**HTML**

Câu 1: cấu trúc HTML: thẻ doctype(Phần khai báo chuẩn của html hay xhtml.), thẻ head(Phần khai báo ban đầu, khai báo về [meta](https://hocwebchuan.com/reference/tag/tag_meta.php), [title](https://hocwebchuan.com/reference/tag/tag_title.php), [css](https://hocwebchuan.com/tutorial/tut_css.php), javascript…), thẻ body(Phần chứa nội dung của trang web, nơi hiển thị nội dung.),

Câu 2: thẻ meta có công dụng gì? thẻ meta trong HTML được sử dụng để cung cấp metadata về tài liệu HTML. Metadata sẽ không được hiển thị trên trình duyệt, nhưng nó sẽ được bộ máy tìm kiếm phân tích để lấy thông tin cơ bản về trang web của bạn

Câu 3: các thuộc tính thẻ meta và công dụng? [charset](https://viettuts.vn/html/att_meta_charset)(Chỉ định mã hoá ký tự cho tài liệu HTML), [content](https://viettuts.vn/html/att_meta_content)(Cung cấp giá trị được gắn liền với thuộc tính http-equiv hoặc name), name(Chỉ định một tên cho metadata)

Câu 4: trong HTML có 2 loại thẻ chính nào? Inline(các item sẽ nằm trên cùng một dòng, ví dụ như <span> . Nếu các items vượt quá độ dài của dòng thì item sẽ xuống dòng mới và chiếm chiều rộng bằng đúng nội dung , Các inline item sẽ chỉ có thể điều chỉnh margin và padding **left and right** (top và bottom thì không thể)) block(các item có kiểu display: block luôn được xuống dòng và chiếm toàn bộ width nếu width không được set)

Câu 5: đường dẫn tuyệt đối và tương đối? Đường dẫn tuyệt đối (là một chuỗi đầy đủ bao gổm *http://*, *tên miền của trang web*, *đường dẫn đến tập tin*) Đường dẫn tương đối (là một phần nhỏ của đường dẫn tuyệt đối, thông thường đường dẫn tương đối là phần *đường dẫn đến tập tin*)

Câu 6: trong thẻ form có gì? 2 thuộc tính action(action dùng để xác định địa chỉ trang web mà những dữ liệu được nhập vào được gửi đến và xử lý) và method(method dùng để xác định [cách thức dữ liệu được đẩy lên](https://freehost.page/cac-phuong-thuc-http-get-va-post/) có 2 loại là get và post)

Câu 7: action để rỗng thì sao? Lấy đường dẫn mình đang vào

Câu 8: get và post? Ưu: get thì truy xuất nhanh hơn , post thì bảo mật tốt hơn. Dữ liệu đươc gửi bởi phương thức POST rất bảo mật vì dữ liệu được gửi ngầm, không đưa lên URL, nếu cần bảo mật thông tin thì sử dụng post.

Câu 9: semantic là gì? Phần tử Semantic trong HTML là mô tả rõ ràng ý nghĩa về cấu trúc của phần tử đó đối với các trình duyệt và lập trình viên. ví dụ <section><header><footer>...

Câu 10: cấu trúc thẻ table? Tag <table> được dùng để tạo một bảng HTML ; tag <table> đơn giản có chứa một hoặc nhiều [<tr>](https://hocwebchuan.com/reference/tag/html5/tag_tr.php)( xác định hàng của table.),  [<th>](https://hocwebchuan.com/reference/tag/html5/tag_th.php)( xác định phần tử tiêu đề của table.) và [<td>](https://hocwebchuan.com/reference/tag/html5/tag_td.php)(xác định phần tử nội dung của table)

**CSS**

Câu 1: catcadin trong css là gì và 3 yếu tố của nó? Catcadin: thuật toán xác định các phần tử thuộc tínhimported **>** Specificity **>** SourceOrder

Câu 2: các css selecter trong css? 5 simple(sử dụng chọn element / class / id.), **combination(**cho phép bạn chọn thành phần rất cụ thể bằng nhiều tham chiếu.**)**, attribute(sử dụng để chọn thuộc tính HTML) , pesuado element, pesuado class

Câu 3: phân biệt các div + ~ > \* p? (>) chỉ chọn phần tử con trực tiếp, (~)tất cả thẻ p sau đó cùng cấp, (\*) lấy all, (+)chỉ chọn các phần tử mà nằm ngay sau phần tử trước đó

Câu 4: box model: 4 thành phần : content > padding > border> marrgin

Câu 5: xử lý marrgin trồng lên nhau? Marrgin âm

Câu 6: box-sizzing : border box? thuộc tính width và height sẽ tự động bao gồm luôn content, padding và border, còn chiều rộng của content sẽ tự động co giãn tương ứng nếu chúng ta thay đổi padding và border, đảm bảo kích thước của cả element sẽ không thay đổi, đồng nghĩa với việc layout tổng sẽ không thay đổi.

Câu 7: over flow xảy ra khi nào? Cách xử lý? Nội dung thẻ con vượt quá thẻ cha, cách khắc chế over-flow: scroll, hidden, auto , visibal,...

Câu 8: responsive ? cú pháp ? là sự tương thích màn hình các thiết bị, các màn hình phổ biến (tự tra không rảnh ghi).

Câu 9: trươc khi bắt tay vào responsive thì phải để ý làm gì đầu tiên? Phải để ý hàng cột, kich thước cố định đã fix bằng px, navbar, button, section...v.v chỉnh sử font size.

Câu 10: tại sao lại mobile first? Xu hướng người xử dụng smart phone lướt web

Câu 10: cách nào chia layout? table float , flex box, grid

Câu 11: phân biệt flex gird box? Flex 1 chiều , grid 2 chiều , CSS Grid tiếp cận theo hướng nội dung còn Flexbox tiếp cận theo hướng bố cục (layout).

Câu 12: sử dụng phần tử thuộc tính clear both ? Thuộc tính clear xác định 2 bên của phần tử (left, right), nơi mà phần tử [float](https://hocwebchuan.com/reference/cssSection/pr_float.php) không được cho phép (ngăn cản thành phần không được float trái, phải hay cả hai).

Câu 13: flex grow và flex shrink? Flex grow sử dụng khoảng trống còn lại trong thẻ div , flex shrink co lại khi chúng ta thay dổi with của cái màn hình, <https://codepen.io/enxaneta/full/adLPwv/>

Câu 14: inline-block? Giống inline nhưng có thể set 4 chiều

Cuâ 15: thuộc tính align-item dùng cho trục nào ? <https://codepen.io/enxaneta/full/adLPwv/>

Câu 16: thuộc tính visibility? Thuộc tính visibility xác định thành phần có được nhìn thấy hay không.

Câu 17: phân biệt actribule và properti? Actribule: hãy hiểu theo kiểu key và value(), properti: mô tả các thuộc tính của object(cả css và DOM) <https://ngocminhtran.com/2017/10/21/lam-viec-voi-properties-attributes-va-du-lieu/>

Câu 18: z-index? z-index càng cao thì element đó đứng trước và hiện lên trên, z-index chỉ có tác dụng với các element khi bạn chỉ định thêm 3 giá trị absolute, fixed hay relative

Câu 19 : before và affter? Nâng cao trải nhiệm ng dùng, điều kiện bắt buộc là phải có nội dung, ví dụ tự tìm cho giác ngộ nhé ☺))))

Câu 20: sự khác nhau giữa position sticky và position fix? position fix: cố định trên HTML , position sticky: fix theo cha, position mặc định là static <https://www.w3schools.com/howto/tryit.asp?filename=tryhow_css_sticky_element> ; <https://viblo.asia/p/tim-hieu-thuoc-tinh-position-sticky-trong-css-6J3ZgwrWZmB>

Câu 21: relative và absolute? **Relative**: Định vị trí tuyệt đối cho các thành phần, không gây ảnh hưởng tới vị trí ban đầu **hay** các thành phần khác. **Absolute**: Giá trị này sẽ giúp định vị trí tuyệt đối cho thành phần theo thành phần bao ngoài, hoặc ít nhất **là** theo cửa sổ trình duyệt. **Fixed**: Định vị trí tương đối cho thành phần theo cửa sổ trình duyệt.

**JAVASCRIP**

Câu 1: hàm nào convert string thành arr và ngược lại? phương thức split(), phương thức from(), toán tử 3 chấm và vòng lặp

Câu 2: hàm fillter? lặp qua qua các phần tử, sau đó tùy vào từng bài toán mà sẽ quyết định chọn phần tử đó hay không. Cuối cùng hàm này sẽ trả về một mảng các phần tử đã chọn

Câu 3: tham trị và tham chiếu? **Tham trị là:** hàm được gọi bằng cách truyền trực tiếp giá trị của biến làm đối số. Thay đổi đối số bên trong hàm không ảnh hưởng đến biến được truyền từ bên ngoài hàm. Javascript luôn luôn truyền tham trị và không bao giờ thay đổi giá trị với kiểu dữ liệu nguyên thủy (String, number hoặc boolen) , **Tham chiếu là**: hàm được gọi bằng cách chuyển trực tiếp tham chiếu / địa chỉ của biến làm đối số. Thay đổi đối số bên trong hàm ảnh hưởng đến biến được truyền từ bên ngoài hàm.

Câu 4: var a = 5;

var b = a; a = 6; console.log(b); 🡺 b = 5;

câu 5: var info = ["Phuc", 22]; var newInfo = info; newInfo [1] = 25; console.log(info); 🡺 info = phúc và 25;

câu 6: phân biệt giữa var let const ? var: local scope, có tính hoisting, let: không cho khai báo lại nhưng cho gán lại, const: không thay đổi cũng như gán lại.

Câu 7: các method của array? **Sort**(sắp xếp các phần tử tăng hay giảm return array), **concat**, **pop**(có chức năng xóa bỏ phần tử cuối cùng của mảng, hàm sẽ trả về phần tử bị xóa), **push**(sẽ thêm mới một hoặc nhiều phần tử vào cuối mảng, hàm trả về chiều dài của mảng mới), **shift**(có chức năng loại bỏ phần tử đầu tiên của mảng. Hàm sẽ trả về phần tử đó), **unshift**(dùng để thêm một hoặc nhiều phần tử vào vị trí đầu tiên của mảng), **splice**(sẽ thay thế một hoặc một số phần tử của mảng bằng một hoặc một số phần tử khác), **slice**(trả về mảng mới mảng cũ không ảnh hưởng)

Câu 8: loop array? For, do while, while, forin(dùng lấy vị trí hoặc object), forof(dùng lấy giá trị),

Câu 9: phân biệt foreach map? Foreach: bỏ qua các giá trị trả về, map: cấp phát bộ nhớ và lưu trữ các giá trị trả về

Câu 10: find của array trả về cái gì ? trả về phàn tử đầu tiên

Câu 11: fillter ? tạo ra mảng mới với phần tử thỏa mãn điều kiện call back, không nhất thiết phần tử bằng với mảng cũ

Câu 12: so sánh 2= và 3= ? 2=: so sanh giá trị , 3=: so sánh cả kiểu dữ liệu nữa

Câu 13: ++i và i++? Một cái trả về i trước khi thực hiện lệnh , cái kia ngược lại

Câu 14: cú pháp template string dùng để làm gì? Dùng để cộng chuỗi, có thể viết thêm biến trong template string

Câu 15: có các cách viết fiunction nào? **cách 1:** Function không có tham số và không trả về bất cứ giá trị gì, **cách 2:** arrow function, **cách 3:** anonymous function, **cách 4:** function có nhiều đối số

Câu 16: function đối số và tham số? Ví dụ cần nhận giá trị để thực hiện logic thì là function có tham số,

Câu 17: các cách truy cập element sử dụng DOM? getElementById, getElementsByTagName, getElementByClassName, querySelcter,

Câu 18: var a = docomnet.querySelector(a[href^="flower"]) <https://www.w3schools.com/cssref/css_selectors.asp>

Câu 19: các cách lắng nghe sự kiện ? 3 cách :inline dùng onclick=”function()”, addEventListener, docomnet.querySelector().onclick()

Câu 20: sự kiện nổi bọt và cách ngăn chặn nó? Khi sự kiện diễn ra sẽ bắt đến nơi xảy ra sự kiện, sau đó thực hiện sự kiện sau đó truyền ra ngoài , đến khi nào không có thẻ cha nào có sự kiện đó nữa thì thôi. Cách shutdown n: event(object) có method stopProparaytionEvent, cách thứ 2: truyền tham số thứ 3 vào addlistener là true

Câu 21: mỗi 1 sự kiện xảy ra có 1 hàm sử lý sự kiện trong hàm này có 1 tham số tự dộng truyền vào là e(gọi là event) ,e này gọi là event object.

Câu 22: e.eventTarget: trả về cái element đấy

Câu 23: có thẻ cha trong đó có nhiều thẻ con , mỗi thẻ con đều có sự kiện onclik riêng , mình sẽ sử dụng event target này sử dụng cho thằng con nào r sử lý sự kiện đó.

Câu 24: trong DOM có những dự liệu nào datatype: docoment, nodeNeed, node, element (trong node có textNode, attribuleNode, nodeElement)

Câu 25: call stack: LIFO (last in first out): , callback Queue: FIFO(first in first out)

Câu 26: cách nhúng js ? internal, inline , external