  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  


**Part 01**

Question: What is the purpose of the finally block?

الـ finally block بيتنفذ دايمًا بعد الـ try/ catch، سواء حصل خطأ أو لأ، وبيستخدم عادةً لإغلاق الموارد أو تنظيف الكود زي إغلاق ملف أو اتصال بقاعدة البيانات.

Question: How does int.TryParse() improve program robustness compared to int.Parse()?

‎ int.TryParse() بيحاول يحول النص لرقم، ولو فشل بيرجع false بدل ما يرمي خطأ، وده بيخلي البرنامج يكمل شغله بدون ما ينهار، بعكس int.Parse() اللي بيرمي استثناء لو التحويل فشل.

Question: What exception occurs when trying to access Value on a null Nullable<T>?

بيحصل استثناء من النوع InvalidOperationException لما تحاول توصل لـ `.Value` في \*\*Nullable<T>\*\* وهي قيمتها `null`.

Question: Why is it necessary to check array bounds before accessing elements?

علشان تتجنب استثناء IndexOutOfRangeException اللي بيحصل لو حاولت توصل لعنصر بره حدود المصفوفة.

Question: How is the GetLength(dimension) method used in multi-dimensional arrays?

‎GetLength(dimension) بترجع عدد العناصر في بُعد معين من المصفوفة متعددة الأبعاد، وبتحدد البُعد بالرقم 0 للبُعد الأول، 1 للثاني، وهكذا.

Question: How does the memory allocation differ between jagged arrays and rectangular arrays?

في rectangular arrays الذاكرة بتتحجز ككتلة واحدة ثابتة لكل الصفوف والأعمدة، لكن في jagged arrays كل صف بيكون مصفوفة مستقلة في الذاكرة وبتتحجز لوحدها، وده بيدي مرونة في أحجام الصفوف.

Question: What is the purpose of nullable reference types in C#?

الغرض من nullable reference types هو تقليل أخطاء الـ NullReferenceException عن طريق تنبيهك في وقت الكومبايل إذا فيه متغير مرجعي ممكن يكون `null`، فتتعامل مع الحالة دي قبل التشغيل.

Question: What is the performance impact of boxing and unboxing in C#?

الـ boxing و unboxing بيأثروا على الأداء لأنهم بينقلوا البيانات بين الـ \*\*stack\*\* والـ \*\*heap\*\* وبيعملوا نسخ للكائنات، وده بياخد وقت وذاكرة أكتر مقارنة بالأنواع القيمة مباشرة.

Question: Why must out parameters be initialized inside the method?

لأن الـ out parameters بيُستخدموا علشان يرجعوا قيمة من الدالة، فلازم تتحدد لهم قيمة جوه الدالة قبل ما ترجع، وإلا البرنامج هيرجع بيانات غير مهيأة.

Question: Why must optional parameters always appear at the end of a method's parameter list?

علشان عند استدعاء الدالة، أي باراميتر بعديهم هيتلغبط مع القيم الافتراضية، فوجودهم في الآخر بيخلي الكومبايلر يعرف يطابق القيم مع الباراميترات صح.

Question: How does the null propagation operator prevent NullReferenceException?

الـ null propagation operator (`?.`) بيوقف تنفيذ الوصول لو الكائن `null`، وبيرجع `null` بدل ما يرمي NullReferenceException.

Question: When is a switch expression preferred over a traditional if statement?

بيكون الـ switch expression أفضل لما يكون عندك مقارنة لقيمة واحدة مع احتمالات متعددة، وعايز كود أقصر وأوضح من تكرار if/else كتير.

Question: What are the limitations of the params keyword in method definitions?

* لازم يكون آخر باراميتر في الدالة.
* مينفعش يكون فيه أكتر من params واحد.
* بيشتغل مع نوع واحد محدد أو مصفوفة من النوع ده بس.

**Part03 Bonus**

**Casting Operator:**  
 تحويل نوع بيانات لنوع تاني. ممكن يكون تلقائي أو لازم تقولله تعمل التحويل.

**Upcasting:**   
تحويل كائن من نوع فرعي لنوع أب. دايماً آمن.

**Downcasting:**   
تحويل من نوع أب لنوع فرعي. ممكن يغلط لو النوع مش صحيح.

**Ref Type Passing**:  
 لما تبعت متغير بـ ref، التغييرات جوه الدالة بتأثر على المتغير الأصلي.