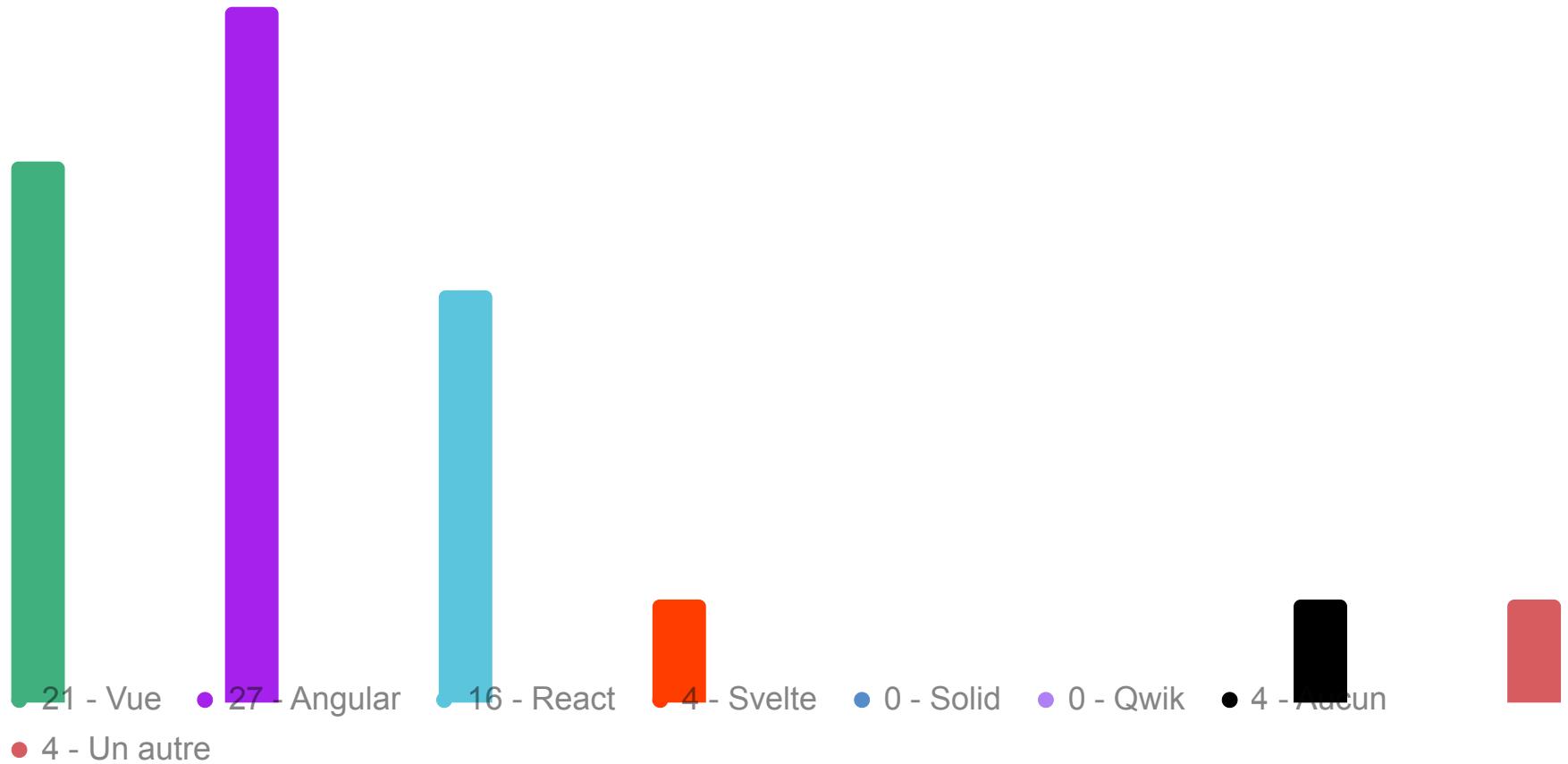


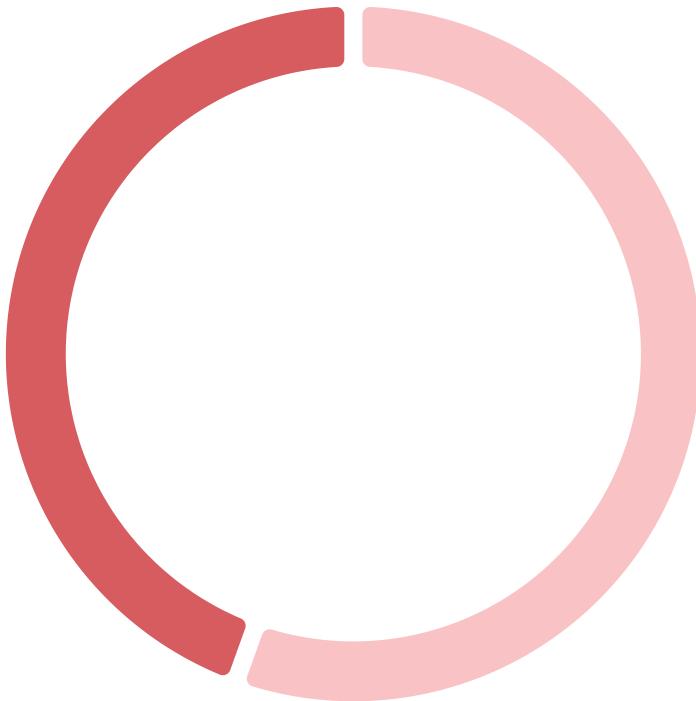
# La réactivité et les signaux : démystifions la magie du frontend



# Quel framework pour le frontend ?



# Qui s'est déjà questionné sur le fonctionnement profond de la réactivité de son framework ?



● 20 - Oui   ● 16 - Non

**ui = fn(state)**

A	B	C
1	Quantité	0
2	Prix unitaire	15
3	Total	0



```
import { computed, effect, signal } from 'alien-signals'      ▷

const quantity = signal(0)
const price = signal(15)

const total = computed(() => quantity() * price())

effect(() => {
  console.log(`Total: ${total()}`)
})

//
```

Total: 0



# Estéban Soubiran

Ingénieur logiciel Avionique chez  Maiaspace



Laravel



Vite



Vue



Nuxt

**The Conference for Ethics and Diversity in Tech:  
MiXiT 2025**

**Beyond the Tech: People Who Made Devoxx France  
2025 Special**

**A Journey to Craft Interactive UI Experiences with  
Dialogs**

**Fusion or the Art of Writing PHP Into a Vue SFC  
Components**



[soubiran.dev](https://soubiran.dev)



@soubiran\_



@soubiran.dev



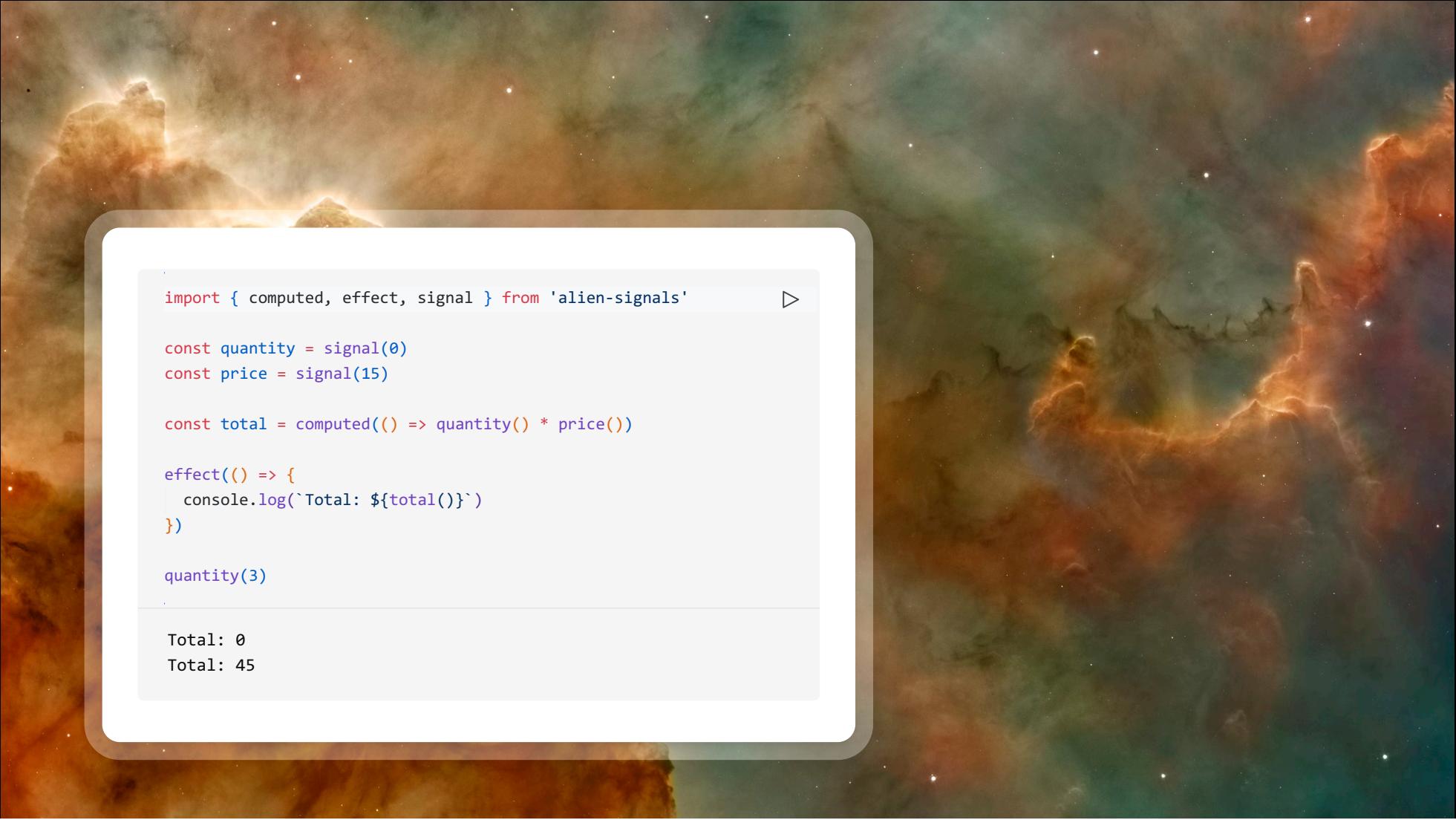
Estéban S



Barbapapazes



Estéban's Open Source



```
import { computed, effect, signal } from 'alien-signals'      ▷

const quantity = signal(0)
const price = signal(15)

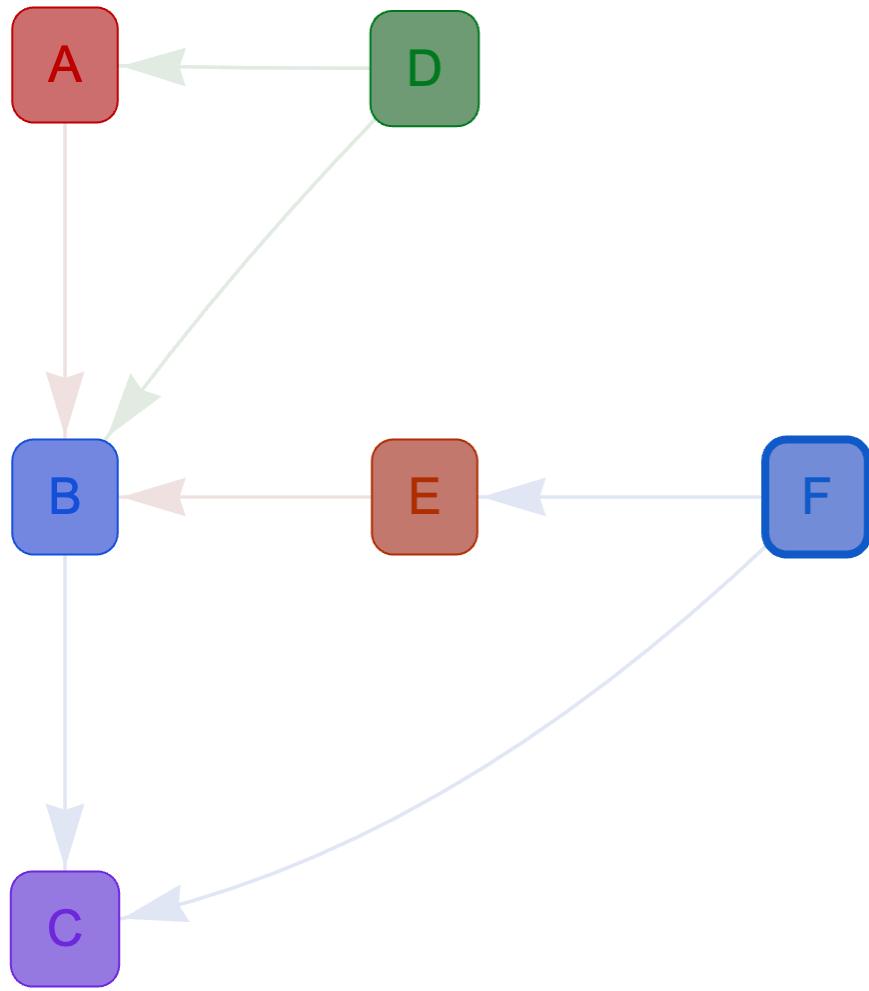
const total = computed(() => quantity() * price())

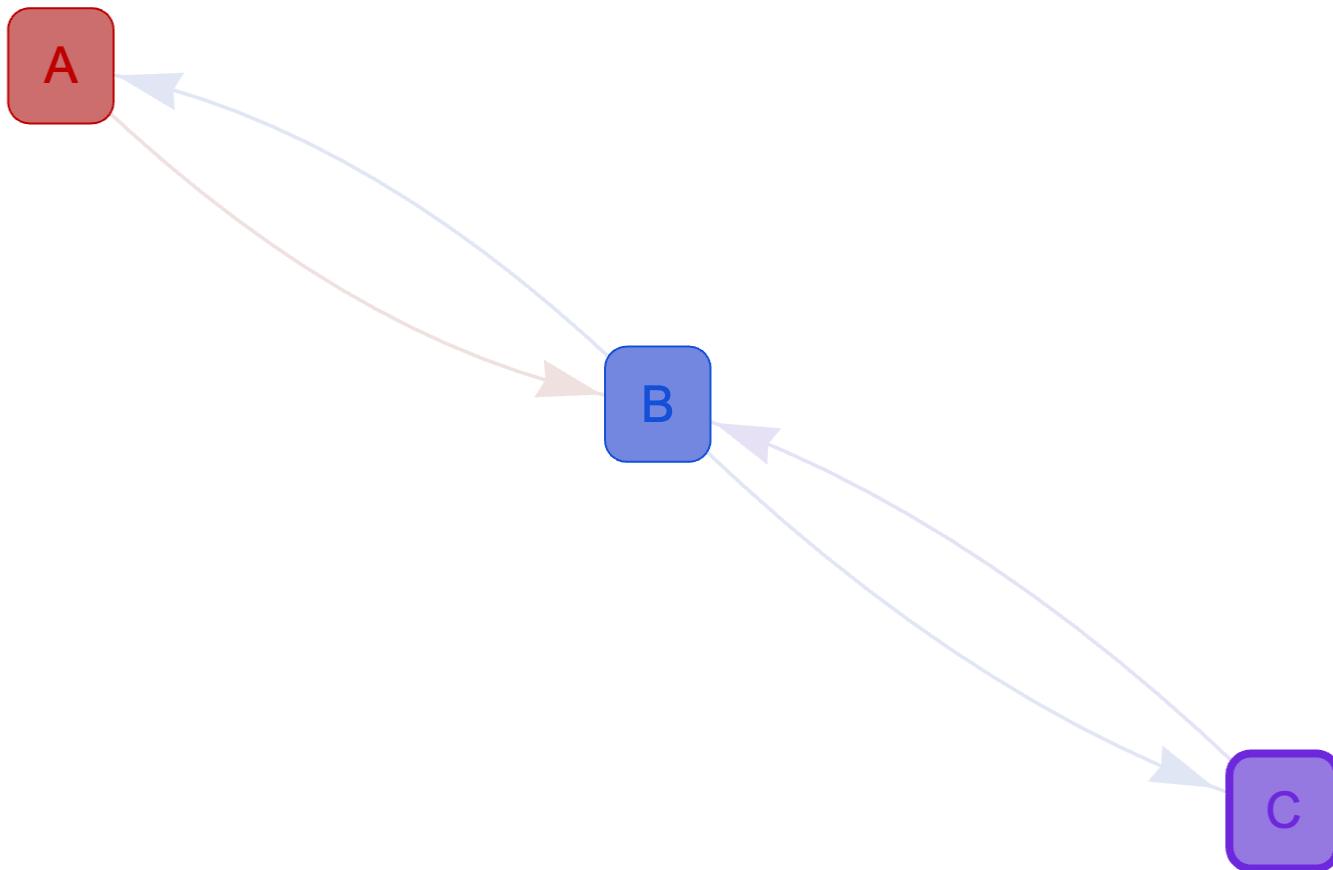
effect(() => {
  console.log(`Total: ${total()}`)
})

quantity(3)

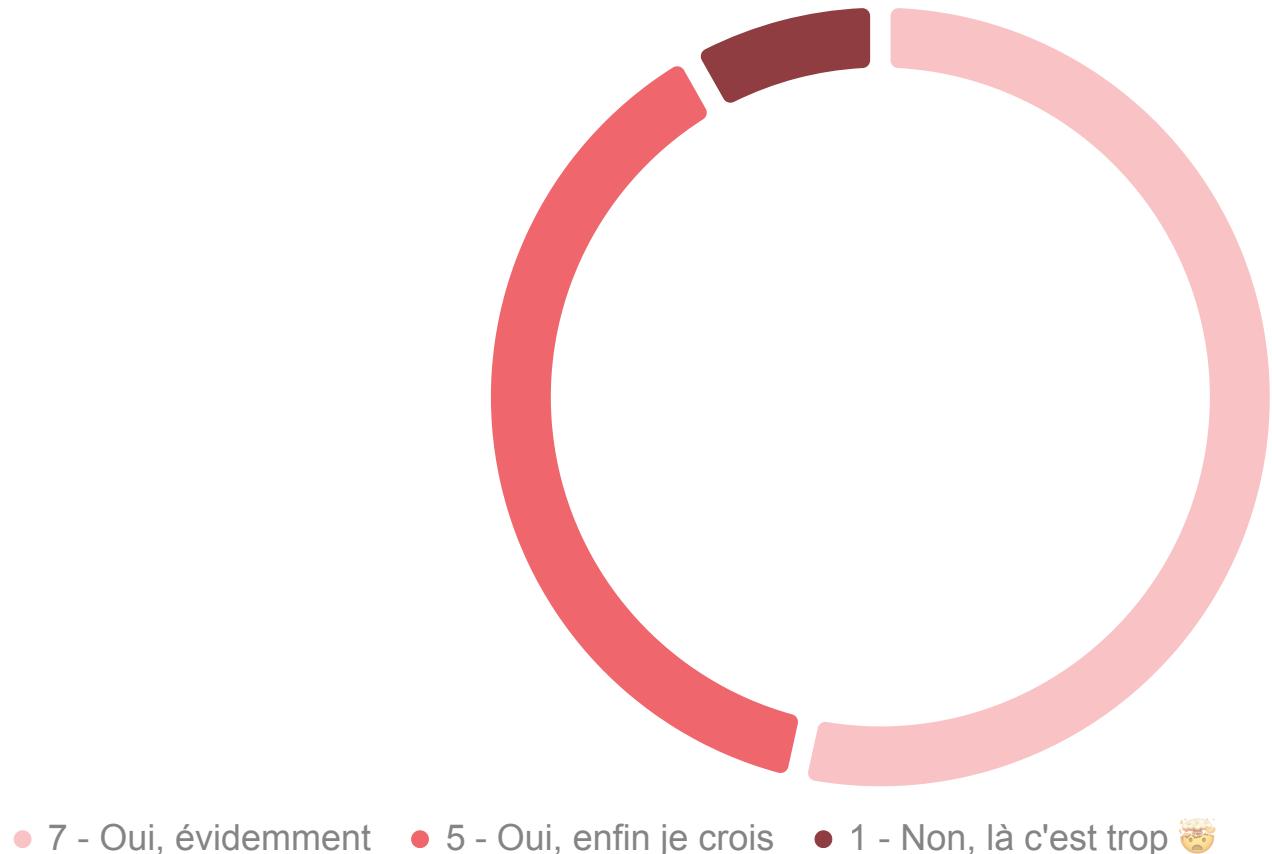

```

Total: 0  
Total: 45

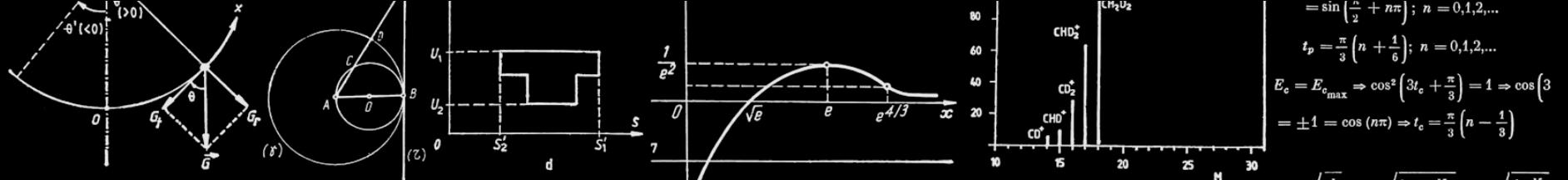




# Vous suivez encore ? 😭



- 7 - Oui, évidemment
- 5 - Oui, enfin je crois
- 1 - Non, là c'est trop 😱



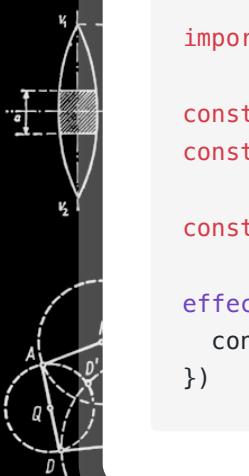
$$\frac{1 - \left(-\frac{1}{n+2}\right)^{n+1}}{1 + \frac{1}{n+2}} + \frac{1}{n+1} \cdot \frac{1 - \left(-\frac{1}{n+1}\right)}{1 + \frac{1}{n+1}} = \int_{-a}^0 x^2 e^{ax} dx = \frac{1}{a} (x^2 e^{ax}) \Big|_{-a}^0 - \frac{2}{a} \int_{-a}^0 x e^{ax} dx \\ -a^2 - \frac{2}{a} \left[ \frac{1}{a} (xe^{ax}) \Big|_{-a}^0 - \frac{1}{a} \int_{-a}^0 e^{ax} dx \right] = -ae^{-a^2} - \frac{2}{a} e^{-a^2} \\ -\frac{1}{n+1}^{n+1} - \frac{1 - \left(-\frac{1}{n+2}\right)^{n+1}}{1 + \frac{1}{n+2}} = + \frac{2}{a^2} \left[ \frac{1}{a} (e^{ax}) \Big|_{-a}^0 \right] = -ae^{-a^2} - \frac{2}{a} e^{-a^2}$$

```
import { computed, effect, signal } from 'alien-signals'

const quantity = signal(0)
const price = signal(15)

const total = computed(() => quantity() * price())

effect(() => {
  console.log(`Total: ${total()}`)
})
```



$$-Q_{41} = vCT_1(1 - \varepsilon^{1/2}) + vC_V T_1(\mathcal{H} - 1), \\ Q_{41} = -vC_V T_1(\mathcal{H} - 1) + vCT_1(1 - \varepsilon^{1/2})$$

$U[V]$	0	0,4	0,6	0,8	0,9	1,0
$I[mA]$	0	-0,4	-0,76	-1,12	-1,5	-1,9

$U[V]$	0	-1	-2	-3	-4	-5
$I[mA]$	0	1,4	2,8	4,2	5,6	7,1

$U[V]$	0	1	2	3	4	5
$I[mA]$	0	1	2	3	4	5

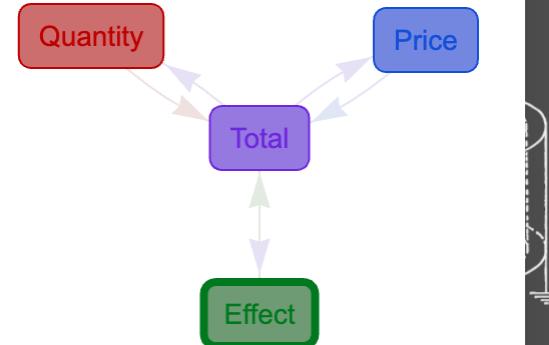
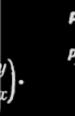
$$y \begin{pmatrix} -t & y \\ t & z-x \end{pmatrix} = \begin{pmatrix} yz - xt & 0 \\ 0 & yz - tx \end{pmatrix} =$$

$$yz - xt)I_0 = -(xt - yz)I_0$$

$$-Q_{41} = vCT_1(1 - \varepsilon^{1/2}) + vC_V T_1(\mathcal{H} - 1),$$

$$-Q_{34} = vC_V T_2(\mathcal{H} - 1) + vCT_4(1 - \varepsilon^{1/2}),$$

$$-T_1 = -T_2 = -T_3 = -T_4 = \sigma$$

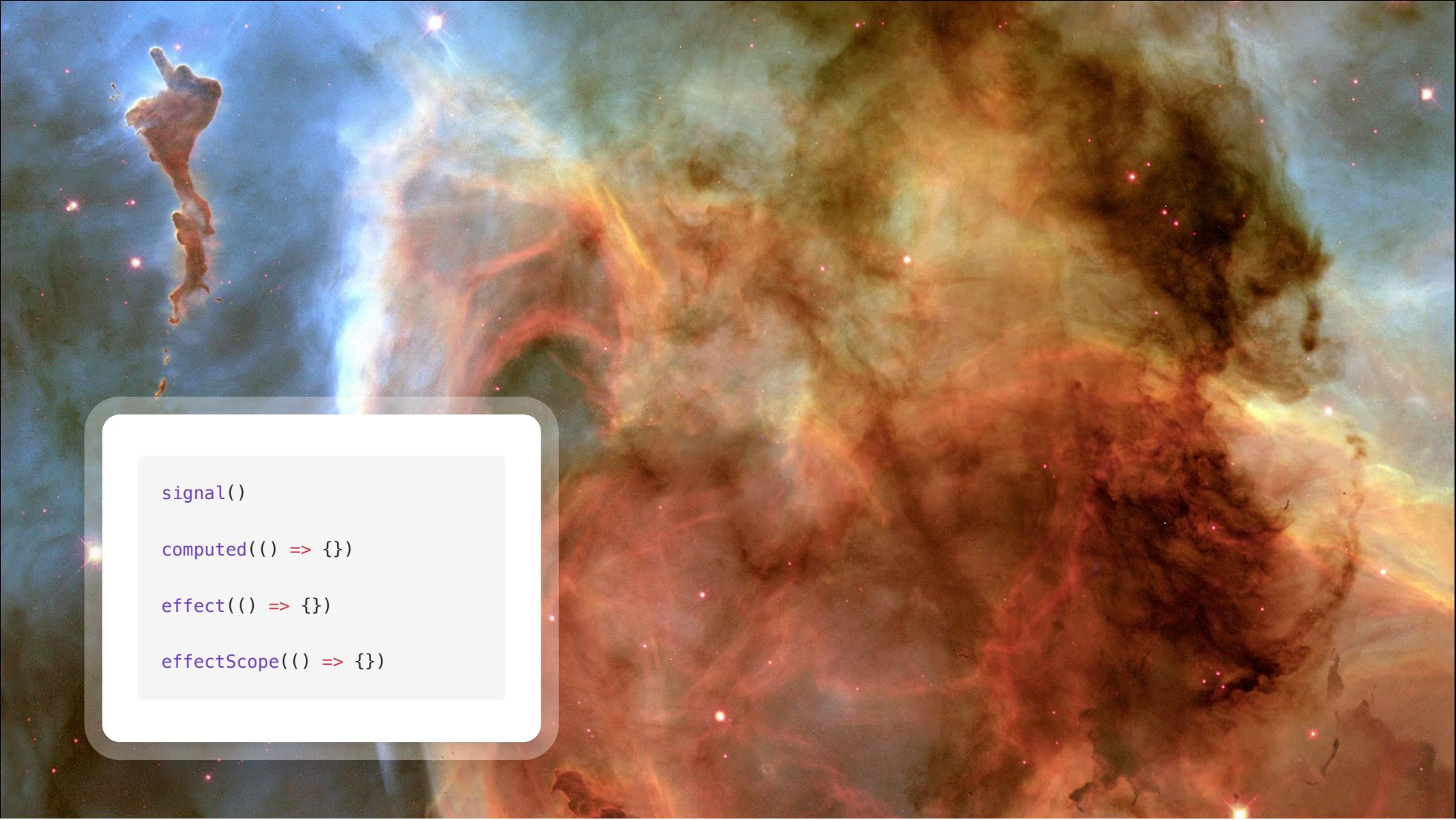


$$= \sin \left( \frac{n}{2} + n\pi \right); n = 0, 1, 2, \dots \\ t_p = \frac{\pi}{3} \left( n + \frac{1}{6} \right); n = 0, 1, 2, \dots \\ E_c = E_{c_{\max}} \Rightarrow \cos^2 \left( 3t_c + \frac{\pi}{3} \right) = 1 \Rightarrow \cos \left( 3t_c + \frac{\pi}{3} \right) = \pm 1 = \cos(n\pi) \Rightarrow t_c = \frac{\pi}{3} \left( n - \frac{1}{3} \right)$$

$$\omega = \sqrt{\frac{k}{m}} = \sqrt{\frac{4\pi m_1 K_p}{3m_1}} = \sqrt{\frac{4\pi K_p}{3}}$$

$$\omega = \sqrt{\frac{g_0}{R_0}},$$

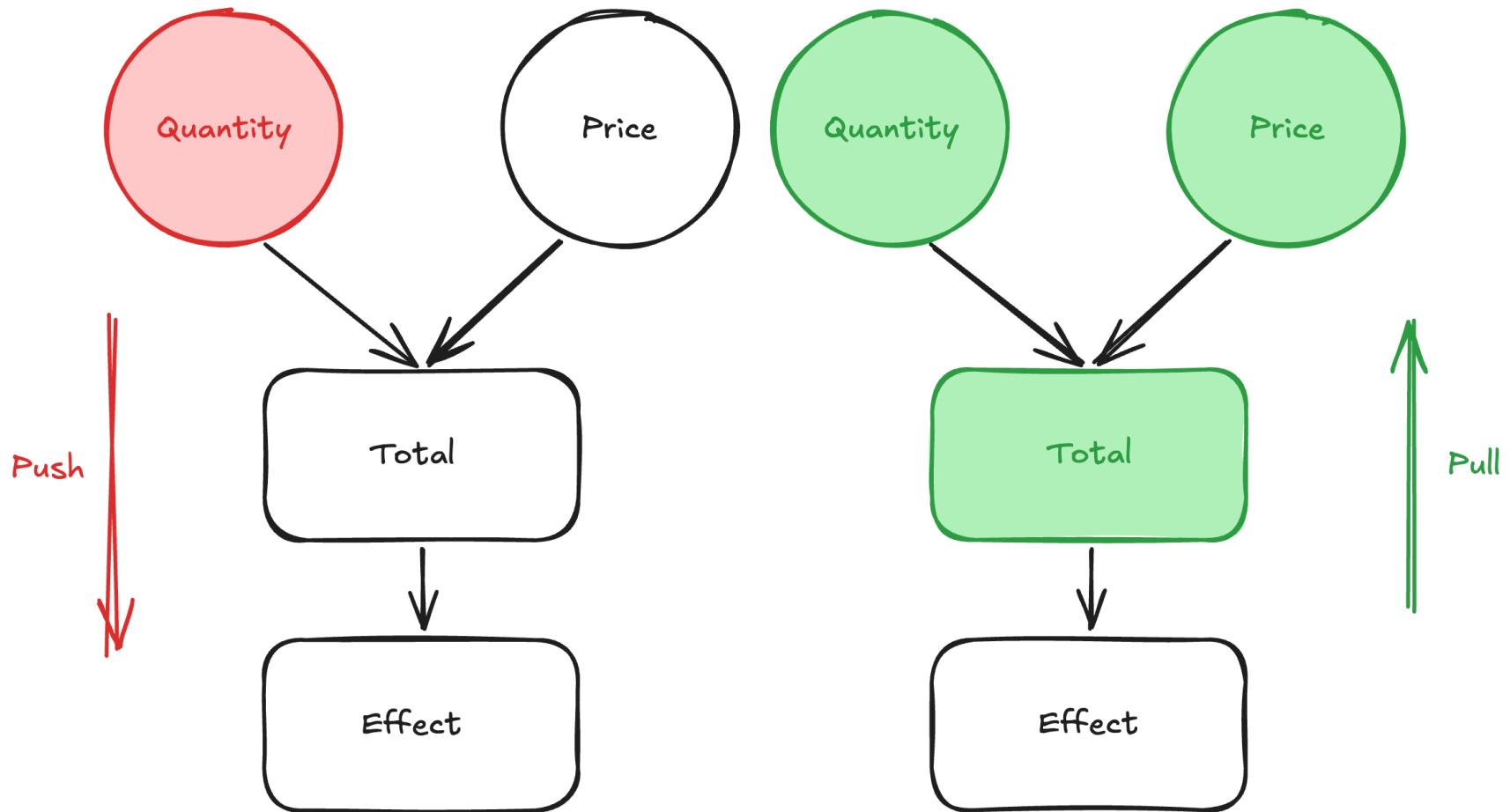
$$T = \frac{2\pi}{\omega} = 2\pi \sqrt{\frac{R_0}{g}} = 5,03 \cdot 10^3 \text{s}.$$

A detailed image of a nebula, likely the Eagle Nebula, showing various gas clouds, dust lanes, and a prominent pillar structure on the left side. The colors range from deep blues and purples to bright reds and yellows.

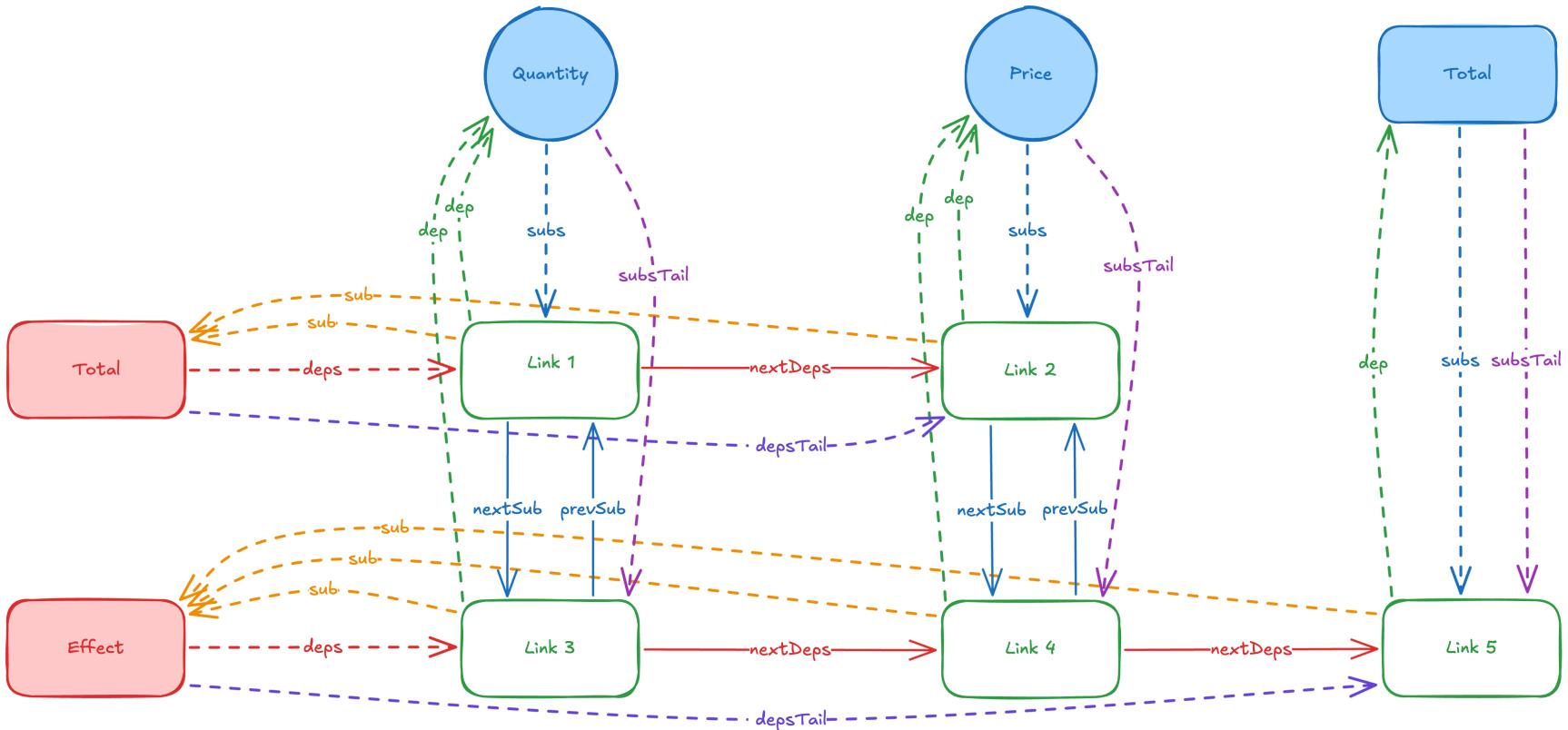
```
signal()  
  
computed(() => {})  
  
effect(() => {})  
  
effectScope(() => {})
```

The background of the slide features a stunning image of a nebula, likely the Horsehead Nebula, with its characteristic dark silhouette and surrounding colorful gas clouds in shades of blue, orange, and yellow. A white rectangular box with rounded corners contains the code.

```
1 export function signal<T>(): WriteableSignal<T | undefined>
2 export function signal<T>(oldValue: T): WriteableSignal<T>
3 export function signal<T>(oldValue?: T): WriteableSignal<T | undefined> {
4   return signalGetterSetter.bind({
5     currentValue: oldValue,
6     subs: undefined,
7     subsTail: undefined,
8   }) as WriteableSignal<T | undefined>
9 }
```



```
1  function signalGetterSetter<T>(this: Signal<T>, ...value: [T]): T | void {
2    if (value.length) {
3      if (this.currentValue !== (this.currentValue = value[0])) {
4        const subs = this.subs
5        if (subs !== undefined) {
6          propagate(subs)
7          if (!batchDepth) {
8            processEffectNotifications()
9          }
10         }
11       }
12     }
13   else {
14     if (activeSub !== undefined) {
15       link(this, activeSub)
16     }
17     return this.currentValue
18   }
19 }
```

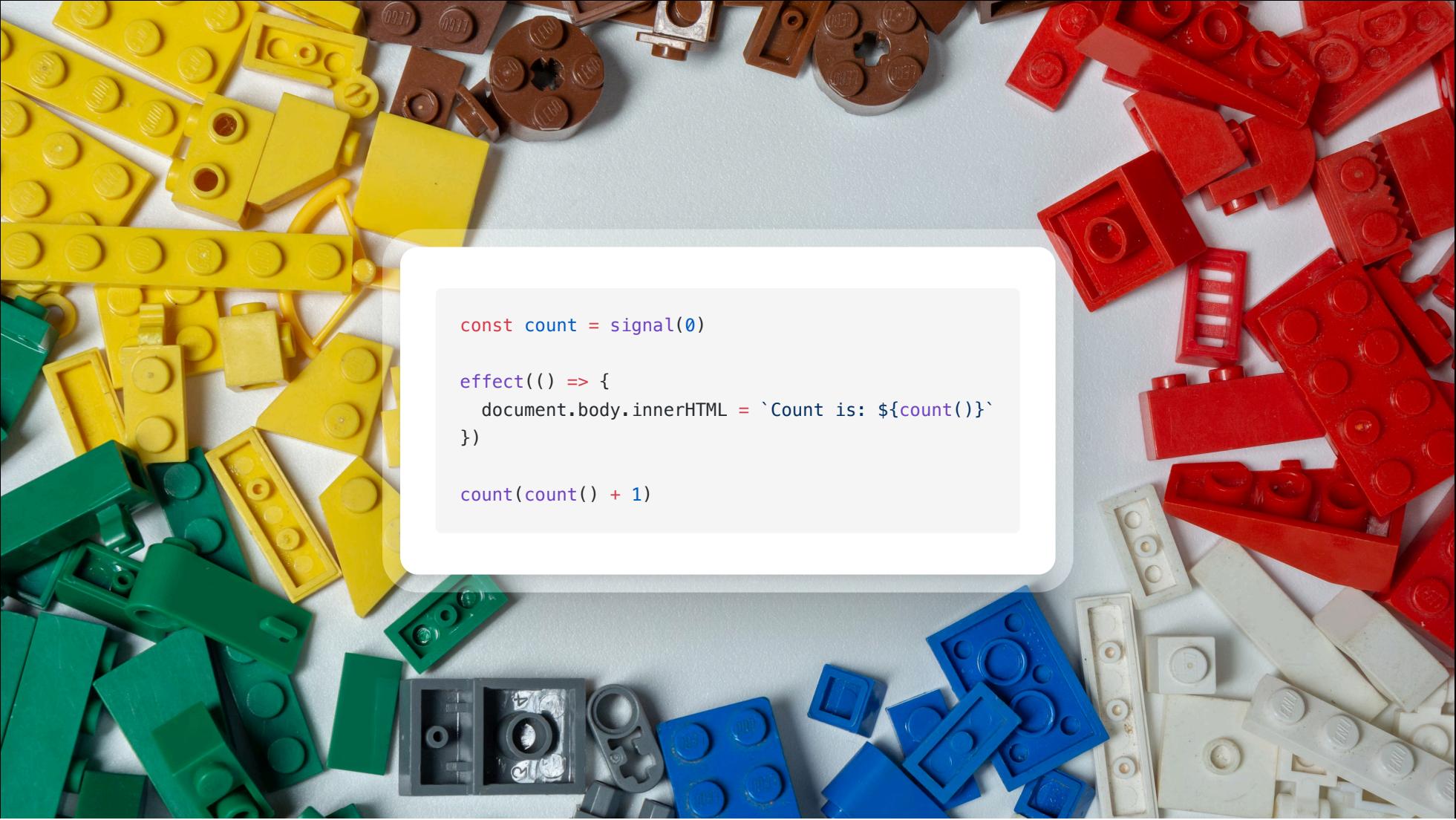


Glitches

Cyclic dependencies

Interaction with mutable state

**Dynamic updating of the graph of dependencies**



```
const count = signal(0)

effect(() => {
  document.body.innerHTML = `Count is: ${count()}`}
})

count(count() + 1)
```

```
instance.scope.on()  
const effect = (instance.effect = new ReactiveEffect(componentUpdateFn))  
instance.scope.off()
```

# Looking for more?

- Explore the [Alien Signals](#) source code
- Watch [Reactivity across frameworks](#)
- Read [Unveiling the Magic: Exploring Reactivity Across Various Frameworks](#)
- Stay curious and keep learning!