**CÂU HỎI AUDIT MODULE 1**

**BOOTCAMP PREPARATION 2.0**

|  |  |  |
| --- | --- | --- |
| **STT** | **Câu hỏi** | |
| 1 | | Repository là gì? có bao nhiêu loại repository?  Trả lời : là nơi chứa toàn bộ mã nguồn , bao cả các file source code và lịch sử thay đổi từng file - Có 2 loại repository + Loại Local repository trên máy tính các nhân của lập trình viên. + Loại remote repository : repo chung được lưu trữ ở máy chủ Git | |
| 2 | | Các cấu lệnh căn bản để làm việc với git?  Git init Git commit –m“” Git status Git push Git add | |
| 3 | | Giải thuật là gì? các cách để biểu diễn thuật toán?  Trả lời: Giải thuật hay còn gọi là thuật toán, là những giải thuật bao gồm các  bước để giải quyết một vấn đề nào đó. - Các cách biểu diễn thuật toán + Mã giả (Preudo –Code) + Lưu đồ (Flowchart) \*\*\*\* Mở rộng thêm câu hỏi : - Lập trình : là quá trình tạo ra các chỉ dẫn để mình ra lệnh cho máy tính  thực hiện công việc nào đó (task) - Ngôn ngữ lập trình là phương tiện viết ra các chỉ dẫn (có nhiều ngôn ngữ lập trình khác nhau Vd như : PHP, java, C#) | |
| 4 | | Mô tả thuật toán tìm kiếm 1 phần tử trong mảng bằng mã giả? Mã giả Lưu đồ - Mô tả thuật toán bằng ngôn  ngữ tự nhiên. - Dùng tiếng anh để mô tả thuật  toán. - Không quy định chặt chẽ - Dùng các khối cố định - Dùng các khối mô tả thuật toán - Có quy định chặt chẽ | |
| 5 | | Mô tả thuật toán sắp xếp mảng số nguyên bằng mã giả | |
| 6 | | Trình bày ý tưởng của 1 thuật toán bất kì? | |
| 7 | | Nêu các thẻ để tạo danh sách ? Sự khác nhau giữa <ul> và <ol> ?  Trả lời : <ol> <li>: để tạo danh sách có thứ tự <ul> <li> tạo danh sách không có thứ tự | |
| 8 | | Các thẻ tạo nên 1 table? Phân biệt th và td?  Trả lời: Các thẻ tạo nên 1 table là : Thẻ <table>:khai báo một mảng  Thẻ <tr> : Khai báo một dòng  Thẻ <th> : Khai báo một ô tiêu đề, dùng để ghi tiêu đề Thẻ <td> : Khai báo một ô dữ liệu, dùng để ghi nội dung bình thường | |
| 9 | | Phân biệt innerHTML và innerText?  Trả lời : - innerHTML – Sẽ xử lý một thẻ HTML nếu được tìm thấy trong một chuỗi - innerText – Không xử lý thẻ HTML nếu được tìm thấy trong một chuỗi | |
| 10 | | Nêu một số thẻ HTML cơ bản mà bạn biết  Trả lời: - Thẻ tạo tiêu đề: Thẻ <h> - Hiển thị văn bản : thẻ <div> , <p>, <span> - Xuống dòng <br> - Thẻ tạo liên kết : thẻ <br> - Tạo ảnh : <img>  -Tạo nút<button> - Thẻ style: dể thay đổi thuộc tính <html>, <head>, <img>,<links>, <title>,<table>,... | |
| 11 | | Các thuộc tính cơ bản của thẻ form  Trả lời : \*\*\* : Mở rộng : - Khái niệm Form : Form cho phép người dùng nhập dữ liệu và gửi thông  tin về phía server (máy chủ web) - Các thuộc tính cơ bản của thẻ form: + action: mô tả URL của server, nơi mà dữ liệu được gửi đến + method: Xác định kiểu phương thức được sử dụng để gửi dữ liệu gồm  POST và GET | |
| 12 | | Phân biệt phương thức get và post trong thẻ form.  GET - Dữ liệu được hiển thị lên thanh địa chỉ của trình duyệt URL. - Không nên sử dụng để gửi các  dữ liệu nhạy cảm ( Chẳng hạn  như mật khẩu ) - Thường sử dụng để gửi những  dữ liệu nhỏ. POST - Dữ liệu được gửi đi không hiển thị trên thanh địa chỉ trình  duyệt. - Được sử dụng để gửi các dữ  liệu nhạy cảm . - Không giới hạn dung lượng gửi đi . Thường được sử dụng để  gửi liệu lớn lên server | |
| 13 | | Cách khai báo biến bằng từ khóa let và var khác nhau như thế nào ?. Phạm vi của biến?  Var Let •Giống nhau -Đều là từ khóa keyword - Cùng được dùng để khai báo biến trong javascript. •Khác nhau - Hai từ khóa var và let khác nhau ở phạm vi sử dụng biến + Phạm vi sử dụng biến sử dụng var là phạm vi hàm hoặc bên ngoài hàm , toàn cục. + Phạm vi biến sử dụng let là phạm vi một khối (block), xác định bởi cặp {}. | |
| 14 | | Trong Javascript có bao nhiêu loại kiểu dữ liệu ?. Làm thế nào để xác định được biến có kiểu dữ liệu gì?  - Các kiểu dữ liệu trong javascript: + String, boolearn, number, NaN, Underfile, null, Object | |
| 15 | | Các cách tạo chuỗi chứa dấu nháy ?  Trả lời :  +Các chuỗi sử dụng dấu nháy kép và dấu nháy đơn có hiệu quả giống nhau.  +Vì không có quy ước hoặc ưu tiên chính thức cho các chuỗi được trích  dẫn đơn hoặc kép, tất cả những gì quan trọng là sự nhất quán trong các file  code | |
| 16 | | NaN là gì? NaN === NaN có đúng không?Vì sao?  Trả lời : NaN là một dạng viết tắt của “Not a Number”. Vì NaN luôn so sánh không  bằng với bất kỳ số nào, kể cả NaN, nó thường được sử dụng để chỉ ra điều kiện  lỗi cho một hàm mà đáng ra phải trả về một số hợp lệ. Khi một chuỗi hoặc một  cái gì đó khác đang được chuyển đổi thành một số và điều đó không thể được  thực hiện, thì ta sẽ thấy NaN. | |
| 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? Các loại toán tử +Toán tử số học + - \* / %: Lấy phần dư của phép chia ++, -- +Toán tử so sánh >,<,>=, <= == so sánh giá trị === so sánh 2 kiểu dữ liệu != trả về true nếu không bằng về mặc  giá trị. !== trả về true nếu không bằng về  mặc giá trị và kiểu dữ liệu +Toán tử logic && trả về true khi cả 2 vế đều true | | trả về true khi 1 trong 2 vế true ! phủ định kết quả EX: !true ◊ False Toán tử gán =,+=, -=,/=,%= | |
| 19 | | Toán tử ba ngôi là gì? Cú pháp?  Toán tử 3 ngôi trong javascript cho phép chúng ta kieemrtra một điều kiện nào  đó trong một dòng code duy nhất và trả về giá trị, thay vì sử dụng nhiều câu lệnh if-else, giúp cho việc code đơn giản và thông minh hơn . - Cú pháp Condition ? Expression true case: Expression false case: Ví dụ: x%==0? alert (“số chẵn”): alert (“số lẻ”) | |
| 20 | | So sánh null và rỗng  Trả lời: - null được sử dụng để chỉ không có gì, đối tượng chưa tồn tại - rỗng được sử dụng để chỉ một chuỗi duy nhất có độ dài bằng không | |
| 21 | | Toán tử == và === khác và giống nhau như thế nào  Trả lời: == so sánh 2 giá trị === so sánh 2 giá trị và kiểu dữ liệu | |
| 22 | | Phân biệt giữa 2 toán tử && và || trong JavaScript?  Trả lời : && Trả về kết quả khi cả 2 đều đúng | | Trả về kết quả khi 1 trong 2 đúng | |
| 23 | | Các hàm dùng để tạo thông báo trong Javascript ?  Trả lời: alert(), confirm(), prompt() | |
| 24 | | Có bao nhiêu Statement control(câu lệnh điều khiển) trong Java Script ?  Trả lời : Có 2 câu lệnh Statement control trong Java Script Câu lệnh if—else: Câu lệnh if, câu lệnh else, kết hợp if .. else lồng nhau Câu lệnh switch - case | |
| 25 | | So sánh sự khác nhau giữa if và switch case?.  IF : - Có thể so sánh hơn, nhỏ hơn, bằng hoặc không - Biểu thức điều kiện của if có giá trị trả về là boolean Swich-Case  - Chỉ có thể só sánh bằng hoặc không bằng - Biểu thức có giá trị trả về là number, string,Boolean  -Swich case hỗ trợ kiểm tra nhiều điều kiện | |
| 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  Trả lời : Switch case chỉ có thể sử dụng toán tử = = = - Sử dụng if bậc thang khi sử dụng tối đa 3 điều kiện với if - Sử dụng Switch – case sử dựng khi nhiều điều kiện | |
| 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?  Biểu thức : for(Khối lệnh khởi tạo(initial – action, Khối lệnh điều kiện lặp(condition), khối  lệnh tăng hoặc giảm(action – after earch interaction)){//Nội dung lặp body} | |
| 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  Trả lời : \*\*\*: Biết trước số lần lặp sử dụng for - Chưa biết trước số lần lặp thì sử dụng while và do while + Sự khác nhau giữa while và do while: o While kiểm tra ddieeeuf kiện rồi mới lặp oDo-while thực hiện lặp rồi mới kiểm tra . (lặp ít nhất 1 lần dù  đúng, dù sai, dù điều kiện lặp bị sai) | |
| 30 | | So sánh break và continue  Trả lời: - Break dùng để thoát bòng hiện tại - Continue nhảy tới vòng lặp tiếp theo | |
| 31 | | Đặc điểm mảng một chiều trong Javascript  Trả lời : Mảng một chiều là kiểu mảng trong đó các phần tử được sắp xếp liên tục và có thứ tự trong bộ nhớ máy tính | |
| 32 | | Các cách khởi tạo một mảng kiểu String trong JavaScript  Trả lời: Các cách duyệt mảng môt chiều: - Dùng for, while, do while, để duyệt mảng - Với vòng lặp for: ofor-in: duyệt index của mảng, chỉ duyệt một chiều  ♣for ( let index in myArray){} ofor of : hỗ trợ duyệt giá trị của mảng, chỉ duyệt một chiều ♣for ( let value of myArray){}  \*\*\*\* Mở rộng câu hỏi : - Mảng 2 chiều thực chất là mảng một chiều , với một phần tử chứa amngr con khác. Cú pháp truy xuất phần tử Arr2d [index-row][index-col] | |
| 33 | | - 1 số hàm thao tác với mảng? - Phân biệt push() và unshift()? - Phân biệt push() và pop()? - Phân biệt shift() và unshift()?  Trả lời : - Chèn phần tử vào cuối mảng: arr.push(11) - Xóa phần từ ở cuối mảng : arr.pop(11) - Thêm phần tử ở đầu mảng: arr.unshift(11) - Xóa phần tử ở đầu mảng: arr.shift() - Thêm ở vị trí bất kì : 4,9,6,7 arr.splice(2,0,11); // thêm 11 tại index 2 arr.spice (2,0,12):// thêm 11,12 tại index 3 arr.spice(0,0,7,8);// thêm 7,8 tại index 0 - Xóa ở vị trí bất kì arr.splice(1,1); //4,6,7 arr.splice(2,2); //4,9 - Hàm chuyển mảng thành chuỗi let str = arr.toString(); let str =arr.join(“ ”) document.write(str) - Hàm đảo mảng Ar.reserve() -Hàm sắp xếp arr1.sort(function(a,b){ return a-b; // sắp xếp tăng return b-a;// sắp xếp giảm }) | |
| 34 | | Phân biệt tham trị và tham chiếu trong Javascript  Kiểu Tham Trị:  +Kiểu này dùng cho các kiểu dữ liệu nguyên thuỷ: String, Number, Boolean, BigInt, Symbol, undefined, null.  +Kiểu này thì cũng khá là đơn giản, khi gán cho nó một giá trị thì nó sẽ lưu lại giá trị  đó và tại một thời điểm thì chỉ lưu một giá trị.  Kiểu Tham Chiếu:  Kiểu này thì phức tạp hơn kiểu tham trị và được dùng cho các kiểu dữ liệu phức tạp: Object, Array, Function. Ở đây khi gán cho nó một giá trị thì nó không lưu lại giá trị này mà thực chất nó lưu lại địa chỉ của ô nhớ lưu giá trị này. | |
| 35 | | Phân biệt giữa Hàm có return và hàm không có return  -Hàm không có return thì không có trả về  -Hàm return thì có trả về :  + Dùng từ khóa ‘return’ trong thân hàm để trả về kq của hàm.  + Mỗi hàm chỉ có 1 giá trị trả về, giá trị có thể là biến,số,chuỗi,mảng  + Gặp từ khóa return thì sẽ thoát khỏi hàm hiện tại.  + Để nhận về giá trị trả về của Hàm. Dùng toán tử gán | |
| 36 | | Lập trình hướng đối tượng là gì  -Còn được gọi là lập trình OOP. Là 1 kỹ thuật lập trình cho phép lập trình viên tạo ra các đối tượng trong code, ánh xạ các đối tượng trong thực tế vào ngôn ngữ lập trình.  -Đối tượng bao gồm:  + Thuộc tính : các kiểu dữ liệu, tính chất của đối tượng  + Hành vi : các hành động của đối tượng | |
| 37 | | Phân biệt class và object  -Class dùng để mô tả nhóm các đối tượng có thuộc tính và hành vi tương tự hau  -Object là thể hiện của class( là sp dc tạo ra từ class | |
| 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)  -Tính bao đóng – encapsulation  -Tính trừu tượng – abstraction  -Tính kế thừa – inhertitance  -tính đa hình – polymorphism | |
| 39 | | Constructor là gì? Trong 1 class có nhiều hơn 1 contrustor được hay không?  -Là hàm dùng để khởi tạo đối tượng được gọi thông qua từ khóa new.  -Trong 1 class chỉ có 1 contructor | |
| 40 | | Những phương thức nào cho phép tương tác với chuỗi   1. **Đếm số lượng ký tự của chuỗi** 2. **Chuyển chuỗi về dạng chữ in hoa** 3. **Chuyển chuỗi về dạng chữ thường** 4. **Nối các chuỗi lại với nhau** 5. **Xác định chỉ số của một chuỗi con** 6. **Trích xuất một ký tự trong chuỗi** 7. **Phương thức substring()** 8. **Phương thức substr()** 9. **Thay thế tất cả nội dung trùng khớp** 10. **Thay thế nội dung không phân biệt trường hợp chữ hoa hay thường** | |
| 41 | | Các câu lệnh để vẽ 1 hình tròn. Giải thích từng câu lệnh? | |