|  |  |
| --- | --- |
| **KEY WORLD** | **MAIN CONTEN** |
| **lập trình và ngôn ngữ lập trình** | Đưa ra các chỉ thị , chỉ dẫn để giải quyết vấn đề |
| **thuật toán** | bao gồm các chỉ thị để giải quyết vấn đề |
| **Pseudo-code** | ngôn ngữ tự nhiên, ( độc lập vớic các ngôn ngữ lập trình) |
| **Flowchart** | biểu tượng , khối có cấu trúc cú pháp chỉ dẫn để giải quyết bài toán |
| **Psedo-code &  FLowchart** | **Psedo-code**   * ưu:đơn giản, gần gũi không cần kiến thức về lưu đồ * nhược: dài dòng, khó hiểu, khó diễn đạt thuật toán   **FLowchart**   * ưu: dễ nhìn, dễ hiểu,dễ diễn đạt * nhược: cần ghi nhớ các khối |
| **Source code** | **is dãy câu lệnh viết bằng java/c/c++/...** |
| **SCMs(Source code Managerment System)** | 1. **Work grop** 2. **manager files & contro version** 3. **work together trên flies, hợp nhất changes** 4. **Follow & check changes được yêu cầu & thực thi** 5. **Follow tình trạng sửa lỗi và thực thi** |
| **VCS( Version contro system)** | **Restore old version of files or toàn bộ dự án**  **check changes đã được thực hiện theo thời gian** |
| **Git** | **is version control system phân tán** |
| **snap shot** | **all source code tại 1 thời điểm** |
| **commit** | **tạo thay đổi đáng kể(tính năng mới, sửa lỗi...) tạo ra snap shot** |
| **Repo( Repository)** | **kho chứa all source code, files ,history of file và commit các changes**   1. **loca repo(máy cá nhân)** 2. **remote repo(máy chủ)** |
| **Các câu lệnh chính** | 1. **git add . ( đưa file ->staged)** 2. **git commit -m " " ( ghi nhận các thay đổi** 3. **git push origin main/master ( đẩy commit từ loca repo-> remote repo)** 4. **git clone ( remote repo)** 5. **git pull (cập nhật source code)** 6. **git init (tạo repo từ loca repo)** |
| **HTML(Hypertext markup language)** | **ngôn ngữ đánh dấu siêu văn bản tạo web** |
| **< HTML>** | 1. **Block(<h>,<p>,<div>): hiển thị độ rộng màn hình** 2. **Inline (<span> <a>): hiển thị độ rộng vừa đủ với nội dung** |
| **Các nhóm thẻ HTML chính** | 1. **<h> : Tiêu đề** 2. **<div>: (block-level)** 3. **<span>: (in-line)** 4. **<br> : xuống dòng** 5. **<ul> (<li>) :List không thứ tự** 6. **<ol>(<li>):List có thứ tự** 7. **<label>:type cho input, need for & id** 8. **<hr>:kẻ một đường ngang** 9. **<font> :kiểu chữ chữ** 10. **<IMG SRC=”URL”>: chèn ảnh** 11. **<a heref="URL">: chèn link** 12. **<audio controls>: chèn audio** 13. **<video controls> : chèn video** 14. **<sub>:2  (tạo văn bản 2 nhỏ phía dưới)** 15. **<sup>:2 ( tạo văn bản 2 nhỏ phía trên)** 16. **<strong>: im đậm văn bản cần nhấn** |
| **Thuộc tính cơ bản HTML** | 1. **"action": <from> URL** 2. **"alt":text thay thế cho các phần tử không hiển thị,** 3. **"class":đặt name cho 1 phần tử** 4. **"color":** 5. **"height"** 6. **"width"** 7. **"hidden" một phần tử không được hiển thị.** 8. **"href":URL** 9. **"id":** 10. **method"<form>:get or post** 11. **"name":** |
| **HTML form** | **is cơ chế cho phép user nhập dữ liệu & sent -> sever** |
| **<form></from>** | 1. **"action"=URL  method=(get or post)** 2. **<fieldset> và <legend> nhóm trường danh sách** |
| **GET** | **Data-> hiển thị trên URL**  **không bảo mật**  **Ít dung lượng** |
| **POST** | **Date ẩn**  **Bảo mật**  **Không hạn chế dung lượng** |
| **các thẻ cơ bản** | **<input>: Use Tạo trường**  **<select> &<<option>> :List ( drop down)**  **<textarea>:text big**  **thuộc tính của <textarea>**   1. **rows: số lượng dòng (height)** 2. **cols: số lượng cột(width)** |
| **Các thuộc tính cơ bản** | 1. **name** 2. **Disabled (vô hiệu hóa trường, User k thể tương tác)** 3. **Value** 4. **Type** 5. **text** 6. **Passworld** 7. **summit** 8. **reset** 9. **radio: chọn 1 value cho trước** 10. **check box: chọn nhiều value cho trước** 11. **hidden: chứa value giá trị của form ex: [admin]** 12. **button: tạo nút gắn "onclick"** 13. **color** 14. **date** 15. **date time-loca** 16. **email** 17. **month** 18. **number** 19. **range** 20. **seach** 21. **tel** 22. **time** 23. **week** 24. **font-size: cỡ chữ** 25. **margin= auto: lề** 26. **cellspacing: khoảng cách giữa các đường viền** 27. **text-align: căn giữa** 28. **background: màu nền** 29. **boder=1px: đường viền** 30. **padding:khoảng cách content đến boder** |
| **TABLE trong HTML** | **Use show list**  **Các thẻ quan trọng**   1. **• <table>:** 2. **• <tr>: row** 3. **• <th>: header** 4. **• <td>: 1 ô data** 5. **"colspan": ô chiếm nhiều cột** 6. **"rowspan" : ô chiếm nhiều dòng** |
| **Javascript**  **File save đuôi.js** | * **usually use  Website,chạy trên trình duyệt** * **thay đổi nội dung, giao diện, tăng tính tương tác trang web** * **File save đuôi.js** |
| **Nhúng Javascript** | 1. **<script></script>** 2. **Use file .js  or web** |
| **Thẻ <script>** | **<script><p>java script</p></script>** |
| **file .js or web** | * **<script src="myScript.js"></script>(cuối tài liệu chạy nhanh hơn)** |
| **comment** | **1 dòng use  // , nhiều dòng use  /\*...\*/** |
| **variable** | **là một tên gọi được gắn cho một vùng nhớ chứa dữ liệu(value)** |
| **getElementById()** | **Use "seach" id="demo"& change content  thành "Hello JavaScript"**  **<p id="demo">JavaScript can change HTML content.</p>**  **<button type="button" onclick='document.getElementById("demo").innerHTML = "Hello JavaScript!"'>Click Me!</button>** |
| **Các sự kiện:( sự tương tác)** | **onclick or addevenListener (click)**  **onchange or addevenlistener (change) <select> & <option>**  **onload:web đã được tải**  **"onmouseover" và "onmouseout" or hàm addven listener( đưa con trỏ chuột vào or ra)**  **onkeydown:phím đã được nhấn xuống** |
| **Các hàm thông báo** | 1. **alert():thông báo thông tin dạng văn bản** 2. **prompt(): ô nhập liệu để lấy thông tin trả về value "string"** 3. **confirm():ô nhập liệu trả value bloolean** 4. **document.write(), : Text** 5. **console.log(): hiển thị cho dev** |
| **Các hàm hiển thị đầu ra** | 1. **document.write(), : Text** 2. **getElementById().innertext/innerHTML=''** 3. **console.log(): hiển thị trong cửa sổ console** |
| **variable** | **là một tên gọi được gắn cho một vùng nhớ chứa dữ liệu(value)** |
| **Var** | 1. **phạm vi toàn cục** 2. **ghi đè** 3. **khai báo trước sau đều được** |
| **Let** | 1. **Phạm vi trong block code ( dấu {})** 2. **không thể ghi đè** 3. **Khai báo trước mới dùng được** |
| **Name for Variable** | **Nên tránh đặt tên biến==từ khoá và phân biệt font A or a**  **Không dùng từ khóa "java script"**  **Tuân thủ:came\_Case (bướu lạc đà)** |
| **CONST** | **•Name cho 1 value cố định và value ko đổi**  **• Ex, Use const PI =3.14159:**  **var area = radius \* radius \* 3.14159;**  **constPI = 3.14159; var area = radius \* radius \* PI;** |
| **khai báo const** | **const CONSTNAME = value;**  **note: Name Const bắt buộc viết hoa** |
| **(Data Type)** | 1. **number :6 or 6.6** 2. **"String":Hello or A** 3. **boolean:true,false** 4. **object: Đối tượng** 5. **Undefined: không xác định**   **• var x = 10; // x mang giá trị 10, kiểu dữ liệu của x là số nguyên**  **• var gender = true; // gender mang giá trị true, kiểu dữ liệu của gender là kiểu boolean(logic)** |
| **Type of Operator** | **• Toán tử là các ký hiệu được sử dụng để thực hiện các thao tác trong các biểu thức và sinh ra kết quả cuối**   1. **số học** 2. **bloolean** 3. **so sánh** 4. **gán** 5. **cộng chuỗi** 6. **type of** |
| **số học (arithmetic)** | 1. **+ Cộng** 2. **- Trừ** 3. **\* Nhân** 4. **/ Chia % Chia lấyphần dư (Modulus) var z= x % y (5/2=z=1)** 5. **++ Tăng 1 giá trị  var  x = 5; x++; var z =6;** 6. **-- Giảm1 giá trị var  x = 5; x--; var z =4;** |
| **so sánh (comparision)** | **(==) :bằng về Value**  **Ex:**  **var x = 5; var y = x == 5;//True ; var z = x == "5"; //true**  **(===) : bằng cả value và Type of**  **Ex:**  **var x = 5; var y = x === 5;//True ; var z = x === "5"; //False**  **(!=) :  không bằng về value**  **Ex:**  **var x = 5; var y = x !=8// True ; var z = x !=5;// False**  **(!==) :Không bằng cả value và type of**  **Ex:**  **var x = 5; var y = x !==5// True ; var z = x!=="5";//False** |
| **Bloolean** | **(true/false):**   1. **&& :”và”  (1 biến False) => biểu thức (a&&b) False** 2. **||:“hoặc” (1 biết True)=> biểu thức (a ||b) True** 3. **! :“phủ định”: Đảo ngược Value like a true => b false** |
| **gán** | **use để gán giá trị cho một biến và use với tất cả các kiểu dữ liệu**  **var x =10;**  **x +=5 //tương đương x+5**  **More : -+, \*+,/+,%+** |
| **'+' String** | **Use để nối hai chuỗi. và  Có thể nối chuỗi với số**  **x = 5  +5 ; (10)**  **y = 5 + "5"; ("55"')**  **z= "hello" + 5; ("hello5")** |
| **typeof** | **lấy về kiểu dữ liệu của một biến hoặc giá trị** |
| **Độ ưu tiên của các toán tử** | 1. **từ trái sang phải** 2. **Use “()” để thay đổi độ ưu tiên của các toán tử**   **ex:**  **var x = 5;**  **var y = 10;**  **var z = (++x \* y) < 5 \* 10 && 6 > 3;**  **(6 \* y) < 5 \* 10 && 6 > 3;**  **( 60 ) < 50 && 6 > 3;**  **false && true;**  **false;** |
| **Các câu lệnh điều khiển** | * **thực thi  từ trên xuống dưới and ---->** * **Use (contro lflow statement)**   **• Các câu lệnh điều khiển của JavaScript:**   1. **If & Swich** 2. **For & Do- While** 3. **nhảy(jumpstatement)** |
| **IF-ELSE & SWITCH-CASE** | * **ra quyết định thực thi dựa vào đk cho trước** |
| **IF** | **IF (khuyết) {**  **document.write("")**  **}**  **Ex:**  **let average  = parseInt(prompt("average"));**  **If (average>=9) {**  **document.write("")**  **}** |
| **IF\_ELSE** | **IF (đầy đủ) {**  **document.write("")**  **}ELSE (){**  **document.write("")**  **}**  **Ex:**  **let average  = parseInt(prompt("average"));**  **IF (average>=9) {**  **document.write("")**  **}ELSE**  **document.write("below")**  **}** |
| **IF-ELSE lồng nhau** | **IF (lồng nhau) {**  **}IF (){**  **document.write("")**  **}ELSE IF(){**  **document.write("")**  **}ELSE{**  **}IF (){**  **document.write("")**  **}ELSE IF(){**  **document.write("")**  **}**  **Ex:**  **IF (a>b {**  **}IF (a>c){**  **document.write("a max")**  **}ELSE IF(a<c){**  **document.write("a ko max")**  **}ELSE (a<b){**  **}IF (a<c){**  **document.write("a min")**  **}ELSE IF(a>c){**  **document.write("a ko min")**  **}** |
| **IF-ELSE bậc thang** | **IF (bậc thang) {**  **document.write("")**  **}ELSE IF (){**  **document.write("")**  **}ELSE IF (){**  **document.write("")**  **}ELSE (){**  **document.write("")**  **}**  **Ex:**  **IF (avarage>=9) {**  **document.write("Exellent")**  **}ELSE IF (avarage>=7){**  **document.write("Good")**  **}ELSE IF  (avarage>=5) {**  **document.write("average")**  **}ELSE (avarage<5) {**  **document.write("Below")**  **}** |
| **Viết mã sạch** | **”tách biến”: Đưa biểu thức của chúng ta ra bên ngoài khối lệnh if và gán giá trị của nó cho một biến (diemtb)**  **var diemtb=avarage>=5;**  **IF (diemtb) {**  **console.log(you are : + " diemtb.");**  **}ELSE{**  **console.log(you are: " another.");**  **}** |
| **SWITCH-CASE** | **switch (biểu thức) {**  **case 1: document.write"" break;**  **case 2: document.write"" break;**  **case n: document.write"" break;**  **default : document.write"undefined" break;**  **Ex:**  **let day = parseInt(prompt("day"));**  **switch (day) {**  **case 1: document.write"Mon" break;**  **case 2: document.write"Tueday" break;**  **case n: document.write"Sun" break;**  **default : document.write"undefined" break;** |
| **GỘP CASE** | **Ex:phân loại ngày trong tuần với ngày cuối tuần:**  **switch (day) {**  **case 1: mon**  **case 2:tue**  **case 3: web**  **case 4: thu**  **case 5: fri**  **console.log("Weekday");**  **break;**  **case 0: sun**  **case 6: sta**  **console.log("Weekend"); }** |
| **SWITCH-CASE lồng** | **Ex:**  **let value = parseInt(prompt ("input value"));**  **switch (true) {**  **case value > 0:**  **alert (1);**  **break;**  **case value < 0:**  **alert(-1);**  **break;**  **default:**  **alert(0);**  **break;**  **}** |
| **IF-ELSE** | 1. **>;<;=;!=** 2. **Đk, với Value trả về là boolean** 3. **ĐK true->1 khối lệnh được thực thi** 4. **Use tối đa 3 IF-ELSE lồng nhau** |
| **SWITCH-CASE** | 1. **= & !=** 2. **Đk, với Value trả về là :String, Number, Bloolean** 3. **Đk True-> 0 có break;-> all khối lệch next được thực thi** 4. **check nhiều Đk trong 1 biểu thức** |
| **Toán tử 3 ngôi** | **condition ? expression\_true\_case : expression\_false\_case**   * **condition: Đk** * **expression\_true\_case : biểu thức Use true** * **expression\_false\_case : biểu thức Use false**   **Ex:**  **• Tìm giá trị lớn nhất trong 2 số:**  **var maxValue = number1 > number2 ? number1 : number2;**  **• Tính số lượng dựa trên điều kiện khoảng cách:**  **va count = isHere? getHereCount(index) : getAwayCount(index);** |
| **3 ngôi lồng nhau** | **Ex:**  **diem<=75 ? đk2>=50 ? good : average ;** |