natUIve framework

Element	Code	Options

Grid	`<div class=row>`
	
	    `<div> … </div>`
	    `…`
	
	`</div>`
		• Columns: just a `div`, `.col3`, `.col23`, `.col34` etc.
		• `.contain` constrains to standard page width
		• Columns have equal width automatically. If a column gets explicit width, the others divide the rest equally among themselves.
		• `.row.wrap` wraps columns on multiple lines
		• `.row.half` for 2-column row on mobile
		• `.row.border` for column borders

Mobile/Desktop Content	`<div class=narrow-screen>`
	    `…`
	`</div>`
	
	`<div class=wide-screen>`
	    `…`
	`</div>`	

natUIve framework

Element	Code	Options

Structure	`<div class=header-and-banner>`
	
	`    <header>`
	
	`        …`
	`    </header>`
	
	`    <div class=banner>`
	
	`    </div>`
	
	`</div>`
	
	`<main id=content>`
	
	`    <div class=contain>`
	`            …`
	`    </div>`
	`    …`
	`</main>`
	
	`<footer>`
	
	`    …`
	
	`</footer>`
		• `.header-and-banner.viewport` sets header/banner to viewport height

Modal Window	`<a href=page.html#container class=modal>`
	
	`    Open in modal window`
	
	`</a>`
		• `#container` is recommended to avoid HTML head parts
		• `openFullWindow(HTML)` shows the HTML code in a modal window.

natUIve framework

Element	Code	Options

Slider	`<div class=slider data-autoslide=5>`
	
	    `<div> … </div>`
	   `…`
	
	`</div>`
		• Any element with `.tab` inside slides makes thumbnail navigation instead of numbered one. supported `.toptabs`
		• `data-autoslide=5` slides to next slide after a 5" delay
		• Create sliders dynamically with `makeSlider(el)`
		• `.slider.vertical-thumbnails` option
		• `.slider.vertical` option
		• `.slider.full-window` option
		• `.slider.full-window.headless` option

Tooltip	`<span class=tool>`
	
	    `Hover for a tip`
	    `<span class=tip> Info </span>`
	    `…`
	
	`</span>`	

Accordion	`<div class=accordion>`
	
	    `<label class=trigger for=unique-id>`
	    `<input type=checkbox class=trigger id=unique-id>`
	
	    `<div> … </div>`
	
	`</div>`
		• Embedded accordions supported

Buttons	`<a href=# class=button> Call to Action </a>`	• `.big` option

natUIve framework

Element	Code	Options

Forms	```<form>```
	
	     `<label>`
	        `<span> Description </span>`
	        `<input> // <textarea></textarea>`
	    `</label>`
	
	     `<label>`
	        `<span>Select a file</span>`
	        `<span class=file-input>`
	            `<input type=file>`
	            `<span></span>`
	        `</span>`
	    `</label>`
	
	     `<div class=checkbox>`
	        `<span> Description </span>`
	        `<label>`
	            `<input type=checkbox> <b class=checkbox-icon> Specific description`
	        `</label>`
	        `…`
	
	     `<label>`
	
	         `<span> Select Description </span>`
	        `<span class=selector>`
	            `<select> … </select>`
	        `</span>`
	
	     `</label>`
	  `…`
	    `<input type=submit>`
	
	 `</form>`
		• Description with Tooltip: `<span class=tool>Description <span class=tip> </span>`
		• `<label class=mandatory>`
		• `<form class=flat>`
		• `<input type=number data-digits=4>`

natUIve framework

Element	Code	Options

Lightbox	`<div class=lightbox>`
	
	`    <a href=1.png title=Image 1> Click for Image 1 </a>`
	`    <a href=2.png> <img src=thumbnail.png alt=Thumbnail> </a>`
	`    …`
	
	`</div>`
		When clicked, a `.lightbox` child link to image forms a common slider gallery with its siblings.

Quote	`<p class=quote>`
	`    …`
	`</p>`	

Aspect ratio image container	`<span class=aspect>`
	
	`    <span style='padding-bottom: Ratio %'></span>`
	`    <img src=image.jpg alt=''>`
	
	`</span>`
		`Ratio` = (y / x) * 100, default 16:9, only when container is 100% wide.

Android Browser scroll polyfill	`<div class=overthrow>`
	`    …`
	`</div>`	

Masonry CSS-only	`<div class=masonry>`
	`    <a>Column</a>`
	`    …`
	`</div>`	

Tables		`<td data-sort>` makes it sortable, where the `data-sort` value can be unspecified, **asc** or **desc**.

natUlve framework

Element	Code	Options
Drop-down nav	```<nav class=drop> <form id=level1> </form> <form id=level2> </form> <input type=checkbox class='trigger burger'> <input type=radio class=trigger form=level1 name=level1> <input type=reset form=level1> <a>One <b class=sub> <input type=radio class=trigger form=level2 name=level2> <input type=reset form=level2> <a>Middle One <b class=sub> Bottom One </nav>```	