Advanced JavaScript Summery

Mebo

#######################################################################  
Name Value Pairs

----------------------------

**var *address*** = **'100 Main St Cairo '**;  
***console***.log(***address***);

ده معناه ان عباره عن فاريبول بياخد كذا قيمه بداخله و لكن فى الاخر هو قيمه واحده محطوطه جوه سترينج

#######################################################################  
undefined قيمه ال

-----------

دى قيمه خاصه موجوده فى الجافا سكريبت معناها ان الفاريبول موجوده بس مفيهاش قيمه بس كده

كود يوضح الكلام ده بقه و الاوت بوت بتاعه

**var *Myname***;  
***console***.log(***Myname***); *// output : --> undefined <---***if** (***Myname*** === **undefined**)  
{  
 ***console***.log(**'Myname is undefined'**);  
}  
**else**{  
 ***console***.log(**'Myname is defined'**);  
}

ملحوظه اوعى عشان تقارن ولا عشان تعمل اى حاجه تروح حاطط جوه اى فاريبول القيمه undefined عشان بتعمل مشاكل على المدى البعيد فانت خد حظرك و متحطهاش فى فاريبول على الاطلاق

#######################################################################  
JS is **Single Threaded Synchronous Execution**

--------------------------------------------------------

ده معناها ان الجافا سكريبت : بتعمل الشغل الى فى ايدها تخلصه و تشوف غيره تنفذه يعنى تنفذ الامر ده دلوقتى خلصته تشوف غيره انما **مش** بتشتغل فى كذا امر مع بعض و تخلصهم مع بعض الا فى حالات معينه

#######################################################################  
Execution Stack

------------------

*/\* start Global scope\*/***function** *a*() {  
 */\* start Func a() scope\*/  
 b*(); *// 2* **var** c ; *// 4  
 /\* end Func a() scope\*/*}  
**function** *b*() {  
 */\* start Func b() scope\*/* **var** d ; *// 3  
 /\* end Func b() scope\*/*}  
*a*(); *// 1***var *d***; *// 5  
/\* end Global scope\*/  
// it is saving Excution like stack  
/\*\*  
 \* 3- b() function  
 \* 2- a() function  
 \* 1- Global Excution  
 \* \*/  
// then you can see it Remove excution after save by this order  
/\*\*  
 \* 1- b() function as it Run and Complete firstly  
 \* 2- a() function as it Run and Complete Secondly  
 \* 3- Global Excution as it Run and Complete in the last   
 \* you can understanc it by Order Numbers i put in code   
 \* \*/*

#######################################################################  
Variable Environment

---------------------------

*/\* start Global scope\*/***function** *b*() {  
 */\* start Func b() scope\*/* **var** Myvar ;  
 ***console***.log(Myvar); *// Order => (3) => output : undefined  
 /\* end Func b() scope\*/*}  
  
**function** *a*() {  
 */\* start Func a() scope\*/* **var** Myvar = 7 ;  
 ***console***.log(Myvar); *// Order => (2) => output : 7  
 b*();  
 */\* end Func a() scope\*/*}  
  
**var *Myvar*** = 5 ;  
***console***.log(***Myvar***); *// Order => (1) => output : 5  
a*();  
***console***.log(***Myvar***); *// Order => (4) => output : 5  
// as it is in Global scope so it take it's variable from Global scope  
  
/\* end Global scope\*/  
/\*\*  
 \* the Output is   
 \* 5 from Global scope  
 \* 7 from a scope  
 \* undefined from b scope  
 \* 5 from Global scope   
 \* \*/*

من الاوت بوت هتفهم ان اللوكال سكوب لا يؤثر ابدا على الجلوبال سكوب و ان حتى لو الفاريبولز متشابهه فكل واحد جوه الاسكوب بتاعه فريد و كانه واحد تانى خالص و بيستجل فى الميمور باسم غير التانى عشان كده لما بحتاج اى واحد فيهم لازم بناديه فى الاسكوب بتاعه و لو نديت فى سكوب مش بتاعه فبيجيب الفاريبول الى الخاص بالاسكوب الى انا فيه يا اسطى مش عشان مثلا سكوب فانكشن a كان قبل اظهار الفاريبول فيروح متغير لأ هتلاقى ان اخر حاجه جابلك الفاريبول الى فى الجلوبال برضوا و ملوش دعوه بالى جره للفريبول فى الفانكشن ايه لانه معتبره واحد تانى خالص   
  
بص اعتبر ان الاسكوب ده اسم اب فممكن يبقى عيال كتير بنفس الاسم بس ابوهم اسمه متغير و بالتالى هقول انا اعرف

* محمود جلوبال عشان انا فى سكوب الجلوبال
* محمود a عشان هو فى سكوب فانكش ال a اللوكال ههههه
* محمود b عشان هو فى سكوب فانكشن ال b اللوكال برضوا

معاك 3 عيال باسم واحد بس ابهتهم مختلفه هل دول يبقوا واحد ؟ لأ طبعا و بالتالى كل اما تروح تاندى لواحده و هو فى الاسكوب بتاع ابوه هيخرجلك محمود الى فى الاسكوب ده بس مش اى حد تانى و بالتالى التلاته مختلفين عن بعض تماما جوه الميمورى انما ظاهريا هما ليهم اسم واحد الى هو محمود حد فاهم حاجه :D

بس ملحوظه مهمه لو انت بقه ناديت على محمود ده فى فانكش a و اصلا فانكشن a ده مجبش عيال اسمهم محمود ساعتها ايه الى بيحصل ؟؟ محمود جلوبال بيرد عشان ده بيعوض عن اى محمود مش موجود على طول

و ده الى هتفهمه من الكود الى جاى ده

*/\* start Global scope\*/***function** *b*() {  
 */\* start Func b() scope\*/* ***console***.log(***Myvar***); *// Order => (3) => output : 5 from Global scope  
 // as the is no Define for Myvar here  
 /\* end Func b() scope\*/*}  
  
**function** *a*() {  
 */\* start Func a() scope\*/* **var** Myvar = 7 ;  
 ***console***.log(Myvar); *// Order => (2) => output : 7  
 b*();  
 */\* end Func a() scope\*/*}  
  
**var *Myvar*** = 5 ;  
***console***.log(***Myvar***); *// Order => (1) => output : 5  
a*();  
***console***.log(***Myvar***); *// Order => (4) => output : 5  
// as it is in Global scope so it take it's variable from Global scope  
  
/\* end Global scope\*/  
/\*\*  
 \* the Output is  
 \* 5 from Global scope  
 \* 7 from a scope  
 \* 5 from Global scope as the is no Define for Myvar in (b) scope  
 \* 5 from Global scope  
 \* \*/*

#######################################################################  
Scope Chain

---------------------------

بص الى انا فهمته ان ياعم جوه الفانكشن ده سكوب لوكال --- بره الفانكشن ده يبقى بالنسبه للفانكشن دى سكوب جلوبال عليها ---- لو فى فاريبول سكوب لوكال و بنفس الاسم بتاع الفاريبول الاسكوب جلوبال الفانكشن هتفضل اللوكال طبعا حتى لو كان انت مستدعى فاريبول قبل ما تكريتها فهتديك undefined و مش هتاخد من الى بره الجلوبال اما لو ماشيه بالترتيب الصحيح فهتاخد اللوكال طبعا اما لو الفاريبول الجلوبال دى دخلت جوه الفانكشن لقت استدعاء بس و مفيش انشاء لفاريبول بنفس الاسم ساعتها الجلوبال هى الى هتشتغل

مثال على جلوبال موقوف حالها و مش شغاله عشان اللوكال شغال غصب عن عين التخين و فى نفس الوقت جلوبال صغير جوه اللوكال بيوريك انك تقدر تعمل جلوبال على مستوى الفانكشن لو جطيت فانكشن تانيه جواها و لاحظ ان الفانكشن الى داخل فانكشن دى لما بتيجى تناديها بتناديها فى اللوكال سكوب بتاع الفانكشن الاب مش الجلوبال سكوب العام

*/\* start Global scope\*/***var *Myvar*** = 5 ; *// it will work if there is not any variable in local Scopes with same name***function** *a*() {  
 */\* start Func a() scope\*/* **function** *b*() {  
 */\* start Func b() scope\*/* ***console***.log(Myvar); *// 3- 7 from the local Myvar in a() which is Global for b()  
 /\* end Func b() scope\*/* }  
 ***console***.log(Myvar); *// 1- undefined as it has local Var but it is before the variable* **var** Myvar = 7 ; *// it is local for func a() Global for Func (b)* ***console***.log(Myvar); *// 2- 7 s it has local Var but it is after the variable  
 b*(); *// can call b() here as it is it's Global Scope in the func a()  
 /\* end Func a() scope\*/*}  
*a*();  
*// we can not Call b(); here it will make Error  
//so output will be 7  
/\* output is :  
\* undefined  
\* 7  
\* 7  
\* \*/*

خد بالك بقه انك تقدر تحط فانكشن جوه فانكشن جوه فانكشن جوه فانكشن بس استدعاء كل واحده بيبقى فى ابوها و مينفعش فى جدها بقه هههههههههه

#######################################################################  
Asynchronous Callbacks

و مناداه الايفينت اذاى فى الجافا سكريبت

---------------------------

بص يا سيدى الموضوع و ما فيه ان عندنا نوعين من الفانكشن Asynchronous Synchronous Vs و الفرق بينهم هنوضحه دلوقتى

أولا ال synchronous دى الفانكشنز العاديه الى هو الفانكشن بتتعمل الكومبيلر يخش يحطها فى الاستاك و لما اعملها كول و اناديها تروح تتنادى و يفضل الكومبيلر شغال جواها لحد ما يخلص و لما يخلص خالص يخرج منها فمثلا لو انا عامل فانكشن رفع فيديو مثلا و الفيديو ده كبير جدا فكده الفانكشن هتاخد وقت كبير جدا على ما يخرج منها و يشتغل على حاجه تانيه و بالتالى كده شكرا الفانكشن دى مش نافعه تتعمل بالطريقه دى خالص لان كده انا وقفت الموقع تماما و شليته

طب ايه حل المشكله دى لما تقبالنا ؟؟

الحل اننا نستخدم Asyncronous ليه بقه ؟؟ لان الطريقه دى بتشتغل ذى ال Threads كده الى هو الكومبيلر هيخش فى الفانكشن دى و مقسم نفسه كذا thread فهيلاقى الفانكشن دى هتطول فهيخلى thread يشتغل عليها و يروح يكمل شغل فى السطور الى بعد الفانكشن دى عادى خالص ب thread تانى و لما الشغل يخلص فى الفانكشن تماما يروح ساعتها مرجع النتيجه بتعتها عادى خالص

مثال على كده من الحياه دلوقتى انا محمود اهه عاوز اجيب بيتزا فرحت اتصلت على محل البيتزا الى هو عملت call يعنى و قولت للراجل اعملى بيتزايه عشان عاوز اكل فراح الراجل ابتدى شغل فى البيتزا و ده يعنى الفانكشن ابتدت تشتغل هل انا بقه هفضل ذى الاهبل مستنيه كده لحد ما يخلص ؟؟ لأ طبعا هروح العب شويه على الكمبيوتر و اضيع الوقت فى اى شغل تانى الى هو يعنى كانى رحت اعمل فانكشن تانيه ملهاش علاقه بالفانكشن الاولانيه الى هو عمل البيتزا و بخلص كل الكلام ده عادى و الراجل شغال بيعمل البيتزا و اول ما بتاع البيتزا يخلص عملها فهيروح الراجل مكلمنى فى التليفون يقولى البيتزايه خلصت و ده معناه ان الراجل هيرد عليا بمكلمه ان البيتزا خلصت يعنى هيدينى النتيجه فى callback ابعتهالى ولا تيجى تستلمها يعنى من الاخر كده هنستلم البيتزا خلاص هناكل ؟؟ لأ طبعا الموضوع لسه مخلصش لان النتيجه الى رجعلى دى هتتحط فى طابور اسمه callback Queue و ده ليه ؟؟ ده يا سيدى ان مش معنى ان الاكل جه فروح اكل ذى الجحش لأ هخلص الى فى ايدى الاول الى هو الفانكشن الى كنت شغال عليها الى هو لو بعلب جيم مثلا هكمله و بعد كده انزل اخد النتيجه عشان جه دورها بعد ماخلصت لعب و كده خلاص هعد اكل و يبقى كده الاسينكرينس فانكشن خلصت ههههههههههههههههههههههه و ارجع اكمل حياتى الطبيعه الى هى باقى الكود يعنى ههههههههههه تخيل بقه انك طالب بيتزا و فراخ و كشرى بقه هههههههه

سؤال محيرنى بقه هو بالشرح ده انا كده فهمت ان الفانكشن الى انا مخليها فيها Async مش بترجع النتيجه بشكل فورى و بتفضل شغاله فيها لحد ما تخلص و بتشغل معاها الكود الى بعده صح , طب ما كده لو الكود الى بعدها معتمد على نتيجتها هيشتغل على نتيجه لسه مرجعتش صح كده ؟؟ بص يا سيدى احنا بنخلى الى بشتغل بس هو الى ملوش علاقه تماما بنتيجه الفانكشن اصلا عشان ميأثرش على نتيجه الفانكشن و بنخلى اى حاجه هتشتغل فى ال Async تبقى حاجه ملهاش علاقه بيها من قريب او من بعيد للعلم بالشى فى فانكشنز كتير جاهزه فى ال JS شغاله Async ذى setTimeout , SetInterval , Events  
و فى طبعا فانكشنز انت بقه بتبنيها بأيدك

بص يا سيدى ده كود بيوضح ال Syncronous Func هنا انا عامل فانكشن بتستغرق 3 ثوانى عشان تخلص فهنا على الرغم من انها بعد فانكشن الكليك هاندلار الى هو المفروض انى لما ادوس على اى مكان فى الصفحه تظهرلى الرساله بتاعت كليك ايفنت الا ان انا لو دوست فى اى وقت قبل ما ال 3 ثوانى يخلصوا مفيش رساله هتظهر خالص و ده معناه ان فانكشن ال waitThreeSecond عطلت كل الصفحه لحد ما تخلص و ده مثال حى على المشكله الى ال delay و العطله الى الفانكشن دى تسببت فيها

*// long running function***function** *waitThreeSeconds*() {  
 **var** ms = 3000 + **new *Date***().getTime();  
 **while** (**new *Date***() < ms){}  
 ***console***.log(**'finished function'**);  
}  
  
**function** *clickHandler*() {  
 ***console***.log(**'click event!'**); *// 3- output : click event!  
 // but it will Shown in browser after Click on Doc*}  
  
*// listen for the click event****document***.addEventListener(**'click'**, *clickHandler*);  
  
*waitThreeSeconds*(); *// 1- output : finished function****console***.log(**'finished execution'**); *// 2- output : finished function*

و هيكون حل الشمكله دى ايه بقه اننا نستخدم ال Asyncronous

تعالى بقه نشوف الكود بال Asyncrouns

بص ياعم الحج الكود عشان يبقى Async لازم يبقى فيه Wait و انا بصراحه هأجل موضوع شرح الكود لحد ما تفهم البروميس عشان تربط و تفهم الدنيا أحسن

#######################################################################  
JS Dynamic Typing

---------------------------

ده معناها ان var بتعرف اى فاريبول على انها Object و فى اى مكان فى الكود تقدر تغيير نوعها عادى خالص على عكس لو بتستخدم let و محددلها نوع ساعتها متقدرش تغير النوع الا بطرق معينه معروفه

و ده مثال يفهمك الوضع

**var *Myvar*** = **true**;  
***console***.log(***Myvar***); *//output true****Myvar*** = **'Mahmoud'**;  
***console***.log(***Myvar***); *//output Mahmoud****Myvar*** = 20;  
***console***.log(***Myvar***); *////output 20  
  
/\* as the var Type is Dynamic So No Error and output will be :   
\* true  
\* Mahmoud  
\* 20  
\* \*/*

اما لو ستاتيك مكنش هيرضى ياخد اى حاجه الا من نوعه بس او من نوع تانى معين بعد البارسينج ولازم يبقى ينفع يتعمله بارسينج كمان يعنى مينفعش اجيب محمود و قوله حوله لرقم مثلا هيشتمنى اكيد يعنى :D

**let *Myvar***:**boolean** = **true**;  
***console***.log(***Myvar***); ***Myvar*** = **'Mahmoud'**;  
***console***.log(***Myvar***); ***Myvar*** = 20;  
***console***.log(***Myvar***);

الكود ده هيدى ايرور عشان انا استخدمت let و قولت ان نوعه Boolean و بالتالى بقى   
static Type و بالتالى مينفعش اجى احط فيه اى بتنجان كده و خلاص

#######################################################################  
Primative Types

---------------------------

انواع الفاريبول من الاخر

* Undefined
* Null
* True OR False
* Num
* String
* Symbol و دى تبع ال ECS 6

#######################################################################  
Opertators

---------------------------

الجمع و الطرح و القسمه و الضرب و المودلاس   
ولازم تبقى عارف انك تقدر تستخدم < و > بس ده ساعتها بيعتبر كانك عامل اف كونديشن و بيطلع Boolean

ده جدول توضيحى شامل

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Operator_Precedence>

**var *Myvar*** = 5 > 4;  
***console***.log(***Myvar***); *//output true*

-------------------------------

طيب الكود ده نتيجته ايه ؟

**var *Myvar*** = 5 + 4 \* 2;  
***console***.log(***Myvar***); *//output 13*

و ده لانه بيعمل الضرب و القسمه قبل الجمع و الطرح

طب لو استخدمنا القوس فى الجمع النتيجه تتغير ايوه طبعا لانه بيشوف الى جوه القوس ينفذه و بعد كده ينفذ عليه الشغل الى ناقص

**var *Myvar*** = (5 + 4) \* 2;  
***console***.log(***Myvar***); *//output 18*

طيب شوف المثال الى جاى ده هيوريك الفاريبول بيخزن فى بعضه اذاى

**var *a*** = 2, ***b***=3 , ***c***=4;  
***a***=***b***=***c***; *// from Rigth to left C=4 so b = 4 So a = 4****console***.log(***a***); *//output 4****console***.log(***b***); *//output 4****console***.log(***c***); *//output 4*

#######################################################################  
Concatenation between String and Num

--------------------------------------------------

***console***.log(1+**'2'**); *//output 12****console***.log(**'1'**+2); *//output 12****console***.log(**'1'**+**'2'**); *//output 12****console***.log(*parseInt*(**'1'**)+*parseInt*(**'2'**)); *//output 3*

لاحظ ان اى حاجه استرينج + رقم بتبقى كانك حطيت كلام جنبه رقم عادى خالص   
بينما لو حولت الرقم الاسترينج ده لرقم فعلا بالبارسينج هتلاقيه اعتبره انتيجر و جمعه عادى خالص و اداك نتيجه جمع صحيحه

#######################################################################  
Comparison Operators

-----------------------------

فى اكبر و اصغر و الكلام العبيط ده و عمتا كله بيستخدم مع الايف كونديشن بس هنيجى بقه للى محتاجين نفهمه صح و هكتب الكود يوضح ايه الى بيحصل

***console***.log(-4 < 2); *//output true****console***.log(-4 > 2); *//output false****console***.log(**true** < 2); *//output true****console***.log(**true** > 2); *//output false****console***.log(**true** == 1); *//output true****console***.log(**true** === 1); *//output false****console***.log(**false** == 0); *//output true****console***.log(**false** === 0); *//output false****console***.log(***Number***(**false**) === 0); *//output true*

***console***.log(***Number***(**false**) === ***Number***(0)); *//output true****console***.log(*parseInt*(**false**) === 0); *//output false parseint does not make false as num****console***.log(*parseInt*(**false**)); *//output NAN****console***.log(*parseInt*(0)); *//output 0****console***.log(*parseInt*(**false**) === *parseInt*(0)); *//output false as false will be NAN after parse to int  
// and 0 paresed to int to 0 so NAN is not equal to NAN*

ذى ما انت شايف الترو بوليان فلو قارنتها بواحد ب == بيدى واحد عادى و بيدى ترو انما اول ما هجى اقوله ان المقارنه ب === هيقولى فولس لان ساعتها دى رقم و دى بوليان لانه بيعتبر الفولس ب 0 و الترو ب 1 بس النوع مختلف و انت عارف === لازم النوع يبقى نفس النوع

طيب الزيرو هنا مصنف انه رقم اذا اعمل للفولس برضوا بارس للرقم فهيدينى فعلا ترو بينما لو عملتله بارس انتيجر هيدى فولس لان الصفر اصلا انا مش معرفه على انه انتيجر لا ا الجافا سكريبت معتبراه رقم و خلاص

و عمتا انت لو حاولت تحول false لانتيجر هيديك NAN مش هيديك رقم اصلا انما لما بتحوله لرقم بيتحول ل 0

ده جدول جميل تعرف منه كل حاجه فى الكلام ده

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Equality_comparisons_and_sameness>

#######################################################################  
dangerous Return Semi Colon

--------------------------

خد بالك ان الريتيرن مينفعش تتكتب فى سطر و الى هترجعه فى سطر لان اللغه اوتوماتيك بتحط بعد السطر ; فساعتها الدنيا هتخرب و عشان تفهم جرب الكودين دول

**function** *MyFunc*() {  
**return** {

**Name** : **'Mahmoid'** , **Age** : 15}  
}  
***console***.log(*MyFunc*());

شوف بقه لما اعمل الريتيرن و الداتا الى المفروض يرجعها فى سطر هيديك انه error على طول

**function** *MyFunc*() {  
**return** {  
 Name : **'Mahmoid'** , Age : 15   
 } ;  
}  
***console***.log(*MyFunc*());

لانه اعتبر ان الكود مكتوب كده و اسوأ كمان

**function** *MyFunc*() {  
**return** ;  
}  
***console***.log(*MyFunc*());