

# RANA FLEXBOX

◀ Nivel 23 de 24 ▶

Las ranas han tenido una fiesta, pero es momento de volver a casa. Usa una combinación de **flex-direction** y **align-content** para llevarlas a sus hojas de lirio.

```
1 #estanque {  
2   pantalla: flexible;  
3   envoltura flexible: envoltura;  
4   /*jhonatan*/flex-direction: column-reverse;  
5   align-content: center;  
6 }
```

Siguiente

Flexbox Froggy es una creación de [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Ajustes](#)

¿Quieres aprender CSS grid? Juega [al Jardín de Cuadrículas](#).



# RANA FLEXBOX

Nivel 22 de 24

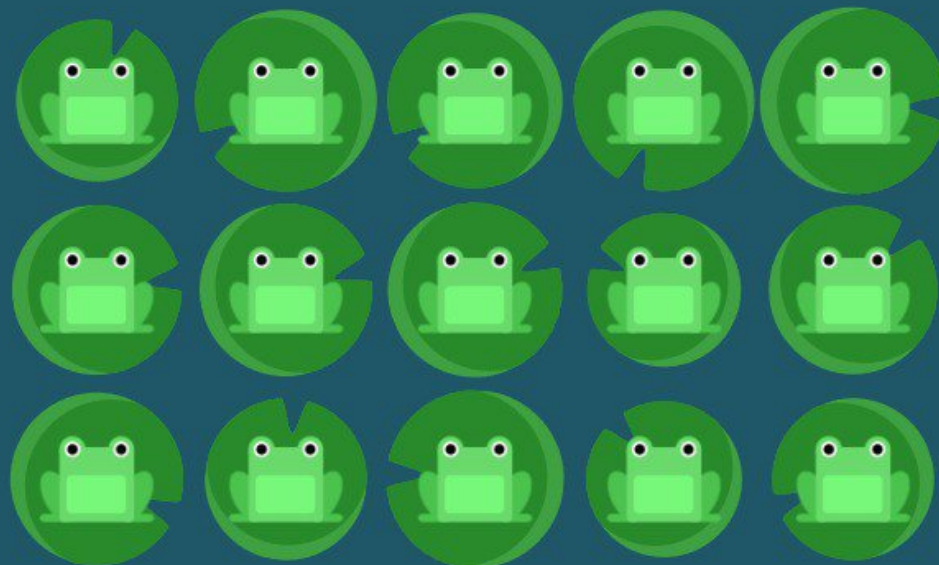
Ahora, la corriente ha agrupado las hojas de lirio en la parte inferior. Usa `align-content` para guiar las ranas hasta allí.

```
1 #estanque {  
2   pantalla: flexible;  
3   envoltura flexible: envoltura;  
4   /*jhonatan*/align-content: flex-end;  
5 }  
6  
7  
8  
9  
10
```

Siguiente

Flexbox Froggy es una creación de [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Ajustes](#)

¿Quieres aprender CSS grid? Juega [al Jardín de Cuadrículas](#).





# RANA FLEXBOX

Nivel 21 de 24

Las ranas están repartidas por todo el estanque, pero las hojas de lirio están agrupadas en la parte superior. Puedes usar **align-content** para establecer como múltiples líneas están separadas una de otra. Esta propiedad acepta los siguientes valores:

- **flex-start**: Las líneas se posicionan en la parte superior del contenedor.
- **flex-end**: Las líneas se posicionan en la parte inferior del contenedor.
- **center**: Las líneas se posicionan en el centro (verticalmente hablando) del contenedor.
- **space-between**: Las líneas se muestran con la misma distancia entre ellas.
- **space-around**: Las líneas se muestran con la misma separación alrededor de ellas.
- **stretch**: Las líneas se estiran para ajustarse al contenedor.

Esto puede ser confuso, pero **align-content** determina el espacio entre las líneas, mientras que **align-items** determina como los elementos en su conjunto están alineados dentro del contenedor. Cuando hay solo una línea, **align-content** no tiene efecto.



```
1 #estanque {  
2   pantalla: flexible;  
3   envoltura flexible: envoltura;  
4   /*jhonatan*/align-content: flex-start;  
5 }  
6  
7  
8  
9
```

# RANA FLEXBOX

Nivel 20 de 24

Las dos propiedades `flex-direction` y `flex-wrap` son usadas a menudo en conjunto con la propiedad abreviada `flex-flow`, la cual fue creada para combinarlas. Esta propiedad abreviada, acepta un valor de cada una separada por un espacio.

Por ejemplo, puedes usar `flex-flow` para establecer filas y envolverlas.

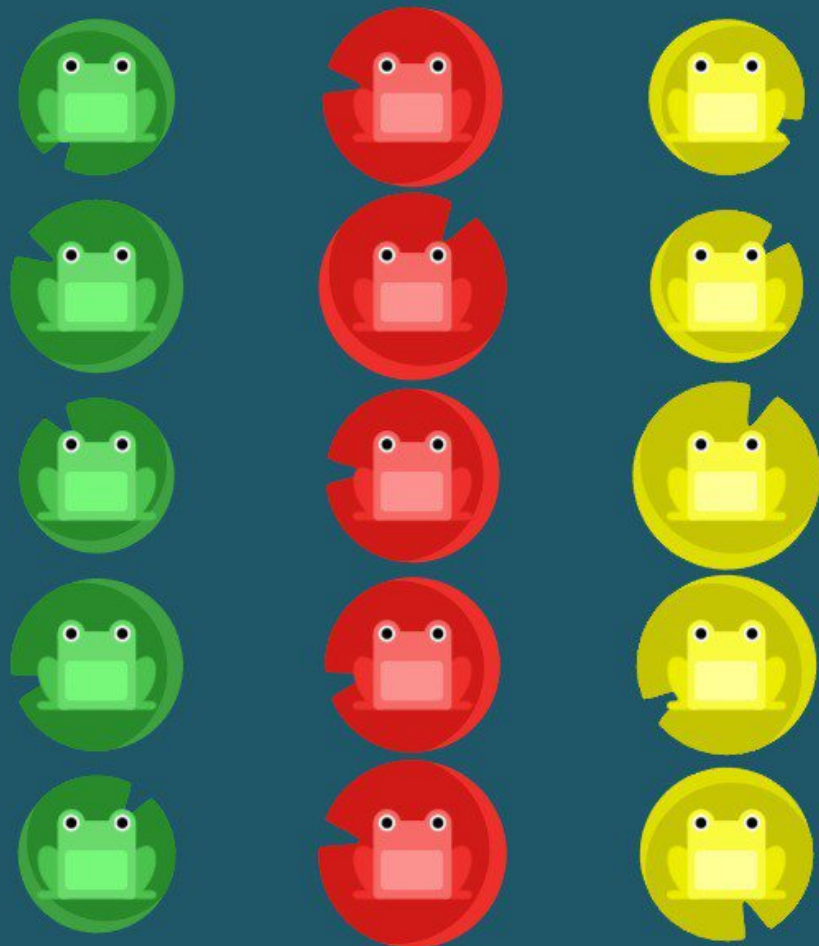
Trata de usar `flex-flow` para volver a realizar el nivel anterior.

```
1 #estanque {  
2   pantalla: flexible;  
3   /*jhonatan*/flex-flow: column wrap;  
4 }  
5  
6  
7  
8  
9  
10
```

Siguiente

Flexbox Froggy es una creación de [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Ajustes](#)

¿Quieres aprender CSS grid? Juega [al Jardín de Cuadrículas](#).





# RANA FLEXBOX

◀ Nivel 19 de 24 ▶

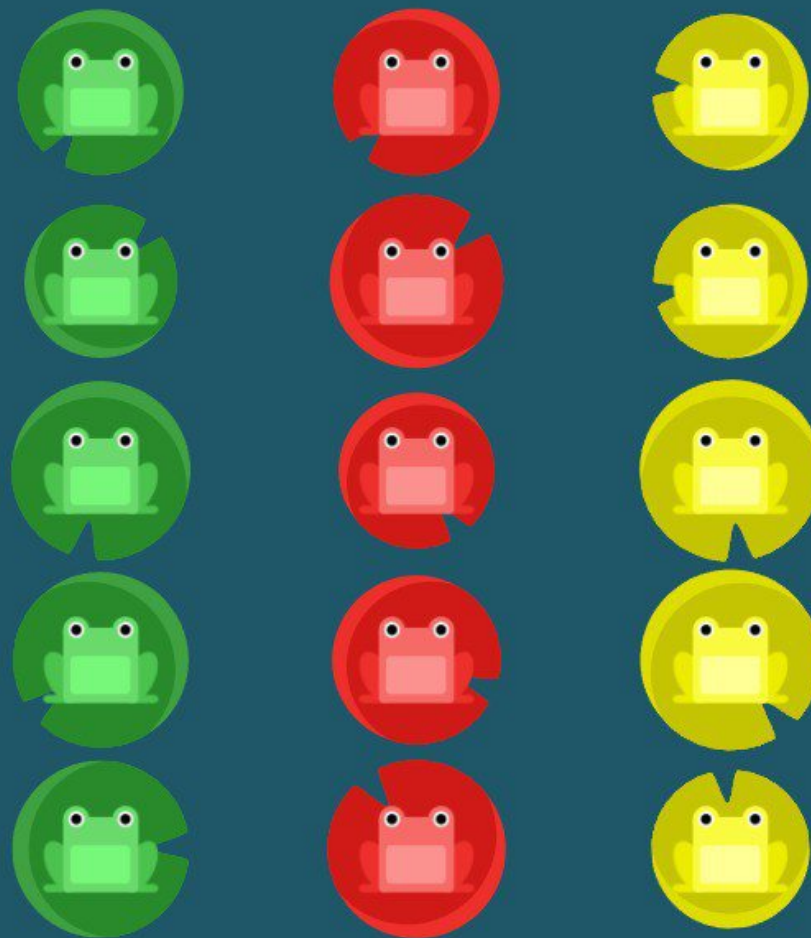
Ayuda a este ejército de ranas a ordenarse en tres columnas usando una combinación de `flex-direction` y `flex-wrap`.

```
1 #estanque {  
2   pantalla: flexible;  
3   /*jhonatan*/flex-direction: column;  
4   flex-wrap: wrap;  
5 }  
6  
7  
8  
9  
10
```

Siguiente

Flexbox Froggy es una creación de [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Ajustes](#)

¿Quieres aprender CSS grid? Juega [al Jardín de Cuadrículas](#).



# RANA FLEXBOX

Nivel 18 de 24

¡Oh, no! Todas las ranas están apretadas en una sola fila de hojas de lirio. Manos a la obra, vamos a distribuirlas correctamente usando la propiedad **flex-wrap**, la cual acepta los siguientes valores:

- **nowrap**: Cada elemento se ajusta en una sola línea.
- **wrap**: los elementos se envuelven alrededor de líneas adicionales.
- **wrap-reverse**: Los elementos se envuelven alrededor de líneas adicionales en reversa.

```
1 #estanque {  
2   pantalla: flexible;  
3   /*jhonatan*/flex-wrap:wrap;  
4 }  
5  
6  
7  
8  
9  
10
```

Siguiente

Flexbox Froggy es una creación de [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Ajustes](#)

¿Quieres aprender CSS grid? Juega al Jardín de Cuadrículas.



# RANA FLEXBOX

Nivel 17 de 24

Combina **order** con **align-self** para ayudar a las ranas a llegar a sus destinos.

```
1 #estaque {
2   pantalla: flexible;
3   alinear elementos: inicio flexible;
4 }
5
6 .amarillo {
7   /*jhonatan*/order:1;
8   align-self:flex-end;
9 }
10
```

Siguiente

Flexbox Froggy es una creación de [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Ajustes](#)

¿Quieres aprender CSS grid? Juega [al Jardín de Cuadrículas](#).





# RANA FLEXBOX

Nivel dieciséis de 24

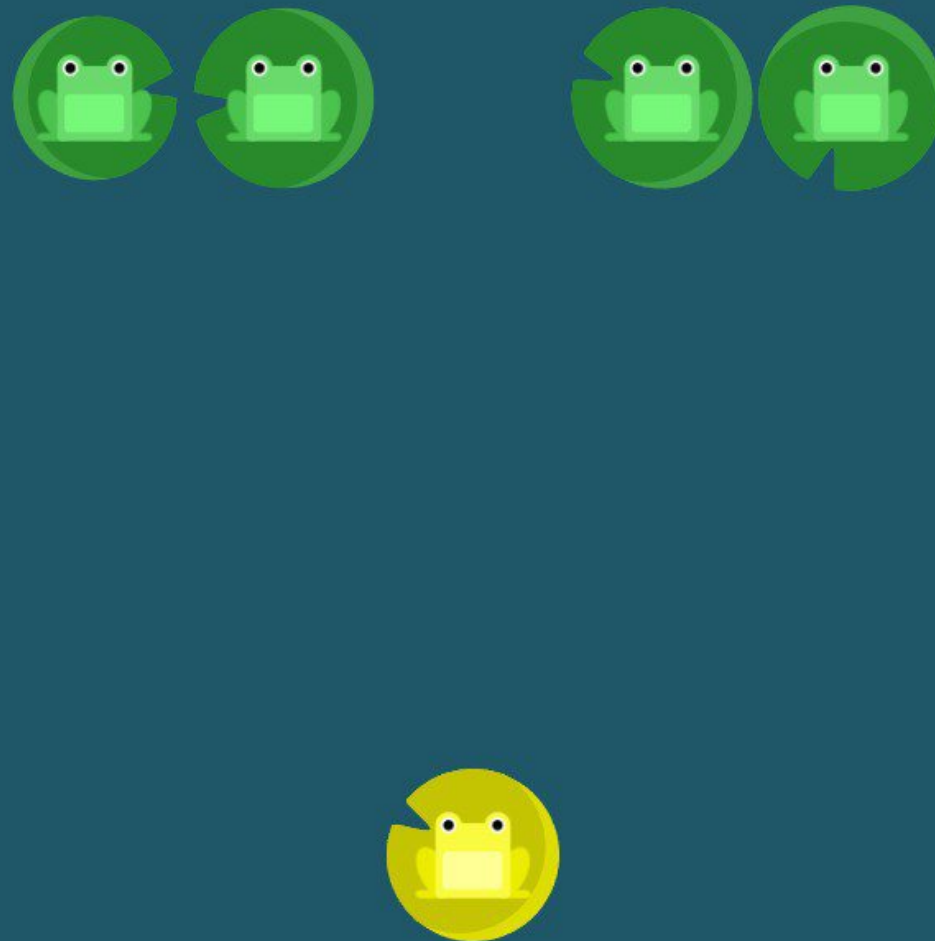
Otra propiedad que puedes aplicar a elementos individuales es **align-self**. Esta propiedad acepta los mismos valores de **align-items** y sus valores son usados para un elemento específico

```
1 #estaque {
2   pantalla: flexible;
3   alinear elementos: inicio flexible;
4 }
5
6 .amarillo {
7   /*jhonatan*/align-self: flex-end;
8 }
9
10
```

Siguiente

Flexbox Froggy es una creación de [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Ajustes](#)

¿Quieres aprender CSS grid? Juega al [Jardín de Cuadrículas](#).





# RANA FLEXBOX

◀ Nivel 15 de 24 ▶

Usa la propiedad `order` para enviar la rana de color rojo a su hoja de lirio.

```
1 #estanque {
2   pantalla: flexible;
3 }
4
5 .rojo {
6   /*jhonatan*/order:-1;
7 }
8
9
10
```

Siguiente

Flexbox Froggy es una creación de [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Ajustes](#)

¿Quieres aprender CSS grid? Juega [al Jardín de Cuadrículas](#).



# RANA FLEXBOX

Nivel 14 de 24

A veces, invertir el orden de una fila o columna de un contenedor no es suficiente. En esos casos, podemos aplicar la propiedad **order** a elementos individuales. Por defecto, los elementos tienen un valor 0, pero podemos usar esta propiedad para establecerlo a un número entero positivo o negativo.

Usa la propiedad **order** para reordenar las ranas de acuerdo a sus hojas de lirio.

```
1 #estanque {
2   pantalla: flexible;
3 }
4
5 .amarillo {
6   /*jhonatan*/order: 1;
7 }
8
9
10
```

Siguiente

Flexbox Froggy es una creación de [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Ajustes](#)

¿Quieres aprender CSS grid? Juega al [Jardín de Cuadrículas](#).



# RANA FLEXBOX

Nivel 13 de 24

Ayuda a las ranas a encontrar sus hojas de lirio usando `flex-direction`, `justify-content`, y `align-items`.

```
1 #estanque {  
2   pantalla: flexible;  
3   /*jhonatan*/flex-direction: row-reverse;  
4   justify-content: center;  
5   align-items: end;  
6 }
```

Siguiente

Flexbox Froggy es una creación de [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Ajustes](#)

¿Quieres aprender CSS grid? Juega [al Jardín de Cuadrículas](#).





# RANA FLEXBOX

Nivel 12 de 24

Ayuda a las ranas a encontrar sus hojas de lirio usando `flex-direction` y `justify-content`.

```
1 #estanque {  
2   pantalla: flexible;  
3   /*jhonatan*/flex-direction:column-reverse;  
4   justify-content: space-between;  
5 }  
6  
7  
8  
9  
10
```

Siguiente

Flexbox Froggy es una creación de [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Ajustes](#)

¿Quieres aprender CSS grid? Juega [al Jardín de Cuadrículas](#).



# RANA FLEXBOX

Nivel 11 de 24

Ayuda a las ranas a encontrar sus hojas de lirio usando `flex-direction` y `justify-content`.

Nota que cuando es una columna, `justify-content` cambia a vertical y `align-items` a horizontal.

```
1 #estaque {
2   pantalla: flexible;
3   /*jhonatan*/flex-direction: column;
4   justify-content: end;
5 }
```

Siguiente

Flexbox Froggy es una creación de [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Ajustes](#)

¿Quieres aprender CSS grid? Juega [al Jardín de Cuadrículas](#).



# RANA FLEXBOX

Nivel 10 de 24

Ayuda a las ranas a llegar a sus hojas de lirio. A pesar de que ellas parecen estar cerca, necesitarás usar `flex-direction` y `justify-content` para llegar hasta allí.

Nota que cuando estableces la dirección a una fila invertida o columna, el inicio y el final también se invierten.

```
1 #estaque {
2   pantalla: flexible;
3   /*jhonatan*/flex-direction: row-reverse;
4   justify-content: start;
5 }
```

Siguiente

Flexbox Froggy es una creación de [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Ajustes](#)

¿Quieres aprender CSS grid? Juega al [Jardín de Cuadrículas](#).





# RANA FLEXBOX

Nivel 9 de 24

Ayuda a las ranas a encontrar su hoja de lirio en la columna usando **flex-direction**. Esta propiedad CSS define la dirección de los elementos en el contenedor, y acepta los siguientes valores:

- **row**: Elementos son colocados en la misma dirección del texto.
- **row-reverse**: Elementos son colocados en la dirección opuesta al texto.
- **column**: Elementos se colocan de arriba hacia abajo.
- **column-reverse**: Elementos se colocando de abajo hacia arriba.

```
1 #estanque {
2   pantalla: flexible;
3   /*jhonatan*/flex-direction: column;
4 }
5
6
7
8
9
10
```

Siguiente

Flexbox Froggy es una creación de [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Ajustes](#)

¿Quieres aprender CSS grid? Juega al Jardín de Cuadrículas.



# RANA FLEXBOX

◀ Nivel 8 de 24 ▶

Las ranas necesitan ponerse en el mismo orden que sus hojas de lirio usando **flex-direction**. Esta propiedad CSS define la dirección de los elementos en el contenedor, y acepta los siguientes valores:

- **row**: Elementos son colocados en la misma dirección del texto.
- **row-reverse**: Elementos son colocados en la dirección opuesta al texto.
- **column**: Elementos se colocan de arriba hacia abajo.
- **column-reverse**: Elementos se colocando de abajo hacia arriba.

```
1 #estanque {  
2   pantalla: flexible;  
3   /*jhonatan*/flex-direction: row-reverse;  
4 }
```

Siguiente

Flexbox Froggy es una creación de [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Ajustes](#)

¿Quieres aprender CSS grid? Juega al Jardín de Cuadrículas.



## RANA FLEXBOX

Nivel 7 de 24

Nuevamente, las ranas necesitan cruzar el estanque. En esta oportunidad, las hojas de lirio tienen mucho espacio alrededor de ellas. Debes usar una combinación de justify-content y align-items.

```
1 #estanque {  
2   pantalla: flexible;  
3   /*jhonatan*/justify-content:space-around;  
4   align-items: flex-end;  
5 }  
6  
7  
8  
9  
10
```

Siguiente

Flexbox Froggy es una creación de [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Ajustes](#)

¿Quieres aprender CSS grid? Juega [al Jardín de Cuadrículas](#).





# RANA FLEXBOX

◀ Nivel 6 de 24 ▶

Mueve la rana al centro del estanque, usando una combinación de `justify-content` y `align-items`.

```
1 #estanque {  
2   pantalla: flexible;  
3   /*jhonatan*/justify-content:center;  
4   align-items: center;  
5 }  
6  
7  
8  
9  
10
```

Siguiente

Flexbox Froggy es una creación de [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Ajustes](#)

¿Quieres aprender CSS grid? Juega [al Jardín de Cuadrículas](#).



# RANA FLEXBOX

Nivel 5 de 24

Ahora usa **align-items** para ayudar a las ranas a llegar al fondo del estanque. Esta propiedad CSS alinea elementos verticalmente y acepta los siguientes valores:

- **flex-start**: Alinee elementos a la parte superior del contenedor.
- **flex-end**: Alinee elementos a la parte inferior del contenedor.
- **center**: Alinea elementos en el centro (verticalmente hablando) del contenedor.
- **baseline**: Muestra elementos en la línea base del contenedor
- **stretch**: Elementos se estiran para ajustarse al contenedor.

```
1 #estanque {  
2   pantalla: flexible;  
3   /*jhonatan*/align-items: flex-end;  
4 }  
5  
6  
7  
8  
9  
10
```

Siguiente

Flexbox Froggy es una creación de [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Ajustes](#)

¿Quieres aprender CSS grid? Juega al Jardín de Cuadrículas.



# RANA FLEXBOX

Nivel 4 de 24

Ahora las hojas de lirio de los bordes se han desplazado a un costado, aumentando así el espacio entre ellas. EE.UU. justify-content. En esta oportunidad, las hojas de lirio tienen el mismo espacio entre ellas.

```
1 #estaque {  
2   pantalla: flexible;  
3   /*jhonatan*/justify-content: space-between;  
4 }
```

Siguiente

Flexbox Froggy es una creación de [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Ajustes](#)

¿Quieres aprender CSS grid? Juega [al Jardín de Cuadrículas](#).





# RANA FLEXBOX

Nivel 3 de 24

Ayuda a las tres ranas a encontrar sus hojas de lirio usando `justify-content`. En esta oportunidad, las hojas de lirio tienen mucho espacio a su alrededor.

Si no recuerdas los valores de una propiedad, puedes pasar el cursor sobre el nombre de una e inmediatamente se mostrarán. Prueba pasando el cursor sobre `justify-content`.

```
1 #estanque {
2   pantalla: flexible;
3   /*jhonatan*/justify-content: space-around;
4 }
```

Siguiente

Flexbox Froggy es una creación de [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Ajustes](#)

¿Quieres aprender CSS grid? Juega [al Jardín de Cuadrículas](#).



# RANA FLEXBOX

Nivel 2 de 24

Nuevamente, utiliza **justify-content** para ayudar a esas ranas a llegar a sus hojas de lirio. Recuerda que esta propiedad CSS alinea elementos horizontalmente y acepta los siguientes valores:

- **flex-start**: Alinee elementos al lado izquierdo del contenedor.
- **flex-end**: Alinee elementos al lado derecho del contenedor.
- **center**: Alinee elementos en el centro del contenedor.
- **space-between**: Muestra elementos con la misma distancia entre ellos.
- **space-around**: Muestra elementos con la misma separación alrededor de ellos.

```
1 #estanque {  
2   pantalla: flexible;  
3   /*jhonatan*/justify-content: center;  
4 }
```

Siguiente

Flexbox Froggy es una creación de [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Ajustes](#)



# RANA FLEXBOX

Nivel 1 de 24

Bienvenido a Flexbox Froggy, un juego donde ayudarás a Froggy y a sus amigos escribiendo código CSS. Guía a esta rana hacia la hoja de lirio en la derecha, usando la propiedad **justify-content**, la cual alinea elementos horizontalmente y acepta los siguientes valores:

- **flex-start**: Alinee elementos al lado izquierdo del contenedor.
- **flex-end**: Alinee elementos al lado derecho del contenedor.
- **center**: Alinee elementos en el centro del contenedor.
- **space-between**: Muestra elementos con la misma distancia entre ellos.
- **space-around**: Muestra elementos con la misma separación alrededor de ellos.

Por ejemplo, **justify-content: flex-end;** moverá la rana a la derecha.

```
1 #estanque {
2   pantalla: flexible;
3   /*jhonatan*/justify-content: flex-end;
4 }
```

Siguiente

