C# Interview Questions Summery By Mahmoud Badawy

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

MB

##################################################################

اول حاجه كده ملخص هحط فيه كل الاساله و الاجوبه الخاصه بال C# و Dot net للانترفيوهات

و نبتدى معاكم اول سؤال

MB

##################################################################

In Debugging What is Different Between (Autos , Locals , Watches)

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

A screenshot of a computer

Description automatically generated  
Locals : دى و انت بتعمل ديباج بتجيبلك كل قيم الفاريبولزو البراميتارز حتى الى فى فانكشن بتناديها و الى عديت عليها و الى انت فى نفس الاسكوب بتعها و بتعرفك انت فى انهى اسكوب دلوقتى و اى تغيير فى اى فاريبول من الفاريبولز دى بتعرفها لان لون الفاريبول و قيمته بيتغيروا للون الاحمر و ده معناه ان حصل تغيير بعد الخطوه الى انت مشيتها دى فبتعرف التغييره حصلت امتى من متابعه ال locals دى لانها بتتعامل مع الاسكوب و بالتالى اى مشكله خاصه بقيمه ما تم تغييرها و انت مش فاهم اتغيرت فين و امتى بتعرفه عن طريق ال locals

Autos : نفس الى اللوكال بتعمله بس الفرق انها بتراقب الفاريبولز فقط و ملهاش دعوه باى براميتر مبتشوفهمش هى بتركز على الفاريبولز فقط و التغيرات الى بتطرأ عليها كل شويه سواء فى نفس الاسكوب او فى اسكوبس مختلفه

Watches : دى بقه يا سيدى بتشتغل بمزاجك انت يعنى تحطلها فاريبول معينه و تتبعها هى على وجه الخصوص علشان متوجعش دماغك بالباقى و عادتا احنا بنستخدم دى بدلا من اللوكالز علشان نعرف نركز على الفاريبول الى بيحصل فيها المشكله بسرعه

MB

##################################################################

Ternary Operator Vs If-Else

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

الفرق الواضح جدا ان if – else مش لازم يعمل ريتيرن لقيمه - ينفع نحط جواها Retern للقيم او ممكن نعمل كود مبيعملش Return لاى قيمه و يبقى مجرد قراءه للداتا او تغيير فى قيمه فاريبول بس مش هعمله ريتيرن عادى

اما ال Ternary Operator : ده لازم يعمل return لقيمه فلو معملتش ريترين لقيمه منه هيديك ايرور

A screen shot of a computer

Description automatically generated

MB

##################################################################

Do While Vs While

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

While : مش هيشتغل لو الى جواه مبيحققش الشرط

Do While : اول مره هيشغل الكود الى جوه ال do حتى لو كان مش محقق الشرط الى جوه while بعد كده يبتدى يطبق الشروط لما يخش بقه فى ال while

var x = 11;

while (x < 11 && x >=0)

{

Console.WriteLine("Hello from While");

--x;

}

var x2 = 11;

do

{

Console.WriteLine("Hello from Do While ");

--x2;

}

while (x2 < 11 && x2 >= 0);

MB

########################################################################

Value Type Vs Refrence Type  
---------------------------------

عندنا بقه انواع من ال Data type من نوع ال Value Type و نوع تانى من نوع Refrence Type

طيب ايه الفرق بينهم ؟؟

ال Value Type ذى مثلا ال ال

Bool , byte , char, decimal, double , enum , float , int , long , sbyte , short , struct , uint , ulong , ushort

و دى يا سيدى الميكانيزم بتعها ان اسمها و قيمتها بيتسجلوا فى ال Stack

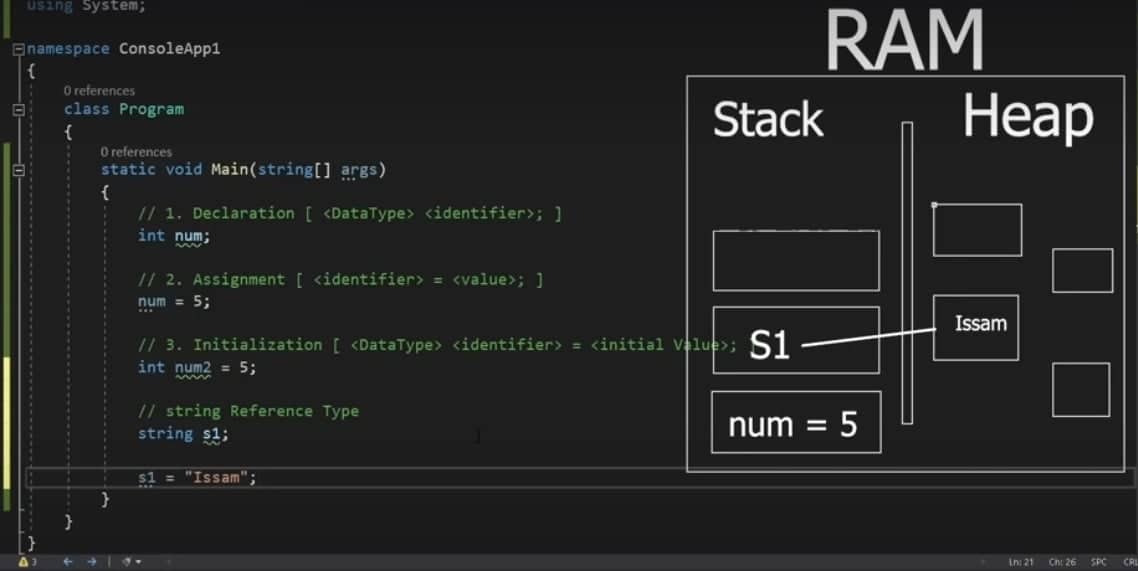
يعنى لما باجى اعرفها ذى ما فى الابلكيشن ده كده

Int16 Num = 12345; // Decleration and assignation Value is in Stack

بينما ال Refrence Type بيكون التعريف او الدكليريشن بتعها فى ال Stack و بيشاور على القيمه الى موجوده فى ال Heap و من اهم الانواع الى من نوع Refresnce Type هى string , Array , List , Objects

string Num = "Mebo"; // Decleration in stack and assignation Value is in Heap

و الصوره دى هتوضح اكتر للناس علشان تفعم طريقه التخزين



ذى ما احنا شايفين الانتيجر اتسجل الدكليريشن بتاعه الى هو ال num فى Stack و كذلك القيمه الخاصه بيه الى هى 5 فى ال Stack  
اما الاسترينج بقه راح اتسجل الدكليريشن بتاعه الى هو S1 فى Stack و راح رايح مخزن فيمته فى مكان تانى اوسع فى المساحه و اسرع اسمه ال Heap

الفرق التانى بقه يا معلم و المهم   
اولا ال Value Type انت لما بتديله قيمه بياخدها بشكل مباشر القيمه و ليس عنوان القيمه وبالتالى لو عمل للقيمه دى اسين على فاريبول جديد فهو بياخد القيمه من الفاريبول الاول برضوا بشكل مباشر و ليس عنوان القيمه و بالتالى لو فرضنا ان الفاريبول الاصلى غير القيمه بعد كده فالفاريبول الى اخد القيمه القديمه منه مش هيتاثر خالص و العكس صحيح

بالتالى طريقه التخزين اشبه ب ان   
واحد راح يتصور فى الاستوديو 2 صور و لما اخد ال 2 صور الورق دول راح مدى صوره لصحبه يعينهاله معاه و صوره لنفسه فبقى كل واحد منهم معاه الصوره دى الى هى ستايل كاجول هنقول قيمتها 5 مثلا فكده القيمه 5 دى مع الراجل و مع صاحبه برضوا

جه بعد ساعه لما صاحبه مشى و اخد الصوره بتعته معاه قعد يفكر و راح الراجل ده قالك الصوره الى معايا دى مش عجبانى و قطعها و راح اتصور صوره تانيه بلبس تانى خالص قيمته و ليكن 10 طيب اصبح دلوقتى الراجل معاه صوره ستايل جديد ب 10 و صاجبه معاه صوره ستايل قديم ب 5 فكده تغيير الصوره الى مع الراجل الاصلى و قيمتها ماثر على الصوره او القيمه الى مع صاحبه و ده الى بيتسمى ال Value type و سميت بالاسم ده نظرا لانها تعتمد على القيمه الفعليه لل Variable   
مثال

void ValueType()

{

int num1 = 5;

int num2 = num1;

num1 = 10;

Console.WriteLine(num1); // 10

Console.WriteLine(num2); // 5

}

ثانيا ال Reference Type ده يا سيدى انت لما بتديله قيمه بياخدها بشكل عنوان و ليس القيمه وبالتالى الفاريبول الاصلى بيعمل للعنوان ده اسين على فاريبول جديد فهو كمان بياخد العنوان ده من الفاريبول الاول و لما بيحب اى حد منهم يعرف القيمه بيروح يجيبها عن طريق العنوان الى اتسجل فى ال Heap ده و بالتالى اى تغيير فى القيمه سواء عن طريق الفاريبول الاصلى او النسخه الى منه الاتنين بيشوفوا القيمه الجديده و ده لان الاتنين مخزنين عناوين و ليس قيم و العنوان متمش تغييره لا ده القيمه المقابله للعنوان هى الى اتغيرت و بما ان الاتنين بصيين على نفس العنوان فالاتنين هيشوفوا نفس القيمه الجديده و لذلك طريقه التخزين اشبه ب ان

واحد راح عمل جروب واتس و عمل جروب مع صاحبه و الاتنين ادمنز جه حط صوره للجروب الى هنعتبرها بقيمه 5 فهو و صاحبه دلوقتى شايفين الصوره الى قيمتها 5 على الجروب ده الى هو هنعتبره العنوان الى هخزن فيه الصوره بتاعت الجروب و الى هى هنعتبرها

جه بعد يومين صاحبه مش عاجبه الصوره الى الاولانى حطها فراح مغيرها لصوره جديده قيمتها 10 فدلوقتى الاتنين صحاب شايفين الصوره الى قيمتها 10 و ادى مثال يا جماعه

void RefrenceType()

{

int[] num1 = { 5 };

int[] num2 = num1;

num1[0] = 10;

Console.WriteLine(num1[0]); // 10

Console.WriteLine(num2[0]); // 10

}