer cosas más complejas. Documentación oficial [aquí](https://revealjs.com/)

```
1 ---
2 format:
3   revealjs:
4     slide-number: c/t    #- current/total {c/t, c, ...}
5 ---
```


Revealjs: tunear con themes

- Podemos cambiar la apariencia de las slides usando los **themes**.
- Listado de themes [aquí](#).
- Para ello tenemos que incluir en el YAML el theme que queremos:

```
1 ---
2 title: "Presentation"
3 format:
4   revealjs:
5     theme: dark
6 ---
```

Hay extensiones para tunear

- Además disponemos de muchos otros themes en **extensiones** en [la web de Quarto](#) y en [Awesome Quarto](#)
- Por ejemplo: los themes [4 estaciones](#), [Inversa](#), [Insper](#), ... [Revealjs Earth theme](#) impresiona.

Revealjs slides (más elementos)

- “Más elementos”, que no son imprescindibles

Revealjs: la primera slide

- La primera slide **se crea automáticamente si pones un título en el YAML**
 - En ese caso, la 1ª slide contendrá los **metadatos**: title, subtitle, author, institute, date
- La **apariciencia de la primera slide** dependerá del tema elegido. **Listado de themes [aquí](#).**
- Tenemos muchas opciones para **personalizar las slides** y, en concreto, para **tunear la primera slide**

Revealjs: tuneando la primera slide

Extensión: modificar el aspecto de la primera slide con YAML

Aquí tienes algunas opciones en el YAML para la primera slide

```
1 ---
2 title: "Mys slides"
3 format:
4   revealjs:
5     theme: default
6     center-title-slide: true           #- centrado vertical
7 title-slide-attributes:
8   data-background-color: "#562457"    #- color del background
9   data-background-image: imagenes/my_imagen.jpg #- ruta a imagen
10  data-background-size: cover          #- {cover, contain}
11  data-background-opacity: "0.4"
12 ---
```

Extensión: modificar el aspecto de la primera slide manualmente

```
1 # My slides {background-image="imagenes/my_imagen.jpg" background-size="cover" background-color="#4f6952" background-opacity="0.5"}
2
3 <h2>Tuneando la primera slide</h2>
4
5 <h3>Pedro J. Pérez \ | 2023-04-04</h3>
```

Rvealjs: el Footer

- Podemos incorporar contenido en el **píe de página** de las slides.
- El footer habitual de estas slides se hace así:

```
1 ---
2 title: "My slides"
3 format:
4   revealjs:
5     footer: "[https://go.uv.es/pjperez/intro.to.quarto] (https://go.uv.es/pjperez/intro.to.quarto)"
6 ---
```

- Pero si te fijas, he cambiado el footer específicamente para esta slide. Lo he hecho con:

```
1 ::: {.footer}
2 <https://www.uv.es>
3 :::
```

Rvealjs: Logo

- También podemos poner un logo. Por defecto se colocará en la esquina inferior-izquierda. La numeración pasaría a estar en la esquina superior-izquierda

```
1 ---
2 title: "My slides"
3 format:
4   revealjs:
5     footer: "[https://go.uv.es/pjperez/intro.to.quarto] (https://go.uv.es/pjperez/intro.to.quarto) "
6     #logo: imagenes/economia.gif
7     logo: "https://upload.wikimedia.org/wikipedia/commons/8/8c/Esuni.GIF"
8 ---
```

- Hay **extensiones** que permiten incorporar más elementos y dar más flexibilidad al footer y al header de nuestras slides. Por ejemplo [esta](#) extensión permite hacer slides como [estas](#)

Revealjs: Content overflow

Si el **contenido** que queremos poner es mucho y no se ajusta, **no cabe en la slide**, tenemos varias posibilidades:

1. Usar la `.smaller` class al crear la slide:

```
1  
2 ## Título de la slide {.smaller}
```

2. Modificar el YAML para hacer las slides “scrollables”:

```
1 ---  
2 title: "My Presentation"  
3 format:  
4   revealjs:  
5     scrollable: true  
6 ---
```



Revealjs: Transiciones

- Documentación oficial [aquí](#)

Transition	Description
<code>none</code>	No transition (switch instantly)
<code>fade</code>	Cross fade
<code>slide</code>	Slide horizontally
<code>convex</code>	Slide at a convex angle
<code>concave</code>	Slide at a concave angle
<code>zoom</code>	Scale the incoming slide so it grows in from the center of the screen.

```
1 ---
2 title: "Presentation"
3 format:
4   revealjs:
5     transition: slide      #- transiciones de las slide: {none, fade, slide, convex, concave, or zoom}
6     transition-speed: fast #- {default, fast, slow}
7 ---
```


Revealjs: Transiciones

- Las transiciones se pueden fijar para una slide:

```
1 ## Título de la slide {transition="convex" transition-speed="slow"}
```

- y hacerlas más complejas

```
1 ## Título de la slide {transition="fade-in slide-out"}
```

Revealjs: Auto-animaciones

- Se puede auto-animar 2 slides próximas. Documentación [aquí](#)

```
1
2 # {auto-animate=true}
3
4 ¿Esto te gusta?
5
6 ## {auto-animate=true}
7
8 ¿Esto te gusta?
9
10 Sí, esto me gusta
11
12 ## {auto-animate=true}
13
14 Sí, esto me gusta **MUCHO**
```

- Se pueden usar **las siguientes propiedades CSS**: position, font-size, line-height, color, background-color, padding y margin.

```
1 ## {auto-animate=true}
2
3 ::: {style="margin-top: 100px;"}
4 Contenido auto-anímado
```

<https://go.uv.es/pjperez/intro.to.quarto>

```
5   :::  
6  
7   ## {auto-animate=true}  
8  
9   ::: {style="margin-top: 200px; font-size: 3em; color: red;"}  
10 Contenido auto-anímado  
11 :::
```

Revealjs: Auto-animaciones (chunks)

- También se puede auto-animar chunks de código. Documentación [aquí](#)

```
1  ## {auto-animate="true"}
2
3  ```r
4  #####- Hagamos un ggplot
5  library(ggplot2)
6  ggplot(iris) +
7  ```
8
9  ## {auto-animate=true}
10
11 ```r
12 #####- Hagamos un ggplot
13 library(ggplot2)
14 ggplot(iris) +
15     geom_point(aes(x = Sepal.Length,
16                    y = Petal.Length,
17                    color = Species))
18 ```
```

Revealjs: Asides

- Los “asides” son similares a las notas al pie de página, pero no llevan numeración.
- Su sintaxis es:

```
1 ::: aside
2 El contenido de un aside se colocará al final de la slides, justo encima de las notas al pie
3 :::
```

Revealjs: Más cosas

- Puedes ocultar una slide:

```
1 ## Título de la slide {visibility="hidden"}
```

- Tamaño de las slides. Documentación oficial [aquí](https://go.uv.es/pjpperez/intro.to.quarto)

Revealjs: Alineación de los títulos

- Los títulos de primer nivel (#) aparecen verticalmente centrados
- Los títulos de segundo nivel (##) aparecen en el margen superior
- Todo se puede cambiar con CSS, o ...
- Si quieres alinear un título a la derecha, una posibilidad es:

```
1 ::: {style="text-align: right;"}
2
3 <h2>Right aligned!</h2>
4
5 ### Right aligned tb!
6 :::
```

Right aligned!
Right aligned tb!

Revealjs: ajustar texto con “r-fit-text”

NO

QUIERO!!!

```
1 ::: r-fit-text
2 NO
3
4 QUIERO!!
5 :::
```

Revealjs: Fragments

- Los **fragments** se usan para resaltar o mostrar poco a poco algunos elementos de una slide: cada elemento con clase `.fragment` aparecerá sólo cuando avancemos a la siguiente slide. Documentación [aquí](#). Hay muchas [clases de fragments](#)

```
1 ::: {.fragment}
2   Fade in
3   :::
4
5   ::: {.fragment .fade-out}
6   Fade out
7   :::
8
9   ::: {.fragment .highlight-red}
10  Highlight red
11  :::
12
13  ::: {.fragment .fade-in-then-out}
14  Fade in, then out
15  :::
16
17  ::: {.fragment .fade-up}
18  Slide up while fading in
19  ...
```

Fade in

Fade out

Highlight red

Slide up while fading in

Revealjs: Fragments en spans

- El uso más directo que veo a los fragments es usarlos con spans para hacer esto:

```
1 Esto es muy [importante!!]{.fragment .highlight-red}
```

Esto es muy importante!!

- Otro ejemplo:

```
1 Los datos indican que [Quarto]{.fragment .highlight-green} mientras que [Rmarkdown]{.fragment .fade-in} va desapareciendo
```

Los datos indican que Quarto crece mucho, mientras que Rmarkdown va desapareciendo

Revealjs: Imágenes

- Por defecto se alinean a la izquierda

```
1 {fig-align="center"}
```



Imágenes: con o sin Stretch

- Por defecto, Revealjs hace que las imágenes se ajusten al espacio vertical que queda en la slide.
- Puedes **desactivar esta opción** en el YAML y/o en slide o imagen concreta. Documentación [aquí](#)
- En el YAML
- En una slide o imagen concreta

```
1 format:  
2   revealjs:  
3     auto-stretch: false
```

```
1 ## Nueva slide {.nostretch}  
2  
3  {.nostretch}
```



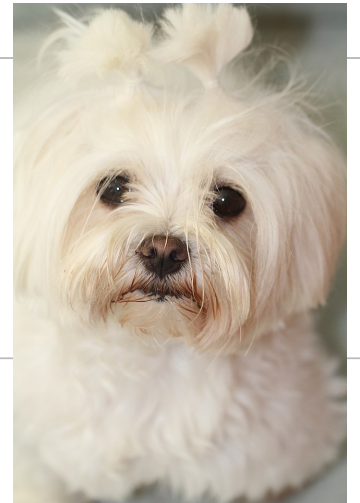
Imágenes: Stack layout

```
1 ::: {.r-stack}
2 {.fragment width="350"}
3
4 {.fragment width="400"}
5 :::
```



Imágenes: Posición absoluta

```
1  {.absolute top=500 left=0 width="250" height="100"}
2
3  {.absolute top=50 right=50 height="250"}
4
5  {.absolute bottom=30 right=-100 width="300" height="300"}
```



Revealjs: Código

Documentación [aquí](https://github.com/hakimel/reveal.js)

Chunks: code line-highlighting

- Podemos hacer que se enfaticen ciertas líneas de los chunks. Para ello has de poner ciertas opciones dentro de los chunks:

```
1 #| code-line-numbers: "3"
```

```
1 #| code-line-numbers: "|4"
```

```
1 #| code-line-numbers: "|3|5|"
```

```
1 library(ggplot2)
2 mtcars |>
3   ggplot(aes(x = disp, y = mpg)) +
4   geom_point() +
5   geom_smooth(method = "loess", formula = "y~x")
```

```
1 library(ggplot2)
2 mtcars |>
3   ggplot(aes(x = disp, y = mpg)) +
4   geom_point() +
5   geom_smooth(method = "loess", formula = "y~x")
```

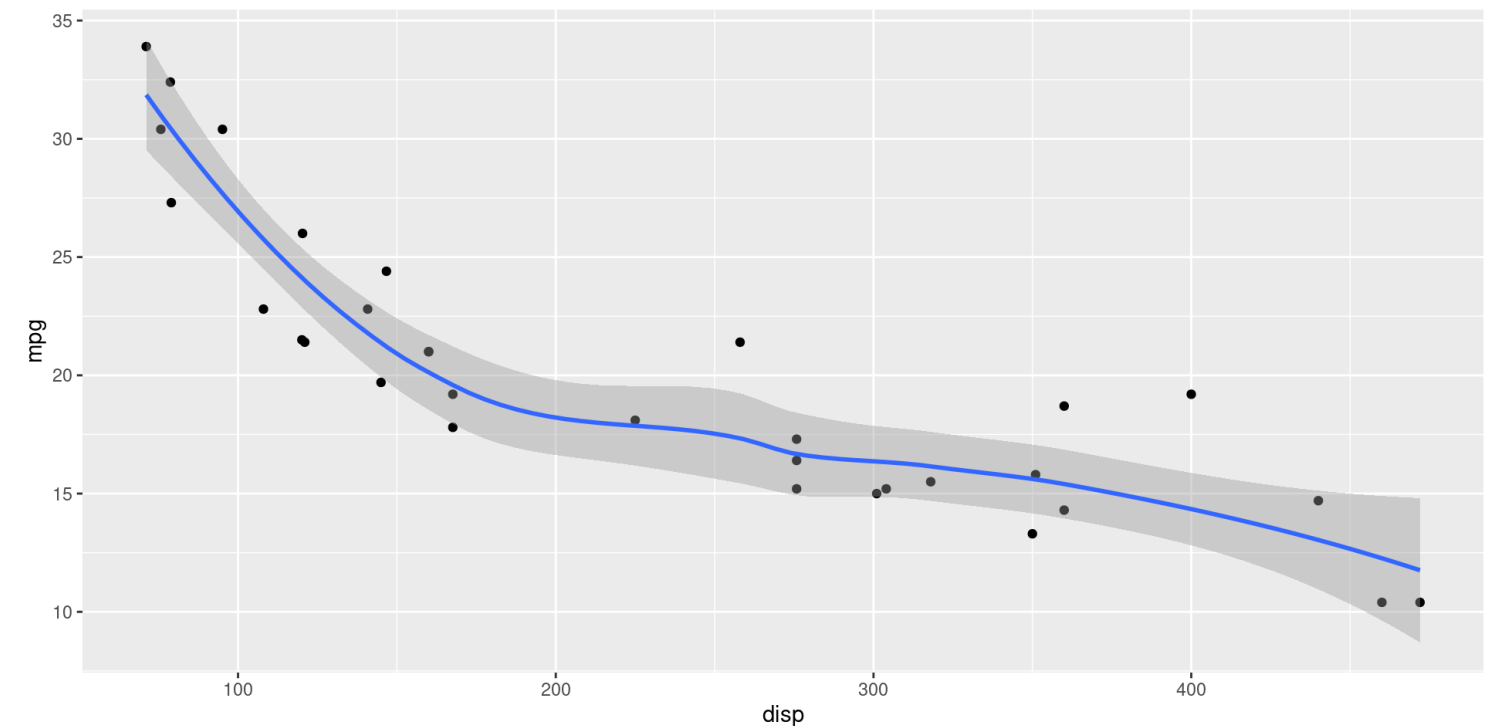
```
1 library(ggplot2)
2 mtcars |>
3   ggplot(aes(x = disp, y = mpg)) +
4   geom_point() +
5   geom_smooth(method = "loess", formula = "y~x")
```

- Documentación oficial [aquí](#). Un buen [post](#) sobre el tema, con otras opciones de sintáxis.

Chunks: Output location

- La opción: `#| output-location:` nos ayuda a colocar el código y los resultados side-by-side. Tiene los siguientes valores: {column-fragment, column, fragment, slide}
- La opción que más me gusta es: `#| output-location: column-fragment`, porque coloca el chunk y el resultado side-by-side y los separa por una pausa. Documentación oficial [aquí](#). Mine Çetinkaya-Rundel lo explica [aquí](#)

```
1 ```{r}
2 #| output-location: column-fragment
3 library(ggplot2)
4 mtcars |>
5   ggplot(aes(x = disp, y = mpg)) +
6   geom_point() +
7   geom_smooth(method = "loess", formula = "y~x")
8 ```
```



Revealjs: tuneando las slides

- Lo mejor es elegir un “theme”
- Hay otras opciones

Revealjs: tunear con themes

- Podemos cambiar la apariencia de las slides usando los **themes**.
- Listado de themes [aquí](#).
- Para ello tenemos que incluir en el YAML el theme que queremos:

```
1 ---
2 title: "Presentation"
3 format:
4   revealjs:
5     theme: dark
6 ---
```

Hay extensiones para tunear

- Además disponemos de muchos otros themes en **extensiones** en [la web de Quarto](#) y en [Awesome Quarto](#)
- Por ejemplo: los themes [4 estaciones](#), [Inversa](#), [Insper](#), ... [Revealjs Earth theme](#) impresiona.

Revealjs: Ejemplos de presentaciones

- [Some good practices for research with R](#) de Etienne Bacher, el repo [aquí](#)
- [Debugging](#) de Shannon Pileggi, el repo [aquí](#)
- [The Genesis of Experimentation](#) de Emi Tanaka, el repo [aquí](#)
- [ggdensity: Improved density visualization in R](#) de James Otto y David Kahle, el repo [aquí](#)
- [Extending ggplot2 with custom Geoms and Stats](#) de James Otto, el repo [aquí](#)
- [Quarto/RMarkdown - What's Different?](#), de TED Laderas. [Vídeo](#)
- [An Antarctic Tour of the Tidyverse](#) de Silvia Canelon, el repo [aquí](#)
- [Hello Quarto](#) de Mine Çetinkaya-Rundel y Julia Stewart Lowndes, el repo [aquí](#), [más cosas](#)
- [Reproducible authoring with Quarto](#) de Mine Çetinkaya-Rundel, el repo [aquí](#) [veamos ,un poco, [estas slides](#)]
- [What can I do with Quarto?](#) de Nicola Rennie, su web [aquí](#)
- [Reproducible Publications w/ Python and Quarto](#) de Thomas Mock
- [Improvements in text preprocessing using textrecipes](#) de Emil Hvitfeldt
- [The Happiest Notebooks on Earth](#) de Allison Hill
- [Quarto Presentations](#) by Quarto team, el repo [aquí](#). Documentación oficial [aquí](#)
- [Intro to Quarto](#) de Isabella Velásquez, el repo [aquí](#)
- [Un buen post sobre Revealjs slides](#) de Meghan Hall
- [Data Science Portfolio with Quarto](#) de Deepsha Menghani, el repo [aquí](#). Vídeo [aquí](#)
- [05 - Presentations Making Slides with Code](#) y [02 - Authoring Quarto](#) de Thomas Mock. Slides que pertenecen al siguiente taller [Get started with Quarto](#)
- [Creating your data science portfolio](#), de Nicola Rennie. Sus talks [aquí](#), el repo [aquí](#)

Revealjs: otras opciones de tuneado

- Podemos cambiar la apariencia con **CSS** ...
- ... y con **SASS**

Tunear con CSS: ejemplo 1 (fenced divs), sacado de [aquí](#)

1) Si incluyes este trozo de CSS, que define la clase “.big-text” en un archivo `.css` o `.scss`

```
1  .big-text {  
2  font-size: 88px;  
3  }
```

2) y lo referencias en el yaml de las slides con

- `theme: [default, assets/my_styles.scss]` o con
- `css: assets/my_styles.css`

3) Entonces, puedes usar el siguiente `fenced div` con clase “.big-text”:

```
::: {.big-text}  
  Un párrafo con el texto muy grande  
:::
```

Para conseguir que lo que escribas dentro del fenced div se vea muy grande, tal como ves abajo:

Un párrafo con el texto muy grande

Tunear con CSS: ejemplo 2 (spans)

- Ejemplo 2 (sacado de [aquí](#))
- Para tunear un elemento dentro de una frase tenemos que usar spans
- La sintaxis es: `[span]{.class}`
- De forma que,
 - si escribes esto: `Esta mañana ha llovido [mucho]{.big-text}`
- verás esto: Esta mañana ha llovido **mucho**

Revealjs: “truquillos”

Revealjs: **Trucos** para usar durante una presentación

Por ejemplo:

- Tenemos el “hamburger menu” para movernos por las slides
- Podemos hacer **ZOOM** en cualquier elemento o trozo de las slides. Solo hay que mantener presionada la tecla ALT y clickar en el ratón. En Linux es la tecla CTRL
- O: permite hacer una **overview** de las slides. Para moverte por las slides en modo overview “usa las flechitas”
- F: slides a **pantalla completa**
- CTRL-SHIFT-F : permite **buscar palabras** dentro de las slides. Dando a ENTER avanzas a la siguiente coincidencia. Además, Si lo haces mientras las slides están en overview te marcará todas las coincidencias

Revealjs: EXTRAS

Documentación [aquí](#) y [aquí](#)

- [Navigation menu](#)
- [Chalkboard](#)
- [Multiplex](#)
- [Speaker view](#)
- [Revealjs Plugins](#). Por ejemplo [fullscreen](#)

Otros formatos

- Beamer
- Power Point
- Slidy

Beamer slides

Beamer slides

- Beamer es un formato de slides, basadas en Latex, **muy popular en el mundo académico**
- Quarto puede hacer Beamer slides. Sólo hay que usar los menús o este yaml:

```
1 ---
2 title: "My Presentation"
3 format: beamer
4 ---
```

- La apariencia visual de las slides vendrá determinada por el **theme** y el **colortheme**. [Aquí](#) y [aquí](#) tienes ayuda para elegirlos. Por ejemplo:

```
1 ---
2 title: "Presentation"
3 format:
4   beamer:
5     aspectratio: 32
6     navigation: horizontal
7     theme: metropolis
8     colortheme: lily
9 ---
```

Beamer slides (documentación)

- [User Guide](#) oficial de Beamer
- Documentación oficial Beamer de Quarto [aquí](#)
- Opciones de YAML de Beamer en Quarto [aquí](#)
- [Beamer en RMD](#)
- [Beamer en StackOverflow](#). Por ejemplo:
 - [Numeración de las slides como en Revealjs](#)
 - [Justificar texto](#)

Más de slides

Algunos truquillos

Algunos truquillos

- Poner varios autores: una solución [aquí](#)

