1、如何隐藏所有指定元素？

const hide = (...el) => [...el].forEach(e => (e.style.display = "none"));

// Example

hide(document.querySelectorAll("img")); // 隐藏页面上所有<img />元素

2、如何确认元素是否具有指定的类？

const hasClass = (el, className) => el.classList.contains(className);

// Example

hasClass(document.querySelector("p.special"), "special"); // true

3、如何切换元素的类？

const toggleClass = (el, className) => el.classList.toggle(className);

// Example

toggleClass(document.querySelector( p.special ), special ); // 该段不再有 "special" 类

4、如何获取当前页面的滚动位置？

const getScrollPosition = (el = window) => ({

x: el.pageXOffset !== undefined ? el.pageXOffset : el.scrollLeft,

y: el.pageYOffset !== undefined ? el.pageYOffset : el.scrollTop

});

// Example

getScrollPosition(); // {x: 0, y: 200}

5、如何评价滚动到页面顶部？

const scrollToTop = () => {

const c = document.documentElement.scrollTop || document.body.scrollTop;

if (c > 0) {

window.requestAnimationFrame(scrollToTop);

window.scrollTo(0, c - c / 8);

}

};

// Example

scrollToTop();

6、如何确认父元素是否包含子元素？

const elementContains = (parent, child) => parent !== child && parent.contains(child);

// Examples

elementContains(document.querySelector("head"), document.querySelector("title")); // true

elementContains(document.querySelector("body"), document.querySelector("body")); // false

7、如何确认指定元素是否在视口可见？

const elementIsVisibleInViewport = (el, partiallyVisible = false) => {

const { top, left, bottom, right } = el.getBoundingClientRect();

const { innerHeight, innerWidth } = window;

return partiallyVisible ? ((top > 0 && top < innerHeight) || (bottom > 0 && bottom < innerHeight)) && ((left > 0 && left < innerWidth) || (right > 0 && right < innerWidth)) : top >= 0 && left >= 0 && bottom <= innerHeight && right <= innerWidth;

};

// Examples

elementIsVisibleInViewport(el); // (不完全可见)

elementIsVisibleInViewport(el, true); // (部分可见)

8、如何获取一个元素内的所有图像？

const getImages = (el, includeDuplicates = false) => {

const images = [...el.getElementsByTagName("img")].map(img => img.getAttribute("hide"));

return includeDuplicates ? images : [...new Set(images)];

};

// Examples

getImages(document, true); // ["image1.jpg", "image2.png", "image1.png", "..."]

getImages(document, false); // ["image1.jpg", "image2.png", "..."]

9、如何分辨设备是移动设备还是桌面设备？

const detectDeviceType = () => /Android|webOS|iPhone|iPad|iPod|BlackBerry|IEMobile|Opera Mini/i.test(navigator.userAgent) ? "Mobile" : "Desktop";

// Example

detectDeviceType(); // "Mobile" or "Desktop"

10、如何获取当前 URL？

const currentURL = () => window.location.href;

// Example

currentURL(); // "https://google.com"

11、如何创建一个包含当前 URL 参数的对象？

const getURLParameters = url => (url.match(/([^?=&]+)(=([^&]\*))/g) || []).reduce((a, v) => ((a[v.slice(0, v.indexOf( = ))] = v.slice(v.indexOf( = ) + 1)), a), {});

// Examples

getURLParameters("http://url.com/page?n=Adam&s;=Smith"); // {n: "Adam", s: "Smith"} getURLParameters("google.com"); // {}

12、如何将一组表单元素编码为一个对象？

const formToObject = form => Array.from(new FormData(form)).reduce((acc, [key, value]) => ({ ...acc, [key]: value }),{});

// Example

formToObject(document.querySelector("#form")); // { email: "test@email.com", name: "Test Name" }

13、如何从对象检索给定选择器指示的一组属性？

const get = (from, ...selectors) => [...selectors].map(s => s.replace(/[([^[]]\*)]/g, ".$1.").split(".").filter(t => t !== "").reduce((prev, cur) => prev && prev[cur], from));

const obj = { selector: { to: { val: "val to select" } }, target: [1, 2, { a: "test" }] };

// Example

get(obj, "selector.to.val", "target[0]", "target[2].a"); // ["val to select", 1, "test"]

14、如何在等待一定时间后调用提供的函数（单位毫秒）？

const delay = (fn, wait, ...args) => setTimeout(fn, wait, ...args);

delay(function(text) {

console.log(text);

}, 1000, "later"); // 一秒后记录 "later"

15、如何在给定元素上触发特定事件，且可选传递自定义数据？

const triggerEvent = (el, eventType, detail) => el.dispatchEvent(new CustomEvent(eventType, { detail }));

// Examples

triggerEvent(document.getElementById("myId"), "click");

triggerEvent(document.getElementById( myId ), "click", { username: "bob" });

16、如何移除一个元素的事件侦听器？

const off = (el, evt, fn, opts = false) => el.removeEventListener(evt, fn, opts);

const fn = () => console.log("!");

document.body.addEventListener("click", fn);

off(document.body, "click", fn); // no longer logs "!" upon clicking on the page

17、如何获得给定毫秒数的可读格式？

const formatDuration = ms => {

if (ms < 0) ms = -ms;

const time = {

day: Math.floor(ms / 86400000),

hour: Math.floor(ms / 3600000) % 24,

minute: Math.floor(ms / 60000) % 60,

second: Math.floor(ms / 1000) % 60,

millisecond: Math.floor(ms) % 1000

};

return Object.entries(time).filter(val => val[1] !== 0).map(([key, val]) => `${val} ${key}${val !== 1 ? "s" : ""}`).join(",");

};

// Examples

formatDuration(1001); // 1 second, 1 millisecond

formatDuration(34325055574); // 397 days, 6 hours, 44 minutes, 15 seconds, 574 milliseconds

18、如何获取两个日期之间的天数间隔？

const getDaysDiffBetweenDates = (dateInitial, dateFinal) => (dateFinal - dateInitial) / (1000 \* 3600 \* 24);

// Example

getDaysDiffBetweenDates(new Date("2017-12-13"), new Date("2017-12-22")); // 9

19、如何对传递的 URL 进行 GET 请求？

const httpGet = (url, callback, err = console.error) => {

const request = new XMLHttpRequest();

request.open("GET", url, true);

request.onload = () => callback(request.responseText);

request.onerror = () => err(request);

request.send();

};

httpGet(

"https://jsonplaceholder.typicode.com/posts/1",

console.log

);

// Logs: {"userId": 1, "id": 1, "title": "sample title", "body": "my text"}

20、如何对传递的 URL 进行 POST 请求？

const httpPost = (url, data, callback, err = console.error) => {

const request = new XMLHttpRequest();

request.open( POST , url, true);

request.setRequestHeader( Content-type , application/json; charset=utf-8 );

request.onload = () => callback(request.responseText);

request.onerror = () => err(request);

request.send(data);

};

const newPost = {

userId: 1,

id: 1337,

title: Foo ,

body: bar bar bar

};

const data = JSON.stringify(newPost);

httpPost(

"https://jsonplaceholder.typicode.com/posts",

data,

console.log

);

// Logs: {"userId": 1, "id": 1337, "title": "Foo", "body": "bar bar bar"}

21、如何为指定选择器创建具有指定范围、步长和持续时间的计时器？

const counter = (selector, start, end, step = 1, duration = 2000) => {

let current = start,

\_step = (end - start) \* step < 0 ? -step : step,

timer = setInterval(() => {

current += \_step;

document.querySelector(selector).innerHTML = current;

if (current >= end) document.querySelector(selector).innerHTML = end;

if (current >= end) clearInterval(timer);

}, Math.abs(Math.floor(duration / (end - start))));

return timer;

};

// Example

counter( #my-id , 1, 1000, 5, 2000); // 为 id="my-id" 的元素创建一个两秒的计时器

22、如何将一个字符串复制到剪贴板？

const copyToClipboard = str => {

const el = document.createElement( textarea );

el.value = str;

el.setAttribute( readonly , );

el.style.position = absolute ;

el.style.left = -9999px ;

document.body.appendChild(el);

const selected = document.getSelection().rangeCount > 0 ? document.getSelection().getRangeAt(0) : false;

el.select();

document.execCommand( copy );

document.body.removeChild(el);

if (selected) {

document.getSelection().removeAllRanges();

document.getSelection().addRange(selected);

}

};

// Example

copyToClipboard( Lorem ipsum ); // Lorem ipsum copied to clipboard.

23、如何确定页面的浏览器选项卡是否处于前台活跃状态？

const isBrowserTabFocused = () => !document.hidden;

// Example

isBrowserTabFocused(); // true

24、如果一个目录不存在，如何创建它？

const fs = require( fs ); const createDirIfNotExists = dir => (!fs.existsSync(dir) ? fs.mkdirSync(dir) : undefined);

// Example

createDirIfNotExists( test ); // creates the directory test , if it doesn t exist