

عند تطبيقه: داخل صنف div ، input ، btn ، link ، div ...
 تطبق في div (user) (optional user)

#HTML

```
<input type="text" id="productInput">
<button id="productBtn" >Add</button>
<div id="productContainer"></div>
```

لما نجيء أي شيء (متردج) فنأخذ مكرر فيه، يجب مرقة معلم id لاستدعاء واحد ومحفظ المطابقات.

الخطوة 1: تعرف المتغيرات: (بيانات المتغيرات التي نريد)

#1 // setup Variables:

```
Var productBtn = document.getElementById("productBtn");
Var productContainer = document.getElementById("productContainer")
```

نعرف objects كل جزء يدخل array.

```
data = [ {key: value} , {id: 1 , name : "iphoneX" , desc : "this is--"} , {id: 2 , name: "iphonell" , desc : " " } , {id: 3 , name: "iphone12" , desc : " " } , {id: 4 , name: "iphone13" , desc : " " } ] ;
```

لـ object, in Comma بمعنى

Console.log(data) → array of objects : سطوع المحتوى

مقدمة 2: بعد اضطرابات يلي خلقته الصفة العامة عنده

#2 draw elements : رسم المحتوى

Function draw(items){
 console.log(items) } مادا يطبع؟
 وآخر زانغ (items) سعرى في
 موجود صار

ex - يعني اطبع كل واحد من المحتوى (ك) لمحنة . #1

function draw(items){
 console.log(items)
draw() → array of objects في Console يطبع
 كل واحد من المحتوى #2

: Code في #3 *

function draw(items){

items.forEach(function(ele){})

function of function

higher order function

console.log(ele)})}

رسالة

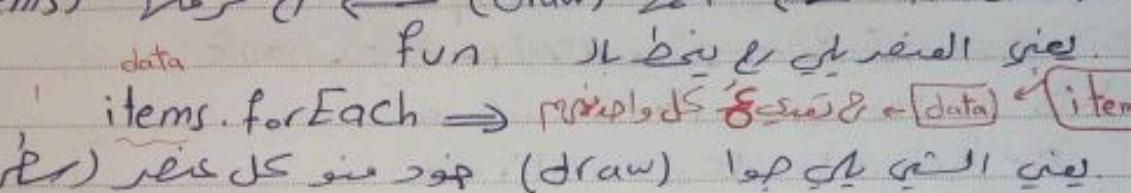
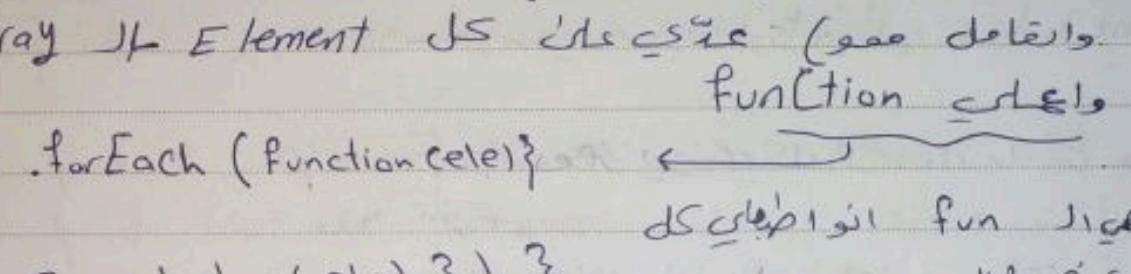
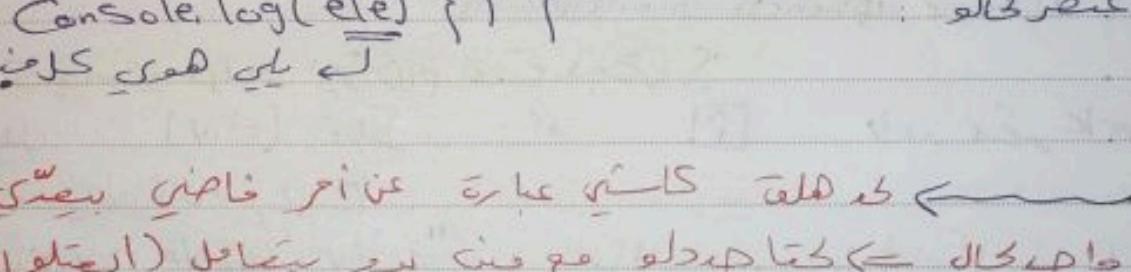
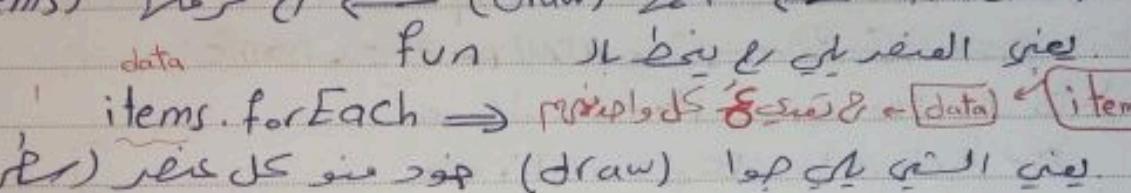
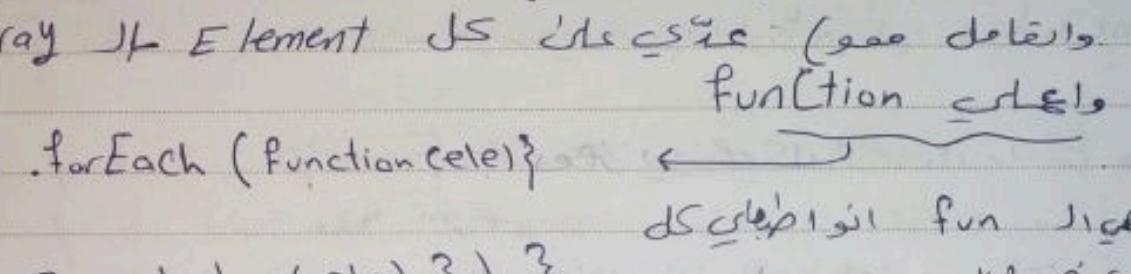
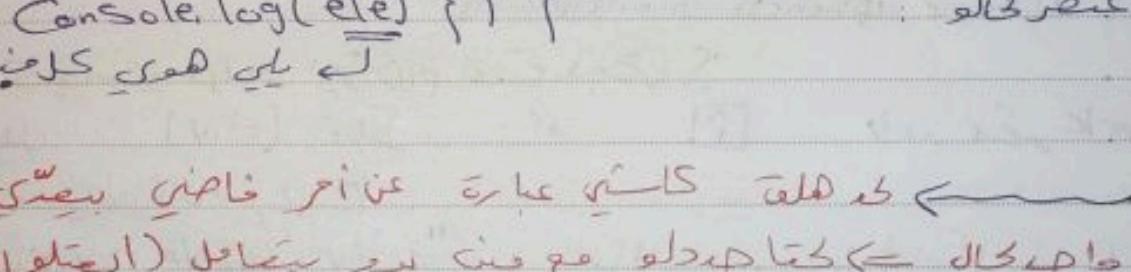
draw(data)

* اخ يطبع كل فتح (ك) بـ
 كار (أ) في الـ والي موسي ↑



لتكون...

ـ مراجعة امتحانات code

عندئـى مـنـجـرـاـتـاـ (array) مـنـكـمـاـ (items) اـعـدـاـتـاـ (data) اـسـنـدـاـتـاـ (draw) لـمـنـعـلـاـتـاـ (function) يـعـنـى الصـفـرـيـاـتـاـ عـنـيـظـاـتـاـ items.forEach =>  يعني الذي يـلـى جـمـعـاـ (draw) هوـوـ مـنـوـ كلـغـفـرـ (رـجـرـ طـالـوـ) وـارـقـامـلـاـ معـوـ (Element) عـنـيـفـيـاـ كلـغـفـرـ (function) وـاعـاتـاـ funtion items.forEach (function ele) { } funtion funtion ele =>     

مسمى كه هلاقه کاسيه عباره عن اخراج خاصه بعده عکل
خاصه کمال \Rightarrow کتابه دله مع فنه بر سرها (ابن الهادی)

draw (data)
items = data

- هنا نناءمسى اطبع بار Consolle [دلك اطبع كامباين]

* مبى اطبخ كل فئات المنتجات \rightarrow div الفئات

`items.forEach(function(ele){`

`productContainer.innerHTML += ele.name`

لابي عالي عالي items

نامايسى كل واحد من الأجهزة

ـ سيدى من واحد الـ objects فقط

بعضى كل الأسماء ولي \rightarrow +=
مقدمة \rightarrow b6

iphone iphonell iphen1iphones

(فول) كيف يمكنني تغيير مقدمة كل بعده؟

block棘刺 لابي \rightarrow div قبل أو

سيطر كل فتح بفتح

`productContainer.innerHTML += "<div>" + ele.name + "</div>"`

is is html كود الـ ضرورة لـ كود

: (data) (جيم) | addData : أخطأة المنهج

: draw(data) \rightarrow غلط نتركه هى من ركوب خط

`window.onload = function() { draw(data); }`

لابي زول ما (العنوان)

تتحمل

لابي على ما هي؟

لانها ~~بس~~ بسع نسب (global) \rightarrow draw(data)

- انتصارات اصحاب الماءات: #اخطوة الاربعة: اصحاب الماءات
add items

```
ProductBtn.addEventListener("click", addItem);  
function addItem(){
```

الزر (add) مع ميّز ببرتّعامل ١٥ جواّر input لائم فرميّ مع اسطوّة ١٥ جواّر fun نفس الميّز.

```
→ Var productInput = document.getElementById("productInput")
```

(رسالة الخطأ ماضي input list check + ⑤)

```
    → if (productInput.value == "")
```

```
→ alert("You should enter product");
```

Console.log(data) → مقدمة (أولاً) كامبيار مقدمة بعدها
لخط المتنية

input.value c = input J1. left side access result: class

۷) سیماتک کل (id) من ای (id) می باشد

add sample input if product below will crash

لصيغة تحت عالم البيانات بـ يطلق (id) على الـ ميلود $id = 4$

id-5 ملکہ id ملکہ (name 11) میں جو رپورٹس اسے

كنت اعمل في المدرسة الابتدائية

(4) ~~length~~ \Leftarrow array and (data) ॥

حل ③ → index \neq last \rightarrow data.length = length

data[data.length] → undefined
data[data.length-1] → # { id: 4, name: "iphonel3", desc: "..." }
data[4].name [red]
data[4].desc [red]
; iephone13(id) es li cili. *

data[data.length-1].id → 4
; iephone13 id: 4, name: "iphonel3", desc: "..." *
data[data.length-1].name → iephone13

`lastId` : `Code` || `grid[i].id`
`var lastId = data[data.length - 1].id` Catch the last id

فول لو ناتاري هندي عصر دماعشي عاصري \rightarrow id
متضرر اهل كينجتون هامبرس id
قط منه افتراض لا (P1)

Var lastId = data.length ? data[data.length - 1].id : 0;
↓
إذا لم يجد id في data
فـ lastId = 0

* ملکہ عیناً رعنی عارض الی منوق catch Products عدی متوی

الخطوة ١: إنشاء مصفوفة data

```
data.push({ id: lastId + 1, name: productInput.value })
```

ـ lastId++
ـ name
ـ input.value
ـ add

ـ ① desc: "this is —"

```
Console.log(data) — (
```

ـ اذا لم يجد المربع فـ addedItem will be added

```
Var addedItem = data[data.length - 1]
```

ـ مكتوب

ـ will be added at its current position

```
productContainer.innerHTML = "<div>" + addedItem.name + "</div>" ;
```

ـ اذا لم يجد المربع addedItem will be added at its current position

ـ اذا لم يجد المربع addedItem will be added at its current position

```
productInput.value = ""
```

ـ تكون ...

* الخطوة الخامسة : بعدها احذف المحتوى
رسى ٦٦، صيغة عالميحة يحذف المحتوى

```
function deleteItem( id ) {  
    // console.log(id)  
    // data.splice()  
}
```

(بعد علامة مسافة، اختصار)

تحذف العنصر يلى عندي
وال splice لاي خلق قيمة : رقم العنصر يلى يحذف من موال data
كم عنصر يلى يحذف ؟

العنصر يلى

2 هو عبارة عن رقم

3 عبارة عن رقم ويليه عدد

console.log(id)

⇒ data.splice(1,1) → حذف العنصر لمحضه
(console.log(data)) id:1

```
// data.splice(1,1) id:1 يحذف object الذى  
// data.splice(1,2) id:1, id:2 يحذف objects 1, 2.  
// data.splice(1,3) id:1, id:2, id:3  
// data.splice(2,1) id:2 يحذف object الذى  
// data.splice(2,2) id:2, id:3 يحذف
```

ولكن عند إلأهنا لا تغير الكود لأنها أمر عالمي
والعنصر الذي يُخطّط كل سقطة هو بحثه.

نول حِرْ : يعني عزيز على الناشر حيث واجه دلائل

; div .js

لـ `innerHTML` سلسلة مُخربة:

1 ProductContainer.innerHTML += <div> + ele.name + </div>;

2 ProductContainer.innerHTML += <div> + addedItem.name + </div>;

: دلائل مُخربة

1 ProductContainer.innerHTML += <div> ele.name </div>;
مُخربة قبل العينة
dynamic
لـ `innerHTML` قبل المُخربة
\$ dynamic value : دلائل click على div

2 ProductContainer.innerHTML += <div> \$ele.name </div>;
string مُخربة قبل dynamic value \$

- كيف يمكن تفعيل المُخربة على div

1 ProductContainer.innerHTML += ' <div onclick=deleteItem(\$ele.id) >
(div) inside click داخل قواعده الساقية
لـ `innerHTML` المُخربة
\$ele.id اسني الـ
لتكون ...
67

؟ deleteItem (، ملحوظة من مع

delete Item(~~ele~~, id)

المقرر عندي (b) لازمه الـ (b) عدد ١٦٦

عندما أضيف على `div` من يابي متغير `object` يأخذ id تبعه
وتحت ذلك `div` يمكن محتوى `div` زر ما ياخذ
لاروخها حيث `object` له صفات `name` `value`
`<div>` `</div>`

Click على المربع div ، يفتح نافذة.

كتابات العصر الذهبي لـ مولانا جلال الدين رضا طلب اعجمي

Sharing L_0 , `deleteItem` ← function $\lambda L \in$

Console Methods Function JS, JavaScript

وستة ببرت ١١ مطبخ دعاء لله

~~Sir~~ ~~Dr.~~ ~~Mr.~~ ~~Mrs.~~ ~~Miss~~ _____'s name at _____ will be

• ایک دنیا میں اسی کے لئے کامیابی کا سب سے بڑا عوامی انتہا ہے۔

2

```
productContainer.innerHTML += '<div onclick="onDeleteItem('+ addedItem.id +')>$<br></div>';
```

الآن نلخصه ونرى فتح id.1 يعني الذي اخترناه

يعني نريد فتح id.1 يعني ما ياخذ على id يعني على id

data.splice(,)

أتم العنصر بـ id
index of element
عدد العناصر التي
number of elements

طريق عرضه في سطح المكتب

وهي الخطوة الأولى

Var index = data.map(function(i) {

return i.id })

foreach

طبع الباقي

- بدأ عدد id على id اخترناه اعد ايده.

.map : array على كل عنصر في arr

return i.id : id تبعه id

ارجع id على arr على كل عنصر في arr

id على arr على كل عنصر في arr

دكتن لـ id بدأ رجع id على arr

id, class & index: crests & all إذا ركّبنا صيغة

العنصر $i=1$ \longrightarrow $index = 1$

decrements id-1 \rightarrow index = 0
array will stop index \leftarrow

وکن اسایدی

indexOf(cid)

١٨١

لے لیے ہے اسی زمانہ کی ملکیت کا عین

function deleteItem(id) {
 id = عنوان ایتم
 id \rightarrow عنوان لیست element برای

→ Function deleteItem(id)

```
Var index = data.map(function(i){  
    return i.id ? } ).indexOf(id)
```

* ونکن ماذا لو کان ار index تری موجود؟ لئے اپنامائی طریقہ:

→ if (index != -1)

```
data.splice(index, 1)
```

als allein d. C. von d.

draw(data) }

console.log(data)

١٤٦٦ فتح احمد بن حنبل

لِعْنَةِ الْمُتَّهِيْلِ

ويمكن بخطأه مُخرِّي مُنْعَةٍ :

iphoneX

iphone11

iphone12

iphone13



ما هو الإلزام

: output ↪

- iphoneX

iphone11

iphone12

iphoneX } draw(data)

iphone11

iphone12

هذا المطبع iPhone3

طبع المطبع iPhone3

طبع المطبع iPhone3

* ناتي أنواعاً أسماء مع مراعى كلها فـ draw(data) د يرجع عالـ fun من نوع عالـ draw

function draw(items){

→ productContainer.innerHTML = "";

items.forEach(function (ele){

:

يسعدني كل مرة

data لا draw بد

يسعدني معاها

دعيت يعني العناصر

كود المرين كامل

//1-Setup Variables:

عوائق كبيرة

```
Var productBtn = document.getElementById("ProductBtn"),  
Var productContainer = document.getElementById("ProductContainer")
```

```
data = [  
    { id: 1, name: "iPhone X", desc: "This product is..." },  
    { id: 2, name: "iPhone 11", desc: "This product is..." },  
    { id: 3, name: "iPhone 12", desc: "This product is..." },  
    { id: 4, name: "iPhone 13", desc: "This product is..." }  
]
```

//2-draw ele

JS API

رسالة

```
function draw(items) {  
    productContainer.innerHTML = "";  
    items.forEach(function (ele) {  
        ProductContainer.innerHTML += `<div onclick=deleteItem  
        (${ele.id})> ${ele.name} </div>`;  
    })  
}
```

//3-load data: رسموا data

```
window.onload = function() {  
    draw(data);  
}
```

//4- add item:

(إضافة العناصر (المتغيرات)

```
ProductBtn.addEventListener("click", addItem);
```

```
function addItem() {
```

```
    var productInput = document.getElementById("productInput");
```

```
    if (productInput.value == "") alert("you should enter...")
```

```
    var lastId = data[data.length - 1].id;
```

```
    data.push({ id: ++lastId, name: productInput.value,  
desc: "This Product is --" })
```

```
    var addItem = data[data.length - 1]
```

```
    ProductContainer.innerHTML += '<div onclick=deleteItem' +'
```

```
(#${addedItem.id}) > ${addedItem.name} /div';
```

```
    productInput.value = "";
```

```
}
```

//5- delete item:

(التعديل على المتغيرات)

```
function deleteItem(id) {
```

```
    var index = data.map(function(i) {
```

```
        return i.id
```

```
    }).indexOf(id)
```

```
    if (index == -1) {
```

```
        data.splice(index, 1)
```

```
        draw(data)
```

```
        console.log(data)
```

lec-8 JS:

Bom

*Bom: Browser object Model

التعامل مع Document, Window و Browser في Bom

-الـ Window : هي المهمة في المكالم

Window history location screen navigator

innerWidth & innerHeight:

أربع مهامات في browser window هي anti-aliasing

*Console.log(window.innerWidth)

→ 1366 میلادی

* `Console.log(window.innerHeight)`

• إدخال فناء الرمز Console للارتفاع $\rightarrow 286$

يُخَذِّلُ الارتفاع المُتَابَع لِلرَّحْمَامِ $\leftarrow \text{innerHeight}$

Window size

كانت تجربة زوجي الأولى tab other than her husband

HTML:

```
<button id="openFB">openFacebook</button>
```

۲۳۱

```
Var btn = document.getElementById("openFB");
```

function cod1() {

```
Window.open ("http://www.comii", "", "", "width=500, height=500")
```

super function

window نمای

N[""] cip eslink

مکانیزم

عنوان

العنوان والرفاع للـ tab

الكتاب

مدخلة: لها ذات صفات عرض وارتفاع يمكنها `String` `data` `built-in` `عادي جزاً`، إذ هي المعلمات تكون.

(3) width & height () ← fun : open . الـ
 ← title () ← link () : قم .
 لـ العرض و الارتفاع
 المبتدئي : ضد عناصر

btn.onclick = Cnd1;

لابد من إضافة `click` على `button` لكي يتم تشغيل المقدمة، وذلك بـ `btn.onclick = Cd1()` .

(ad()) \subseteq (بكلمات عما يجري fun) (كـ fun)

وهي تقدّم الـ fun بعد ماريفع المفعمة وداعي مهتم بالكلمات

```
<button id="closefab">close Facebook</button>
```

Var bt12 = document.getElementById("closeFb")

function code2c();

→ x.Close()

Method of finding

3

btn2.onclick = Code2()

الـ Code هو صيغة انتافيت

لذلك في زن أقطاب المagnet ينافس fun يذكر الصلة لـ عمقلو

$\subseteq \text{Cod}1$ if res,

```
var x = window.open("http://~","~","width=500,height=500")
```

Screen:

width:

العرض المتوفر في window هي القيمة المطلوبة
`console.log(window.screen.width)` → 1366 { (width, screen)
`"Console.log(screen.width)` → 1366 { (width, screen)
 ارجوكم اهم window

`Console.log(screen.availWidth)` → 1366

[← العرض المتوفر]

[← العرض كل ما هو متاح للاستخدام]

مساحة المكتبة من نوع حفظ (bar) هي المساحة المتاحة

`Console.log(window.innerWidth)` → 1366

Window.screen.width = screen.width

لما نكتب (window.screen)

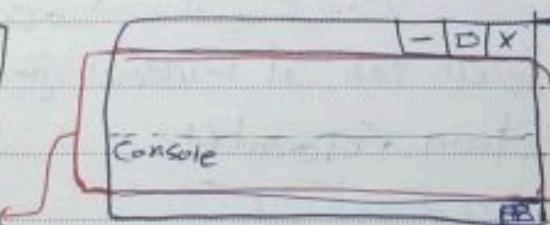
height:

الارتفاع المتوفر في window هي القيمة المطلوبة

- 1) `Console.log(screen.height)` : 768 ارتفاع الشاشة الكامل
- 2) `Console.log(screen.availHeight)` 720 المساحة المتوفرة
- 3) `Console.log(window.innerHeight)`



availHeight = body height

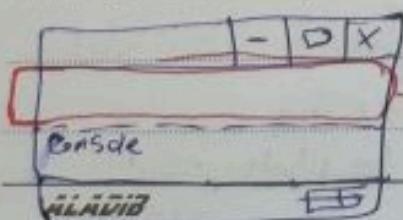


body = availHeight

ارتفاع الشاشة الكامل (768) بعد إزالة (body) : availHeight

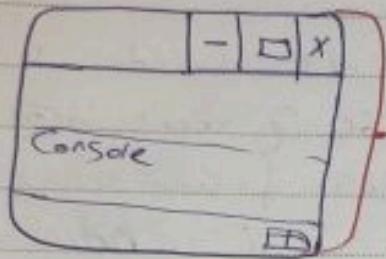
ارتفاع الشاشة بعد إزالة bars (720) : innerHeight

كل ما يزيد عن window.innerHeight : window.innerHeight



innerHeight = window.innerHeight

innerHeight



`screen.height`

مثلاً `height: 100vh` يعطى نفس الارتفاع المتوفر في الـ `availHeight` لـ `S` عندما تكون تردد الـ `height` إما `auto` أو `100vh`.

location:

*location.assign() : رفع عالمي للعنصر

facebook website \rightarrow عاشر توجيه لـ location.assign (العنوان) \rightarrow location.assign (url) tab أو window \rightarrow (موقع) link بعد ذلك \leftarrow

`location.reload();` is used by user till this line

refresh قن

* location.href :
hyper reference ↴

تحتاج إلى الـ security لـ تحقيق الـ security

نحو الاربعاء ازدياداً في عدد المدارس (location,bref,brie) \rightarrow front بـ front ملار كـ على \rightarrow مدارس

*history: (-websites) الموضع (غيره-نهاية) - تعلم الذي في صفات الموضع

(bom) browser is slow speeds.

• Eleventh Civil Page is due: history 11.

يمكنك اداً انت بالفعل تعلم و دخول في الفيسبوك

history الای تاریخ

لأنه حاكيت فاتح خري قبلها \rightarrow undefined
لـ روح كالملائكة

لے کنت خا^ند

فَلَهِي

history.forward() \rightarrow facebook
لـ بـعـد الـ عـمـل

الـ history :
ـ back() ← من history
ـ forward() → history
ـ clear() يـ clear history
ـ clearHistory() يـ clear history
ـ clearAllHistory() يـ clear all history

*navigator (nävijäätölähet) browser li

وكان الفتن يطلع عليه به نوره في مرض

```
Console.log(navigator appName)
```

بيانات المراقبة المعاصرة

الستيفل ساع (browser) تجعل

```
Console.log(navigator.appCodeName)
```

Google Chrome المفاجئ Mozilla ↵

Console.log (navigator.platform)

~~Java class~~ ← Win32
headers = 15

卷之三

Console.log(navigator.appVersion)

لـ نـسـتـهـا الـ بـرـوـزـرـ

Console.log(navigator.language)

en-US ←

المـ لغـةـ الـ بـرـوـزـرـ (الـ بـرـوـزـرـ)

Console.log(navigator.online) online & Navigator.onLine

ـ يـ عـيـ true ← الـ بـرـوـزـرـ الـ أـنـلـيـنـ

ـ يـ عـيـ مـعـنـىـ أـنـلـيـنـ بـرـوـزـرـ الـ أـنـلـيـنـ

Time

كـ ثـمـنـهـ الـ بـرـوـزـرـ الـ جـاـسـ

SetTimeout : 1 لـ بـرـوـزـرـ اـ تـقـلـيـدـ اـ أـخـمـصـهـ دـ رـجـمـهـ

SetInterval : 2 or more لـ بـرـوـزـرـ تـ قـلـيـدـ أـ كـرـيـفـهـ دـ رـجـمـهـ.

functions في الأـ بـرـوـزـرـ

Var x = SetTimeout(_____ , 2000)

لـ بـارـامـيـترـ الـ أـخـرـيـ

function () { } ← الـ تـوقـيـتـ الـ بـرـوـزـرـ

ـ يـ تـقـلـيـدـ بـرـوـزـرـ (يـ عـيـ بـرـوـزـرـ دـ رـجـمـهـ)

(المتـ بـرـوـزـرـ)

Var x = SetTimeout(function () {

 alert("Pay → ") } , 2000)

لـ بـرـوـزـرـ بـ تـقـلـيـدـ دـ رـجـمـهـ دـ رـجـمـهـ دـ رـجـمـهـ

* Var x = SetInterval(function () {

لتـ كـونـ ...

خطاب لغوي هم يقبل الـ 25 في login حسابي ياتا يعني لها
يعني ما هي ظرفية alert الـ **clearTimeOut** :
clearTimeOut: → يوقف

$\Rightarrow \text{clearTimeOut}(x)$

لهم مات بغير فلاح

لوار `ClearTimeOut(x)` \Leftarrow `user` \Leftarrow سجل دخول \Leftarrow مکانیزم \Leftarrow اعماق

* Var y = setInterval (function() {

```
alert("Do you want to login?") , 2000)
```

لـ يعنـى كـ مـانـيـن بـ طـلـع الإـعلـان .

ClearInterval(y)

- إذا سمعت هر قنطرة هاد الماء من نهر

Dom II (rec-GJS) is LD at -

Create

```
<div class="Container"> WebDevelopment </h2>
<h2 class="title1"> -- </h2>
<h2 class="title2"> -- </h2>
<h2 class="title3"> -- </h2>
<h2 class="title4"> -- </h2>
```

JS بـ **createElement** و **tag** (واعرضيه بيانات وعدينه بـ **Dom**)

Var d = document.createElement("div") div object

id + attr لیکن سلسلی از **div** ها

`d.id = "div10" → id 属性
d.className = "div10" (attr + style) class 属性`

setAttribute("id", "PGE",

```
d.setAttribute("id", "div10")
```

```
d.setAttribute("className", "div1")
```

- اف اس بی اکتب جواہ کیتے ؟

```
<div> </div>
```

```
2.innerHTML = "islam"
```

Console.log(d) → <div class="div1" id="div1> islam</div>

مُؤْمِن (text copy) text I create ~~set so *~~
~~div~~

```
Var txt = document.createTextNode("mohamad")
```

عنوانی متن را خواهد بود

d. appendChild(txt)

لهم اصلحني

تم الالتفاف حول

$\Rightarrow \text{Console.log(d)}$ →

لتكون ...

• HTML الـ div هار **نـصـيـفـهـ** *

وـعـدـتـهـاـ لـمـاـ مـلـأـتـهـ مـقـبـلـهـ ؟ مـقـبـلـهـ

title2 مـقـبـلـهـ Container

JS **لـازـمـ** لـفـقـهـ التـغـيـرـاتـ

Var ct = document.querySelector(".Container")

Var t2 = document.querySelector("title2")

→ ct.insertBefore (d, t2) **لـصـيـفـهـ**

ct دـيـلـيـلـهـ لـمـاـ مـلـأـتـهـ مـقـبـلـهـ

d دـيـلـيـلـهـ مـلـأـتـهـ مـقـبـلـهـ

t2 دـيـلـيـلـهـ مـلـأـتـهـ مـقـبـلـهـ

ct
Console.log()

```
<div class="Container">
  <h2 class="title1"> Web </h2>
  <div id="class">
    islam mohamed
  </div>
  <h2 class="title2"> Web </h2>
```

Var & let & Const

متقارن بين const و let و var

أوهم مقارنة

1-scope 2-reassignment 3-redclaration 4-hoisting

Scope:

Var:

Var x=100

Console.log(x) → 100

Console.log(window.x) → 100 => Var → global

window ليس كذلك أو ليس معرفة

→ function Code1()

Var x=90

Console.log(x)?

Code1() → 90 ✓

→ function Code2()

Var x=80?

Console.log(x) →

global scope

غير معرفة

العلم اتي قمت بتعريفه او x(Var)

غير معرفة لماذا؟! عندما اعرف او x(Var) المثل

هي function scope لا

↔ ? - ?

local: ← fun او x(Var)

Var : global scope + local (if) function scope

function if 8,10 ↪

function scope ↪

let ↪

let x=60

Console.log(x) → 60

Console.log(window.x) → غير معرفة / undefined

(عمر سرقة window) ↪ (let if)

let: block scope

Const ↪

Const x=80

Console.log(window.x) → undefined

⇒ Const: block scope

local scope ↪ block scope ما يعرق بين ↪

Var : global scope + local scope or function scope

let: block scope , }

ما يعرق على (window) ↪
(النافذة العامة المفتوحة)

Const: block scope.

2- reassignment:

إذاً يمكن إعادة

Var (C)

reassignment ممكن

Var X = 60;

Console.log(X) → 60

X = 120

Console.log(X) → 120

g

→ Var X = 60;

X = 120;

Console.log(X) → 120

Let (C)

reassignment ممكن

let X = 50

X = 70

Console.log(X) → 70

Const (X)

reassignment ممكن / not for assignment

C.

Const X = 50

X = 40

Console.log(X) → error

لتكون...

3] redeclaration:

(✓)

$$\text{Var } x = 40$$

Var $x = 70 \rightarrow$ هذه هي إعطاؤه قيمة جديدة لـ x .
اعادة تعریف مجدد لـ x .

Console.log(x) → 70

~~let~~ (x) not for redeclaration

let $x = 30$

let $x = 70$

`Console.log(x)` → error → معرف بالجهة قبل (x) .

Const (X) not for redeclaration

Const $x = 90$

Const $x = 50$

Console.log(x) → error

4- hoisting: ↗)

`Console.log(x)` → x is not ...

الـ(٦) غير معرفة بغير

Var 1- `Console.log(x)` → undefined

$$2 - \text{Var } X = 50$$

\Leftarrow مطابقاً لـ Var ١، نعم المرفع

$\leftarrow \text{R} \rightarrow \text{باید}(x) \sqcup \underline{\text{Var}} \rightarrow \text{غمد مطمئن از}$

Declaration

Var X

assignment

الريلفاج اول سطر بالكود ويطلق على (نبع الكود)

مقدمة ← ويسليط الضوء على المقدمة للخ سار حتى يظهر نفسه.

75 - Var X = 40

عندما

Var X // 1 ←

ذلك اذا طبّقت خطوة المثلث 70 Console.log(x) يعني ماحصل على undefined

لما يجري خطوة المثلث ← مقدمة ←

مقدمة اضافية ← 75 ← 40 ← Console.log(x) ← نزع بخطوة المثلث

⇒ Var → window → global

⇒ declaration (تعريف) يخص كل المواقع

عما يجري في نافذة بعد assignment (اعطاء)

تعريف المتغير

Var X → declaration (تعريف) الاول لـ Var

X = 50 → assignment (اعطاء) ass

لـ اعطاء المقدمة

global ← Var ← window ← اعادة ال ← اعادة ال ←

Variable (متغير) undefined (غير معروف) اي

اي تم ايجاد declaration على ولكن لم يتم اعطائه بعد.

ايجاد assignment (اعطاء) له.

لتكون...

let: (X)

لا يدعم

console.log(x) →

let x = 50

assignment side effect (x) before access لا يمكنه فعل.

const: (X)

لا يدعم

console.log(x) →

const x = 50

	Var	let	const
scope	global + function @ local	block	block
assignment	✓	✓	X
redeclaration	✓		X
hoisting	✓	X	X

متالعما

1 Var x=15

2 function Codec()

3 if(false)

4 Var x=40 ; → declaration x

5 console.log(x) ? if معا او

6 - Codec() → undefined

اركوان

مالذ؟ ←

الذر if(false) ← يعني لا تجده بوجة ذلك علني

الظر (3) و (4) ما ع نقدرهم ← يمكن او declare Var

(5) Var x=40 في ادراك ← Redeclaration

(functions scope of) fun (2) داخل او ← ذهب اي اسطر

Var x ← وكتبته هنا

اذا تحقق اذا يجيء ← if

Var x ← لم تتحقق حمل سيرى او (40) وكتبه سيرى او

ما الصيارات اللى استخدمناها لفول الانعة :

funscope + hoisting :

↑ hoisting وكم هو بس ايه الرصبة

ازنون داخل او function Local scope

declaration

if(false) →

لاتجده

if(true) →

ادخلي دفعه

```
let y = 30  
function Code1() {  
    if (false) {  
        let y = 90;  
        console.log(y);  
    }  
    Code1();
```

مقاييس

أكواب [30]

منبر انت انت let

يوجد سؤال هام هنا في السبيل (التجهيز).

* مصطلحات (حالة للمعنى):

Var Code1: 15; → Variable assignment

function Code1() { → function declaration
 console.log("ahmed");

Console.log (Code1) → # 15

Console.log (type of Code1) → # number

→ Variable assignment > function declaration

أو

Var Code2 → variable declaration

function Code2() → function declaration

Console.log ("a");

Console.log (Code2) → a

Console.log (type of Code2) → function

Arrow function

ـ arrow function : تستخدم لكتابة الدالة بخط مختصر :
ـ من قبل كاتب function ويعززها (=>)

$$\text{Var } X = 50$$

→ Var code1 = (x) => console.log(x)

Parameter ↴

→ Var Code1 = (x,y) => Console.log(x+y)

Console.log(10,30) → 40

ec-script - 6

$\rightarrow \text{var Code3} = (a, b) \Rightarrow a * b;$

عکس اجتماعی الداله قبل traditional function

فی الْمَلَکَاتِ

JL_o hoisting

تعریفی \Rightarrow مفهومی توصیفی

Code1()

function Code1()

Var x = 5)

```
Console.log(x) }
```

جبل عادی

التعريف المبكر لبيانات المتغيرات Hoisting في أو expression function

Var x = function() {
 Var y = 19;
 console.log(y);
 x();
}

في ترتيب قبل ترتيب fun

Var x; declaration Without assignment
Var x; Redeclaration without assignment

* Var y = 9; declaration with assignment

Var y = 20; Redeclaration with assignment

with reassignment : له ميزة

لأنني بعد ذلك أقوم بـ redeclaration reassignment overwriting value.

* Var y = 20 declaration with assignment

y = 50 reassignment

lec-10 JS

Default Parameters

Number, String, function

ex: x
function(a,b){
 Console.log(a*b)
}
 $x(5,10) \rightarrow 50$
Arguments $\rightarrow a=5, b=10$
Parameter

default parameters فی (== new **call** parameter)
لی **this** default value افتراضی parameter \rightarrow يعنی دخله ای **default** \rightarrow دخله ای **default** \rightarrow دخله ای **default**.

function $x(a, b=20)$
Console.log(a*b)

$x(10, 2) \rightarrow a=10, b=2 \rightarrow 20$
 $x(10) \rightarrow a=10, b=? \rightarrow \text{default} \Rightarrow 20 \Rightarrow 200$
parameter دخله ای

number من المفع Parameter ها هیا *

* (من زمان) کافی نیست های اخیر

function $x(a, b)$

$b = b || 20$

Console.log(x)

باید ساخته باید این ترتیب باشد

(20) وادعا میشود

لیکن صارعیم برخواهد که همه یعنی مفروض

: string في المقدمة default step من

* Function `x(a, b='Customer 1')`

```
Console.log(a ,b ) }
```

`X(25, "ahmed") → 25, ahmed`

X(25) → 25, customer1

* function x(a="Customer", b=20){

Console.log(a, b)?

X() → Customer, 20

X("Sally", 22) → Sally, 22

: fun ← default definiتی اع-

function Code1(x = Code2())

Console.log(x) }

Code1(500) → 500

Let $y = 0$

function Code2() {

$$y + = 1 \quad // y = y + 1$$

```
return y }
```

Code1 () → 1

مايكن بارفون
Codecs \Rightarrow default

Code 1() → 2

Code1() → 3

بيانات عالمية مرجعية

Code1(500) → 500
500 = X

Code1(40) → uo

Code1() →

(u)
✓

حل احواب : دل (1) دل (2) دل (4)

X = uo

لما يدخل حل ماكينت فتحي مدخله اون قبل كان
متى تراست عل ① End Function

← X = Code2() اسلوب fun هى ←
• ① سجن عالي كلها سجن

Code1() → 5

Code1(20) → 20

Code1() → 6

: select ↴ سؤال *

```
function Code (name, greeting, message=greeting + "+" + name)
    return [name, greeting, message]
```

```
Console.log(
    Code("ahmad", "welcom", "welcome ahmed in our web"))
    name"           greeting"           message
```

```
Console.log( Code("ahmed", "Welcome"))
```

Welcome ahmed

مايك الذهبي مطهير در

Spread Operator

مايك الذهبي مطهير در

Const arr1 = [50, 60, 90, 120]

Console.log(arr1) → [50, 60, 90, 120]

Console.log(...arr1) → 50 60 90 120

فرزغ البيانات من اد arr1 عرضها ياتي . يفرغ جسم arr1
ويعطي العناصر كلها اد arr1 ويفصل الترتيب الذي كانت فيه .

مايك الذهبي مطهير در

عند بيانات متراكمة ← بيانات ←
عند عرض ← متراكمة ←

Const obj1 = {

City: "Syria",

age: 25,

fname: "Islam", }

ويأخذ obj2 حفظ مكتوب مثل الای تقبل بشرطه عليه
← يكتب بيانات اد obj2 بعد ← وباقي

Const obj2 = {

...obj1,

lname: "mohamad" }

`console.log(obj2) → {city: 'syria', age: 25, fname: 'Islam', lname: 'Moh'}`

دُرُس . إِنْتِلْجِي دُكْ ← Destructions : react if دُوك

Destructions

في كل فئة تأخذ القيمة المقابلة لها العنصرة الـ (i) في دالة
Const arr1 = [10, 20, 30]

كثافة والقيمة المقابلة
`Const [a, b, c] = arr1`
`destructions` معرفة $\rightarrow [a, b, c] = [10, 20, 30]$
`console.log(c)` $\rightarrow \boxed{30}$

* Const [, , c] = arr1

Console.log(c) → 30

* `Const [arr1, arr2] = arr1`
array متغير متعدد مكون من العنصر المقابلة لـ العنصر `Console.log(c) → [20]`

```
* Const Person1 = {  
    name : "islam" ,  
    age : "26" }
```

```
Const { name: x ,  
        age: y ,  
        City = "mansoura" } = Person1
```

Console.log(age) → ~~undefined~~ / error

١٩١٦

Console.log(x, y) → islam, 26

: obj دلالة

Const { name: x, age: y, City: "Mansoura" } = { name: "islam",
age: 26 }

name دلالة name دلالة
islam ← Value دلالة دلالة ← (x) ← variable دلالة
islam ← x دلالة دلالة ← Key ← name دلالة

ذيل المحتوى مفهوم دلالة

let [name1, name2, ...rest] = ["esraa", "amany", "ahmad", "mohamed", "ali", "asmaa"]

→ Console.log(name1) → esraa

Console.log(name2) → amany

Console.log(...rest) → ahmad, mohamed, ali, asmaa

Console.log(rest) → ahmad -

let [name1, ...rest, name2] = ["asraa", "amany", "ahmad", ...] }
: عندما

Console.log(...rest) → Error

لأنه لا يكفي أن تكتب rest

لأنه يمكن منك عرض rest ← (→) قبل الباقي تغير rest كلها.

ولذلك rest هو الباقي ، ← عدد rest --x

Rest Element must be the last Element

Spread Operator must be the last element

let [name = "islam", name2 = "mohamed"] = [fname = "ahmad"]
: لأنه *

Console.log(name, name2) → ahmad mohamed

This

يُعمد لـ this : *

Window

Console.log(this) → window

this.x = 500 → (window.x = 500) → 500

Console.log(this.x) → 500

Console.log(window.x) → 500

window تعود على this في هذه الحالات .

لتكون ...
سيون ...

2] This inside fun: `console.log` is `this` it calls `is`.
`=window`

`function x() {
 return this ?`

`Console.log(x())` → `window`

• `window` its key is `fun` `is` `this` it.

3] This inside method inside obj → object class

`Var person1 = {
 fname: "islam",
 lname: "mohamed"`

object `is` `function` it [call]

method : `is`

function inside obj Called

method

`fullName: function x() {
 return this.fname + " " + this.lname;`

`obj` `this`

`obj` method

fun = > > >

تعدد على `obj`

`Console.log(person1.fullName)`

`f x()`

function `help`

function it `return this.fname + " " + this.lname`
it's own fun it `it` it's own calculation

`Console.log(person1.fullName())` → islam mohamed

4- this = element.tag HTML child of tag سعديه اس اف this

Call

Var obj1 = {

name: "islam"

Var name = "mohamed"

function x() {

return this.name

}

Console.log(x()) → mohamed
window.function() this

object needed

Console.log(x.Call(obj1)) → islam

x.fun Call call

fun, Egypt ← obj1

Emp1 ← obj1

window.Egypt ← this

Console.log(x.apply(obj1))

call

function will be used fun handle job given

ل تكون ...

Template string

* important

(JS into HTML) نوافذ المتصفح

Const x = `backtick

```
<div>
<span> this is ux7 </span> </div>
```

string

(react JSX) JS does HTML code: JSX

Console.log(x) → <div> this is ux7 </div>

Const x =

```
<div>
<span> this is ${ux7} </span>
</div>
```

Console.log(x) → <div> this is 28 </div>

Var Car1 =

Call is JSX *

Car Model: function()

return this.CarName + " " + this.model

{

{

Console.log(Car1.CarModel) → undefined

Var Car2 = {

Car Name: "BMW X6" ,
model : "2021" ?

Console.log (car1.CarModel . Call (Car2)) → BMW X6 2021

Car2 ↳ this ↳ car2

سخنرا عندها نريد انة معلمات مخزنة قبل بـ **Call** ↳
معلمات علية

Const C = {

name: "BMW" ,

model: ["2020" , "2021"] ,

method ↳

result() { object inside ↳

Console.log (this.name) } ,

result2() {

this.model.forEach (Function(model))

Console.log (this) } ;

عندها يدخل كل

عندها يدخل كل علامة

عندها يدخل كل علامة

دعا عندها رفع **forEach** كي وظيفه ()

عندها يدخل كل item عندها المعمليات ↳

function () { يقىء على !!

لتكون ...

Car1.result2() →

object ۱) clip method بیب دهند
۲-this فی نفس اور window class (this) ۱)

→ result2()?

die [this] model. forEach --- {

مکالمہ علیہ

Console.log(this) → window

Card-object II

Wieder aus

result2() method in C++

دِسَّهُ دِسَّهُ this is obj to decide this decision

تعود على نفسك (obj) this قبل المفعول بالفتح

this is (this) ✓

see end of data in [GitHub](#)

All this is due to the fact that this decision is critical.

: art de faire une chose.

بِعْدَ طَرِيقَتِي :

٢- result2: يابي موفق الـ (result2) عكتبون مثل ماكاله (يعلمونه)
 result2()

`Var x = this`

X. model.forEach(function(model))

~~Consolidation~~(~~X~~) { } , }

2

result2()

الطريقة الثانية طلابيـة:

```
this.model.forEach(function(model){  
    console.log({} , this);  
    obj不可缺少});
```

`Car1.result2() → 2020-2-21`

method `is` function is this class ~~as~~ ^{is}.

method 1, in this functional and this is an in-
order of this is the order

Type of errors

(one will spell like usg.)

I] Syntax error = Parsing error:

کے ایجاد کا کد

مختلف بعض) \leftarrow (نحو اخطاء ولكن له أئمه

Syntax is a + Melodic JS JL

Parsing ~~is~~ + ~~not~~ React JS

function $\{ \}$

الامتحان حاملاً

عنهما

2] runtime error:

مظاہر التضییی

مثال: عندك fun تبادرك عن خيال معيّن لا تتحقق **فقط** معيّن
وهذا الذي ياباني به تبادرك عاليه (مقدار موحد عندك) \rightarrow دفعك
عمر تيفوند زعيم بغير انتها.

(منا) : تناولت عن خنزير معمد صور

3. logical errors:

فكرة مانع عارف

مثال: بدر تعلیم شی منظر سیم فرمیم رالہ ادھم عوین
طبلے حافظہ منظری۔

٤) reference error: عدم تنادی عنوان خوب (متن صرف)

متغير: متغير (X) هو صيغة ملائمة عامل Δ

١٩ runtime error و reference error بین المفہومین

function() { x + 4 }

انواع معرفی (x) عکس برای window

اى مي تاد بيت عاليه فنه في مت مفهوم
- من مفهوم دة فنه (function).

ـ runtime : هناك ثانٍ : عندى مدة عمليات وعمم متول الناتج يلى
ـ يطلع اهمى عليه نسبة معينة \rightarrow والناتج ما يطلع عنى \rightarrow القيمة
ـ مارجى تطلع \rightarrow المظا أو المركبة (أنتوى التقى). (نسبة القيمة من مجموعة)

Try Catch

إذا لم يتمكن من المتابعة فـ (Catch)

Code for runing
try { function () {
 }
 }

Catch (e) { لـ try catch
 سيتم دخول معاودة try
 Catch
 }

try { (function () {
 console.log('fatima')
 })
 }
 }

Catch (e) {

Console.log(e.description) {

رسالة خطأ تم اكتشافها
ارسال للـ backend

Strict mode

ـ منه تفاصيل و معايير : strict mode ـ (مدى المعايير)

Ecma Script-6 (JavaScript) لـ اجهزه

"use strict"

تعريف المتنبرات (المطبخ)

$$\boxed{10} \quad x = 50 \quad // \text{JS}$$

Console.log(x) → 50

(أضفت افتراضياً Var

"use strict" //ecma script-6

X-50

`Console.log(x) → error`

Var $x = 50$

Console.log(x) → 50

میں کو تعریف کر
global var
بکھر کر
اذا اے کتبخانہ
تھوڑی بسا فرمائیں
is strong جا

function () ()

function Codecs {

X=100

Consequently $(x)^3$

Code() → 100

"use strict"

functioncadency

$x_2 \neq 0$

Console.log(x) ?

`Code()` → error (من متفرقة)

3 delete:

Var x = "aya" //JS
delete x ; →

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

"use strict"

Var x = "aya"

لیک مڈھل

`Delete x ;` → Syntax error : Delete of an unqualified

تعريف (identifier) في режим strict mode.

١١- (delete و strict mode : Strictmode
strict mode و delete لـ مستر

strict mode (حالة صارمة) هي عبارة عن قيمتين: `global` (عامة) و `local` (محلي).

TS Jee vat

4] duplicated Parameters

1185:

function Code(a,a){}

Console.log (a+a) }

Code (200, 7) →

→ 14

a-7

Console.log(500, 100) → 200

$$a = 100$$

فلا يرى ما كان تاده . الخامسة مرئي كهيا في المائية .

علم المنهج أو المعرفة الـ (أ) ماهيّة المعايير \rightarrow (رسائل عاشرة للفصل السادس)

٤٣١ كان المأذون في إتمام: يأخذ بالرتب

الأخضر

"use strict" //ecma script-6

function (a,a){

Console.log(a+a)?

Code(500,100) → SyntaxError : Duplicate Parameters name
not allowed in this Context.

لأنه مسمى

النيدل Strict mode يعمم بالغاء الباقي كل ما إذا خطأ
- (duplicateParameters) (Duplicate Parameter Error).

Array helper method

* كل وحدة منها جاينر متعددة تجري على المảng

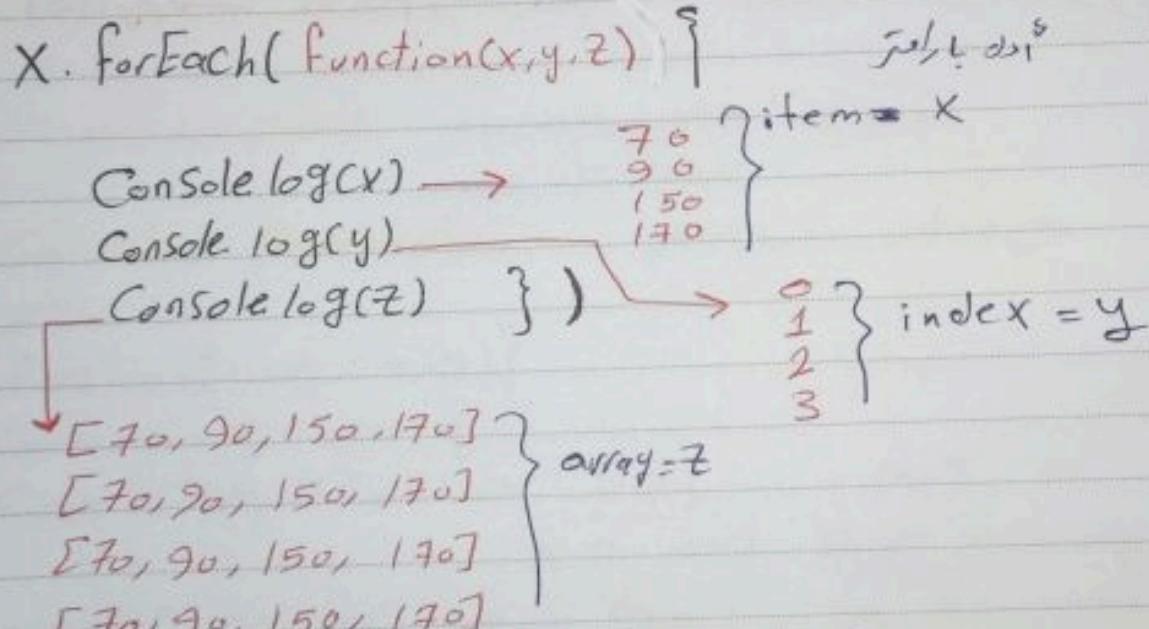
1) **forEach :**

تعود ب值得一

(single Value)

لوعندي مثلاً (3) و اقيس كل الـ forEach على مخزن array .
لوعندي مثلاً (3) array (index) item .
arr.forEach(item, index, array)

: Var x = [70, 90, 150, 170]



* (x) عندي ← forEach كل عنصر في عناصر المخزن
ذلك من بابه الاولى جلسوا (y, item) زورون معبعينه (z) هرق
عنصر الثاني (90) ونفعنا (function) وهو يخلي مخزن محتوى (x)
و بالاعتراض (x) هو item منه ..

* ForEach : (y) ← غيرق فالعنصر (y) مرات ويكمل وحة . كيل
عنصر يخلي مرق كل (لوعندي مثلاً (3) عنصر) من يخلي مرات .

لتكون ...

أرجعى بالمعنى عدّهات = عدد العناصر المعرفة
لأن (3) بـ array و function على كل عناصر
لعمده ورقيه لبيه.

2] map

array + func → array (أيضاً array)

ex: Const arr1=[10, 20, 30, 40]

Const double=arr1.map(function(item){
return item*2 })

Console.log(double) → [20, 40, 60, 80] ← array

الخطوة الأولى (عند map):
10*2 ← نتائج ← (10)

20*2 ← نتائج ← 20

30*2 ← نتائج ← 30

40*2 ← نتائج ← 40

• arrow function يكتبونها كـ map ← func.

arrow function يكتبونها كـ map ← func (ecmaScript-6)

Const arr1=[10, 20, 30, 40]

Const double=arr1.map(item) => item*2)

map ← func
item ← param

لارفع كـ func
ولا نضع كـ func

Console.log(double)

ex2:

هذا المثال موجه للأطفال
أfreight أرقام ونهايات الطبع ربما:

```
let numbers = [1, 4, 9, 16, 25]
let roots = numbers.map(function(num) {
    return Math.sqrt(num)
})
```

Console.log(roots) → [1, 2, 3, 4, 5] //array

Note: map doesn't change original array!

Console.log(numbers) → [1, 4, 9, 16, 25]

ex3:

let arr1 = [{key: 1, value: 500}, {key: 2, value: 600}, {key: 3, value: 700}]
key → key: 1, value: 500 → index of 0
key → key: 2, value: 600 → index of 1
key → key: 3, value: 700 → index of 2
arr1 [{1: 500}, {2: 600}, {3: 700}] → array
Value also key ما يخرب كل

let x = arr1.map((obj) => {
 let y = { value: obj.value }
 y[obj.key] = obj.value
 return y
}): [{1: 500}, {2: 600}, {3: 700}]

Console.log(x) → array of objects: [{1: 500}, {2: 600}, {3: 700}]

لتكون...

٣- Filter : array -> يعود + معرفة الأشياء التي تلبي

filter doesn't change the original array.

Ex1: Const arr=[1,2,3,4,5,6,7,8,9]

نريد أن يتم فاتحة المعرفة وإرهاق الأعداد العزبة:

```
const odds = arr1.filter(function(item){  
    return item % 2 == 1 })
```

Console.log(odds) → [1, 3, 5, 7, 9] // array

- ١٠) `filter` : يرجع `array` فيه عناصر `array` ذات `log1` اى `True`
- ١١) `map` : يطبق `function` على جميع العناصر
- ١٢) `reduce` : يجمع عناصر `array` في مجموعه واحدة

```
Const evenNum = arr1.filter(item => item[1] == 0)
```

Console.log(evenNum) → [2, 4, 6, 8]

Ex2: Const names = ["amany", "osama", "Sally", "ahmad", "esraa",
"islam", "zaki", "aya"]

لذلك في الـ array يمكن إدخال الأسماء التي تم ترتيبها في

لَا تُرِيدُ حنْ (٤) أَمْرٌ :

Const X = names.filter(word => word.length < 5) الخطوة

↓ كل عبارة المصنوعة

Console.log(X) → ['Zaki', 'aya']

4- reduce(): signal value + ما يadder

ex Const arr1 = [1, 2, 3, 4, 5, 6, 7, 8, 9]

↓ arr1 العنصر الأول يتم الجمع على

Const sum = arr1.reduce(function(acc, current))

↓ القيمة الراهنة القيمة الحالية

return acc + current }

Console.log(sum) → 45 // single Value

reduce (العنصر الراهن, الرقم) يعرف عدد

: عن أول مرر (العنصر الأول من المصنوعة) (1) خطوة

1) Current = 1 , acc = 0 default $1 + 0 = \boxed{1}$

2) Current = 2 , acc = 1 $\Rightarrow 2 + 1 = \boxed{3}$

3) Current = 3 , acc = 3 $\Rightarrow 3 + 3 = \boxed{6}$

4) // = 4 , acc = 6 $\Rightarrow 6 + 4 = \boxed{10}$

5) Current = 5 , acc = 10 $\Rightarrow 10 + 5 = \boxed{15}$

6) Current = 6 , acc = 15 $\Rightarrow 15 + 6 = \boxed{21}$

7) Current = 7 , acc = 21 $\Rightarrow 21 + 7 = \boxed{28}$

8) Current = 8 , acc = 28 $\Rightarrow 28 + 8 = \boxed{36}$

9) Current = 9 , acc = 36 $\Rightarrow 36 + 9 = \boxed{45}$

إذا أردت اعطى قيمة معرفة لل acc [Note]

: default

Const sum = arr1.reduce(function(acc, current){
return acc + current; }, 100)

Console.log(sum) → 145 //Value

أنت هنا لأنك مرر على دخل عصري acc = 100

طريقه امثل [] ، وصيغة في reduce

الذوكي : fun ← logic

الثانية : العنصر المسندة لل

reduce(fun , 100)

باتخذ قيمة لـ العاشر بتغير

لليارات الأول خلا

الذي هو acc



بعد آخر الدرجات بـ (6) مواد عن كل طبق واحد على دار الحسنه

. reduce first

ex2:

let players = ["shenawy", "zawad", "samson", "tao",
"mohamed", "zizo", "islam"]

بيانات اللاعبين يتم تمريرهم على دار array wi

كل سعفه قاعدة عامة باسم الدالة filter method حيث تتطبع
أي دالة على أي مصفوفة تمر دائرتها في الدالة

function code(arr, x) {
 array, arr[] ↳ اعمد الامر سبيع عنده
 ↳ arr[] index كل عناصر
 return arr.filter(function(item){
 return item.indexOf(x) != -1 ;
 ↳ if index != end
 ↳ if item != null
 })
}

```
Console.log (Code(players, "S")) → ["shenawy", "Samson",  
"islam"]
```

Console.log(code(players, "z")) → ["zozo"]

5] Some: false أو true يعود →
array داخل سطر search : some []
كذلك الشكل يعود بـ true إذا وجدوا مترافقاً
false : يعود بـ
filter ← إذا أردت فقط

```
Const x = arr1. Some(function(item){  
    return item > 8 })
```

```
return item > 0 })
```

Console.log(x) → true

return item > 2

سؤال انترفيو: لعمضي بالشرط:

عما يرجع من المربع الذي يتحقق منه الشرط؟

index = 2

[1, 2, 3, 4, ...] is some if
[~~1, 2, 3, ...]~~] X X [2]

يتحقق من الشرط

فجواب true + true crew some if

6 **every**(*array, callbackFunction*) *false* or *true* → يعمد

ex Const arr1 = [1, 2, 3, 4, 5, 6, 7, 8, 9]

Const x = arr1.every(function(item){
return item > 2 ? })

Console.log(x) → false → **لأنه ليس جميع العناصر**

سؤال: عما يرجع من **every** إذا لم يتحقق الشرط؟

index = 0

[1, 2, 3, ...]

[0, 1, 2, ...] index

false + false ← الشرط غير متحقق
ولا يكمل

return item > 0

Console.log(x) → true

سؤال: إذا لم يتحقق الشرط في العنصر رقم 8

index = 8

لقد تعلم

7] find : يبحث عن logic مصنف و إذا لم يجده لا يرجعه وإن وجدته

w3school

نحوه

Const Code = [{id : "a"}, {id : "b"}, {id : "c"}, {id : "d"}, {id : "e"}]

: (C) له قيمة

غير المبتدأ لا يكتبه

Const X = Code . find (function(item) {
return item.id === "C" })

Console.log(X) → C

طبعاً (2) ~~يختلف مثلاً من some + find : note~~ عند ما يدخل

index

some → true / false

find → برمجة العناصرية لـ find

8] findIndex : يعنى ترتيب (index) العنصر الذي يبحث عنه

Const Code = [{id : "a"}, {id : "b"}, {id : "c"}, {id : "d"}, {id : "e"}]

Const X = Code . findIndex (function(item) {

return item.id === "C" })

Console.log(X) → 2 الترتيب رقم 2 index

find : إذا وجدت سمات سearched

findIndex : إذا وجدت ترتيب سearched

9] sort : لستنام array لترتيبها

let names = ["islam", "mohamad", "ali", "sayed"]

names . sort();

Console.log(names) → [ali, islam, mohamad, sayed]

لتكون ...

• (2) الـ a من المُعْطَى مـا يـُجـدـي (أـنـجـمـنـا)

Note: Sort Change the original array

الـ sort يـُعـدـ عـاـلـيـةـ مـفـعـلـةـ الـ صـلـيـدـ

ex2: Sort(a,b) مـا يـُتـابـعـ مـا يـُتـابـعـ

كـلـ كـلـ كـلـ كـلـ كـلـ كـلـ

1- Var nums = [4, 200, 30, 270, 18]

2- nums.sort(function(a,b))

3- return a-b ?

4- console.log(nums) → [4, 18, 30, 200, 270] // رـجـاعـيـهـ

كيف عمـ الفـعـلـ ؟ اـعـبـرـ (a) ← الـ رـمـمـ الـ اـنـوـرـ

الـ رـمـمـ الـ تـائـيـ (b) ←

1) a=4 b=200 → a-b = 4-200 = (-) (أـنـجـمـنـا، لـجـلـجـلـ)

[4, 200]

2) a=200 b=30 → 200-30 = (+) (أـنـجـمـنـا، لـجـلـجـلـ)

[4, 30, 200]

3) a=4 b=30 → 4-30 = (-) ✓

4) a=30 b=200 → 30-200 = (-) ✓

5) a=200 b=270 → 200-270 = (-) ✓

[4, 30, 200, 270]

6) a=270 b=18 → 270-18 = (+) X (عـكـسـ لـعـلـيـهـ)

[4, 30, 200, 18, 270]

7) a=4 b=30 → 4-30 = (-)

a=30 b=200 → 30-200 = (-)

a=200 b=18 → 200-18 = (+) ✓

[4, 30, 18, 200, 270]

$$a=4 \quad b=30 \rightarrow 4-30 = (-) \checkmark$$

$$a=30 \quad b=18 \rightarrow 30-18 = (+) \checkmark$$

$[4, 18, 30, 200, 270]$

$$4-18 = (-) \checkmark$$

$$18-30 = (-) \checkmark$$

$$30-200 = (-) \checkmark$$

$$200-270 = (-) \checkmark$$

فأعنى : عندما تمسك بأعماق النيزك أثناء الترتيب في array ، فالخطوة من البرائحة تغير الأعماق .

* الترتيب التنازلي :

ex3:

Var nums = [4, 200, 30, 270, 18]

nums.sort(function(item){

return b-a })

Console.log(nums) → [270, 200, 30, 18, 4]

نفس كل شيء فقط العرق (b-a) .

نمسك a=4 b=200 → b-a = 200-4 = (+)

[200, 4]

a=4 b=30 → 30-4 = (+) نمسك دفعات العرق

[200, 30, 4]

a=200 b=30 → 30-200 = (-) \checkmark

a=30 b=4 → 4-30 = (-)

a=4 b=270 → 270-4 = (+)

[200, 30, 270, 4]

a=200 b=30 → 30-200 = (-) \checkmark

a=30 b=270 → 270-30 = (+) \checkmark

[200, 270, 30, 4]

لتكون ...

$$a = 200 \quad b = 270 \rightarrow 270 - 200 = (+)$$

[270, 200, 30, 4]

$$a = 27^\circ \quad b = 200^\circ \rightarrow 200^\circ - 27^\circ = (-) \quad \checkmark$$

$$Cl = 200 \quad b = 30 \rightarrow 30 - 200 = (-) \quad \checkmark$$

$$a = 30 \quad b = 4 \rightarrow \quad 4 - 30 = (-) \quad \checkmark$$

$$a=4 \quad b=18 \rightarrow 18-4 = \underline{\underline{(+))}}$$

[270, 200, 30, 18, 4]

$$x = 270, b = 200 \rightarrow 200 - 270 = (-) \checkmark$$

$$2 - 200 \quad b = 30 \rightarrow 30 - 200 = (-) \quad \checkmark$$

$$b = 18 \rightarrow 18 - 30 = (-) \checkmark$$

$$\alpha = 18 \quad b = 4 \quad \rightarrow \quad 4 - 18 = (-) \quad \checkmark$$

مختصر عند تعيين كل الابعاد بات سانت ← تكون الـ array مربعة.

To] reverse :

ترتيب عناصر المصفوفة يمكن عمله
وهي تعرف أيضاً على الـ array الأحادية

```
let names = ["Car", "anas", "islam", "ba"]
```

names, reverse()

Console.log(names) → [islam, car, barana]