**BOOTCAMP PREPARATION 2.0**

|  |  |  |
| --- | --- | --- |
| **STT** | **Câu hỏi** | |
| 1 | | Repository là gì? có bao nhiêu loại repository?   * Là kho nơi chứa lưu trữ toàn bộ mã nguồn - bao gồm toàn bộ các file và lịch sử lưu trữ các file đó – chứa tấ cả các commit * Có 2 loại repository: * Local repo:kho lưu trữ cục bộ tại máy tính LTV * Remote repo:kho lưu trữ tại 1 máy chủ hay sever * Snapshot :lát cắt thời gian lưu trữ toàn bộ mã nguồn(hình ảnh) tại 1 thời điểm.Có thể quyết định tạo hay quay lại bất kỳ lúc nào. * Commit:là cách tạo ra snapshot,được tạp ra khi có nhiều thay đổi đáng kể đối vs mã nguồn như: tạo 1 tính năng mới – sữa lỗi – cải tiến mã nguồn | |
| 2 | | Các câu lệnh căn bản để làm việc với git?   * Git clone: sao chép 1 repo; * Git innit: khởi tạo 1 repo; * Git add: đưa các file vào vùng stage; * Git commit: ghi nhận các thay đổi, tạo ra các snapshot; * Git push: đưa các thay đổi từ local repo lên remote repo; * Git push-f:xử lý lỗi trả lại tất cả về ban đầu; | |
| 3 | | Giải thuật là gì? các cách để biểu diễn thuật toán?  Thuật toán/giải thuật (Algorithm) là tập hợp hữu hạn các bước (chỉ thị) để giải quyết 1 vấn đề (bài toán);  Có 2 cách biễu diễn thuật toán thông dụng: mã giã (pseudo-code) và lưu đồ (flowchart) | |
| 4 | | Mô tả thuật toán tìm kiếm 1 phần tử trong mảng bằng mã giả?  Begin-> Input N dãy số a1,a2,….aN. i=0 number=?  While(i<=N)  Do  If a[i]=number then number có trong mảng  End if  Else numer không có trong mảng  i++  End Do  Display ->End | |
| 5 | | Mô tả thuật toán sắp xếp mảng số nguyên bằng mã giả?  Begin  Input N dãy a1,a2….an i=0  For ( i = 0; i < num.length; i++)  { for (let j = 0; j < num.length - i; j++)  { if (num[j] <= num[j + 1])  { let b = num[j];  num[j] = num[j + 1];  num[j + 1] = b;  } } }  Display dãy  End | |
| 6 | | Trình bày ý tưởng của 1 thuật toán bất kì?  Begin -> Input (a1,a2,….aN),N,max=a1,i=2  While (i<=n) D0  If (a1>max) then max=ai  End if  I=i+1  End Do -> Output | |
| 7 | | Nêu các thẻ để tạo danh sách ? <ul><li> & <ol><li>  Sự khác nhau giữa <ul> và <ol> ?  <ul> hiển thị dấu chấm tròn đầu dòng <ol>hiển thị theo số thứ tự đầu dòng | |
| 8 | | Các thẻ tạo nên 1 table? Phân biệt th và td?   * Các thẻ tạo bảng: * <table></table> tạo danh sách hiển thị dữ liệu theo cột và hàng. * <tr></tr>: tạo hàng. * <th></th>:Tạo ô tiêu đề.Mặc định in đậm và nằm center ô * <td></td>: Tạo ô dự liệu.Mặc định in thường và nằm left ô | |
| 9 | | Phân biệt innerHTML và innerText?  -Đối với các thẻ hiển thị văn bản như div, span, h, p thì sử dụng innerText or innerHTML để hiển thị nội dung.Còn đối với các thẻ input thì sử dụng .value để hiển thị   |  |  | | --- | --- | | Inne.Text hiểu tất cả như một nội dung văn bản thuần túy.  Không thể chèn các thẻ HTML + CSS  Nó trả về văn bản mà không có thẻ phần tử bên trong. | Inner.HTML hiểu các thẻ HTML và có thể hiển thị như 1 file HTML.  Có thể CSS,chèn các thẻ HTML vào nội dung hiển thị  Nó trả về một thẻ có thẻ phần tử bên trong. | | |
| 10 | | Nêu một số thẻ HTML cơ bản mà bạn biết   * <p> Hiển thị văn bản\_block(chiếm hết 1 dòng) * <span> Hiển thị văn bản\_inline (chiếm đúng chiều rộng nội dung) * <h1>….<h6> Tiêu đề kích thước lớn nhất đến nhỏ nhất * <br>: xuống hàng <hr>:dùng tạo đường gạch ngang khi xuống dòng * <img> hiển thị hình ảnh   <img scr=”link dẫn đến hình ảnh” alt=”chú thích cho hình ảnh”   * <a href=”chứa link liên kết” target=” “> target=”blank” hiển thị link ở tag khác, target=”…” hiển thị đè lên chính trang hiện tại | |
| 11 | | Các thuộc tính cơ bản của thẻ form:  Form là biểu mẫu chứa các ô dữ liệu (ô input) được sinh ra vs mục đích tổng hợp thông tin người dùng nhập vào để gởi đến sever.  Thuộc tính cơ bản của form:   * Action:chứa giá trị URL của sever nơi gởi dữ liệu * Name :tên form * Method: quy định phương thức gởi GET & POST | |
| 12 | | Phân biệt phương thức get và post trong thẻ form.   |  |  | | --- | --- | | Get  -Khi gởi dữ liệu đi sẽ hiển thị trên URL  -Thông tin gởi đi không bảo mật,an toàn  -Dữ liệu gởi đi bị giới hạn | Post  -Không hiển thị  -Thông tin nhạy cảm gởi đi an toàn hơn.  -Ko giới hạn,thường để gởi dữ liệu lớn về sever | | |
| 13 | | Cách khai báo biến bằng từ khóa let và var khác nhau như thế nào ?. Phạm vi của biến?  -Biến là tên gọi được gắn cho 1 vùng nhớ dữ liệu.  -Dữ liệu được lưu trữ gọi là giá trị.  -Giá trị có thể bị ghi đè lại.  -Bắt buộc khai báo biến trước khi sử dụng.   |  |  | | --- | --- | | -VAR : có thể khai báo lại biến trùng tên;  -Không bắt buộc khai báo trước khi sử dụng(nhờ cơ chế hoisting tất cả các biến được khai báo bằng từ khóa var sẽ được đưa lên đầu tiên của chương trình khi chạy (nhưng không khai báo giá trị);  -Phạm vi truy cập của var nằm trong function; | -LET:không thể;  -Phải khai báo let trước khi sử dụng;  -Nằm trong phạm vi truy cập block scope {} | | |
| 14 | | Trong Javascript có bao nhiêu loại kiểu dữ liệu ?  Có 2 kiểu dữ liệu:   1. Kiểu dữ liệu nguyên thủy:  * String: kiểu chuỗi. Được viết trong cặp “” hoặc ‘’ * Number: kiểu số, bao gồm cả số nguyên và số thực * Boolean: trả về true hoặc false * Null: đại diện cho 1 biến không chứa giá trị nào cả * Underfind: biến có dữ liệu và giá trị là undefind (không xác định) * Symbol: kiểu dữ liệu mới được sử dụng sau này. Thường dùng cho đối tượng.  1. Kiểu đối tượng:  * Array (mảng) * Object: Do chương trình cung cấp hoặc lập trình viên tự định nghĩa   Làm thế nào để xác định được biến có kiểu dữ liệu gì?  Type of:dùng để kiểm tra kiểu dữ liệu của biến.  Phân biệt Null vs undefine:  -Null là kiểu dữ kiệu dc gán cho biến,thường đc hiểu là không biết | |
| 15 | | Các cách tạo chuỗi chứa dấu nháy ? “ ‘Các cách tạo chuỗi chứa dấu nháy ?’ ” | |
| 16 | | NaN là gì? NaN === NaN có đúng không?Vì sao?  NaN là viết tắt của not a number | |
| 17 | | Cho 1 bài toán liên quan đến ++ trước và ++ sau. Yêu cầu dự đoán kết quả, và tại sao lại có kết quả như vậy? | |
| 18 | | Các loại toán tử trong js? Cho biết độ ưu tiên của các toán tử trong một biểu thức? | |
| 19 | | Toán tử ba ngôi là gì? Cú pháp? | |
| 20 | | So sánh null và rỗng | |
| 21 | | Toán tử == và === khác và giống nhau như thế nào | |
| 22 | | Phân biệt giữa 2 toán tử && và || trong JavaScript? | |
| 23 | | Các hàm dùng để tạo thông báo trong Javascript ? | |
| 24 | | Có bao nhiêu Statement control(câu lệnh điều khiển) trong Java Script ? | |
| 25 | | So sánh sự khác nhau giữa if và switch case?. | |
| 26 | | Switch case so sánh == hay ===. Đặt ra trường hợp là so sánh bằng thì khi nào sử dụng if bậc thang? Khi nào sử dụng switch case; | |
| 27 | | Các biểu thức và luồng thực thi của for. Nếu thiếu 1 hoặc tất cả các biểu thức thì vòng for sẽ chạy như thế nào? | |
| 28 | | Đặt ra 1 bài toán. Xác định bài toán cần sử dụng vòng lặp nào? | |
| 29 | | So sánh giống và khác nhau giữa for, while và do..while | |
| 30 | | So sánh break và continue | |
| 31 | | Đặc điểm mảng một chiều trong Javascript | |
| 32 | | Các cách khởi tạo một mảng kiểu String trong JavaScript | |
| 33 | | - 1 số hàm thao tác với mảng? - Phân biệt push() và unshift()? - Phân biệt push() và pop()? - Phân biệt shift() và unshift()? | |
| 34 | | Phân biệt tham trị và tham chiếu trong Javascript | |
| 35 | | Phân biệt giữa Hàm có return và hàm không có return | |
| 36 | | Lập trình hướng đối tượng là gì | |
| 37 | | Phân biệt class và object | |
| 38 | | Các đặc điểm trong Lập trình hướng đối tượng. Đưa ra 1 ví dụ 1 trong 4 tính chất. (lưu ý cần phải trình bày được 4 tính chất bằng tiếng anh) | |
| 39 | | Constructor là gì? Trong 1 class có nhiều hơn 1 contrustor được hay không? | |
| 40 | | Những phương thức nào cho phép tương tác với chuỗi | |
| 41 | | Các câu lệnh để vẽ 1 hình tròn. Giải thích từng câu lệnh? | |