

<h1>**Formation web**</h1>

<h2>Partie 4: <em>Svelte</em></h2>

# Déclaratif vs. impératif

Décrire une page plutôt que décrire comment la construire

Déclaratif	Impératif
<ol style="list-style-type: none"><li>1. J'ai une variable</li><li>2. Elle est utilisée là</li><li>3. Elle est changée ici</li></ol>	<ol style="list-style-type: none"><li>1. J'ai une variable</li><li>2. Il faut la mettre là</li><li>3. Quand elle change ici, il faut la mettre à jour là</li></ol>

# Déclaratif vs. impératif

Décrire une page plutôt que décrire comment la construire

Déclaratif	Impératif
<pre>&lt;script&gt; let v = "" &lt;/script&gt;  &lt;p&gt;v vaut {v}&lt;/p&gt; &lt;button   on:click={     () =&gt; { v++ }   } &gt;+&lt;/button&gt;</pre>	<pre>&lt;script&gt; let v = "" document.querySelector('button')   .addEventListener('click', () =&gt; {     v++     document.getElementById('valeur').innerText = v   }) &lt;/script&gt;  &lt;p&gt;v vaut &lt;span id="valeur"&gt;&lt;/span&gt;&lt;/p&gt; &lt;button&gt;+&lt;/button&gt;</pre>

# Déclaratif vs. impératif

Décrire une page plutôt que décrire comment la construire

Déclaratif	Impératif
<pre>&lt;script&gt; let v = "" &lt;/script&gt;  &lt;p&gt;v vaut {v}&lt;/p&gt; &lt;button   on:click={     () =&gt; { v++ }   } &gt;+&lt;/button&gt;</pre>	<pre>&lt;script&gt; let v = "" document.querySelector('button')   .addEventListener('click', () =&gt; {     v++     document.getElementById('valeur').innerText = v   }) &lt;/script&gt;  &lt;p&gt;v vaut &lt;span id="valeur"&gt;&lt;/span&gt;&lt;/p&gt; &lt;button&gt;+&lt;/button&gt;</pre>

# Déclaratif vs. impératif


Décrire une page plutôt que décrire comment la construire

Déclaratif	Impératif
<pre>&lt;script&gt; let v = "" &lt;/script&gt;  &lt;p&gt;v vaut {v}&lt;/p&gt; &lt;button   on:click={     () =&gt; { v++ }   } &gt;+&lt;/button&gt;</pre>	<pre>&lt;script&gt; let v = "" document.querySelector('button')   .addEventListener('click', () =&gt; {     v++     document.getElementById('valeur').innerText = v   }) &lt;/script&gt;  &lt;p&gt;v vaut &lt;span id="valeur"&gt;&lt;/span&gt;&lt;/p&gt; &lt;button&gt;+&lt;/button&gt;</pre>

# Déclaratif vs. impératif

Décrire une page plutôt que décrire comment la construire

Déclaratif	Impératif
<pre>&lt;script&gt; let v = "" &lt;/script&gt;  &lt;p&gt;v vaut {v}&lt;/p&gt; &lt;button   on:click={     () =&gt; { v++ }   } &gt;+&lt;/button&gt;</pre>	<pre>&lt;script&gt; let v = "" document.querySelector('button')   .addEventListener('click', () =&gt; {     document.getElementById('valeur').innerText = v   }) &lt;/script&gt;  &lt;p&gt;v vaut &lt;span id="valeur"&gt;&lt;/span&gt;&lt;/p&gt; &lt;button&gt;+&lt;/button&gt;</pre>



**net7.dev/formaweb-4**

# La réactivité

Des fonctions pour l'HTML

<code>&lt;p&gt;{a}&lt;/p&gt;</code>	Mettre la valeur de a dans mon HTML
<code>\$: f(a, b, c)</code>	exécuter f dès que a, b ou c changent
<code>bind:a={b}</code>	Quand a change, changer b Quand b change, changer a



# **Les composants**

Des fonctions pour l'HTML

# Les composants

Des fonctions pour l'HTML

Définition <i>utils.py</i>	Utilisation <i>main.py</i>
<pre>def f(x):     return x + 5</pre>	<pre>from utils import f  print(f(8))</pre>

# Les composants

Des fonctions pour l'HTML

Définition <i>utils.py</i>	Utilisation <i>main.py</i>
<pre>def f(x):     return x + 5</pre>	<pre><b>from utils import f</b>  print(f(8))</pre>
Définition <i>F.svelte</i>	Utilisation <i>App.svelte</i>
<pre>&lt;script&gt;   export let x &lt;/script&gt;  {x + 5}</pre>	<pre>&lt;script&gt;   <b>import F from './F.svelte'</b>  &lt;/script&gt;  <b>&lt;F x={8} /&gt;</b></pre>

# Les composants

Des fonctions pour l'HTML

Définition <i>F.svelte</i>	Utilisation <i>App.svelte</i>
<pre>&lt;script&gt;   export let x &lt;/script&gt;  {x + 5}</pre>	<pre>&lt;script&gt;   import F from './F.svelte'    let a = 5 &lt;/script&gt;  &lt;F bind:x={a} /&gt;</pre>

# **Des raccourcis**

Parce que on aime le sucre (syntaxique)

# TypeScript dans Svelte

On peut!

# TypeScript dans Svelte

On peut!

```
<script lang="ts">  
  export let value: string;  
  export let important: boolean;  
</script>
```

# TypeScript dans Svelte

On peut!

```
export const theme: Writable<undefined | {  
  name: string;  
  dark: boolean;  
}> = writable(undefined)
```



# Des raccourcis

Parce que on aime le sucre (syntaxique)

<code>&lt;Composant {a} /&gt;</code>	<code>&lt;Composant a={a} /&gt;</code>
<code>&lt;Composant bind:a /&gt;</code>	<code>&lt;Composant bind:a={a} /&gt;</code>
<code>&lt;Composant truc /&gt;</code>	<code>&lt;Composant truc={true} /&gt;</code>
<code>&lt;Composant class:b={c} /&gt;</code>	<code>&lt;Composant class={c ? 'b' : ""} /&gt;</code>
<code>&lt;Composant class:b /&gt;</code>	<code>&lt;Composant class:b={b} /&gt;</code>
<code>&lt;élément style:a={b} /&gt;</code>	<code>&lt;élément style="a: {b};" /&gt;</code>

# Des ressources



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



Churrros™

Development team

**Merci!**