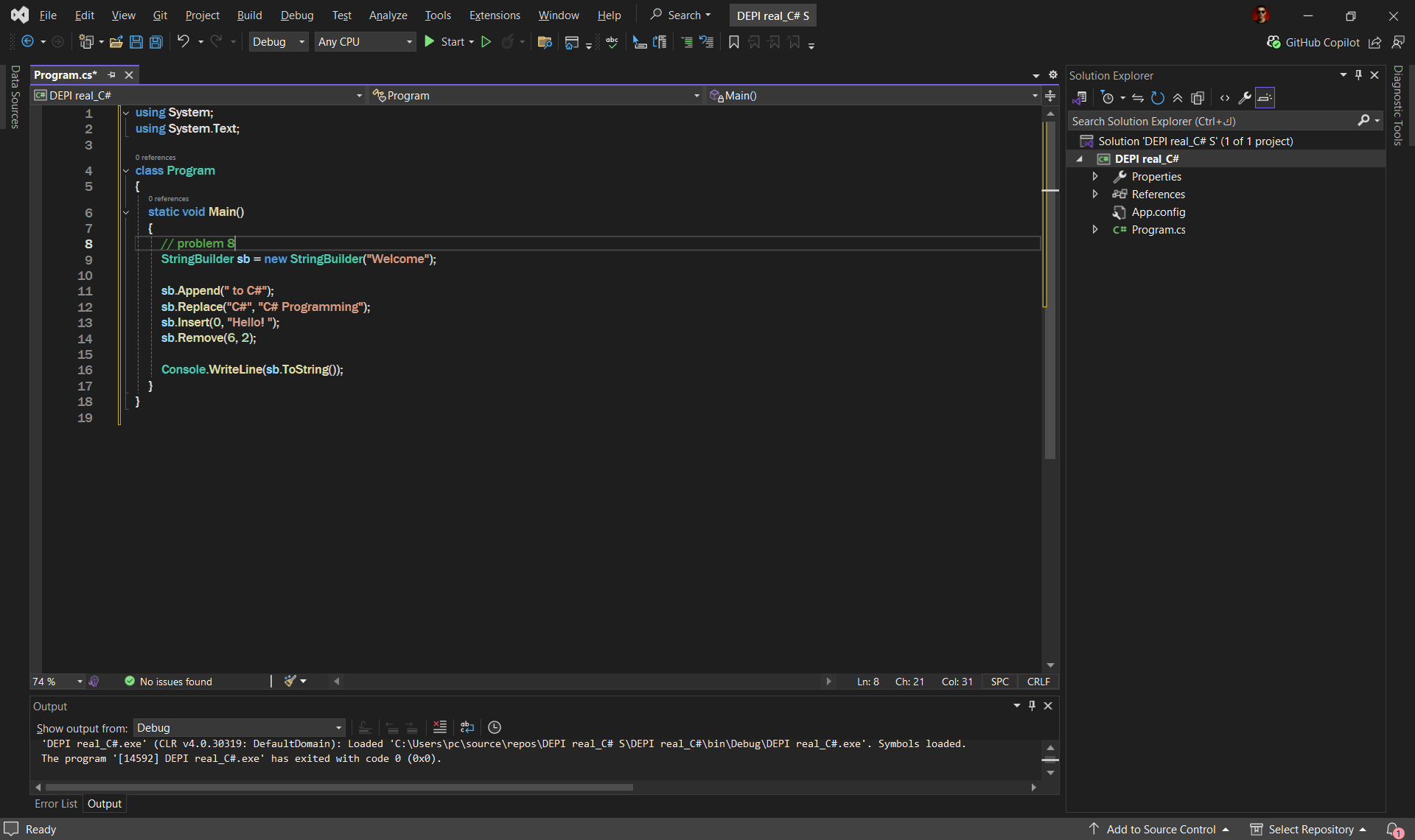
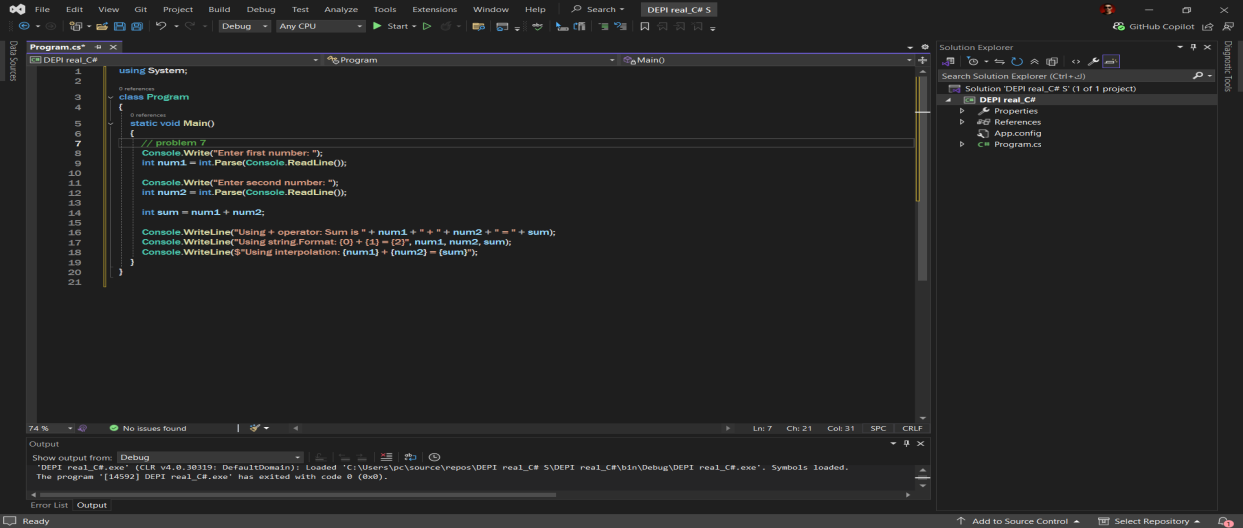
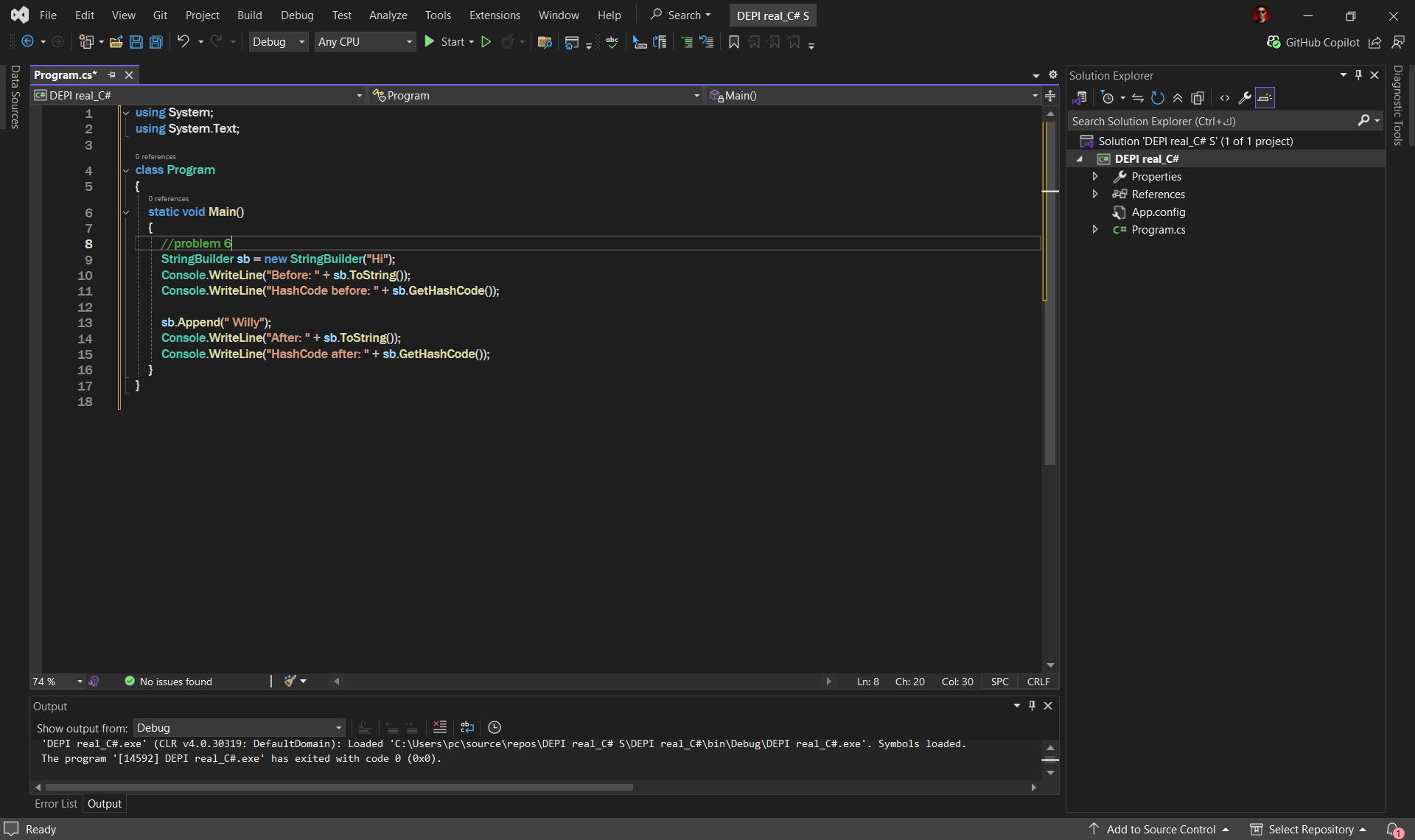
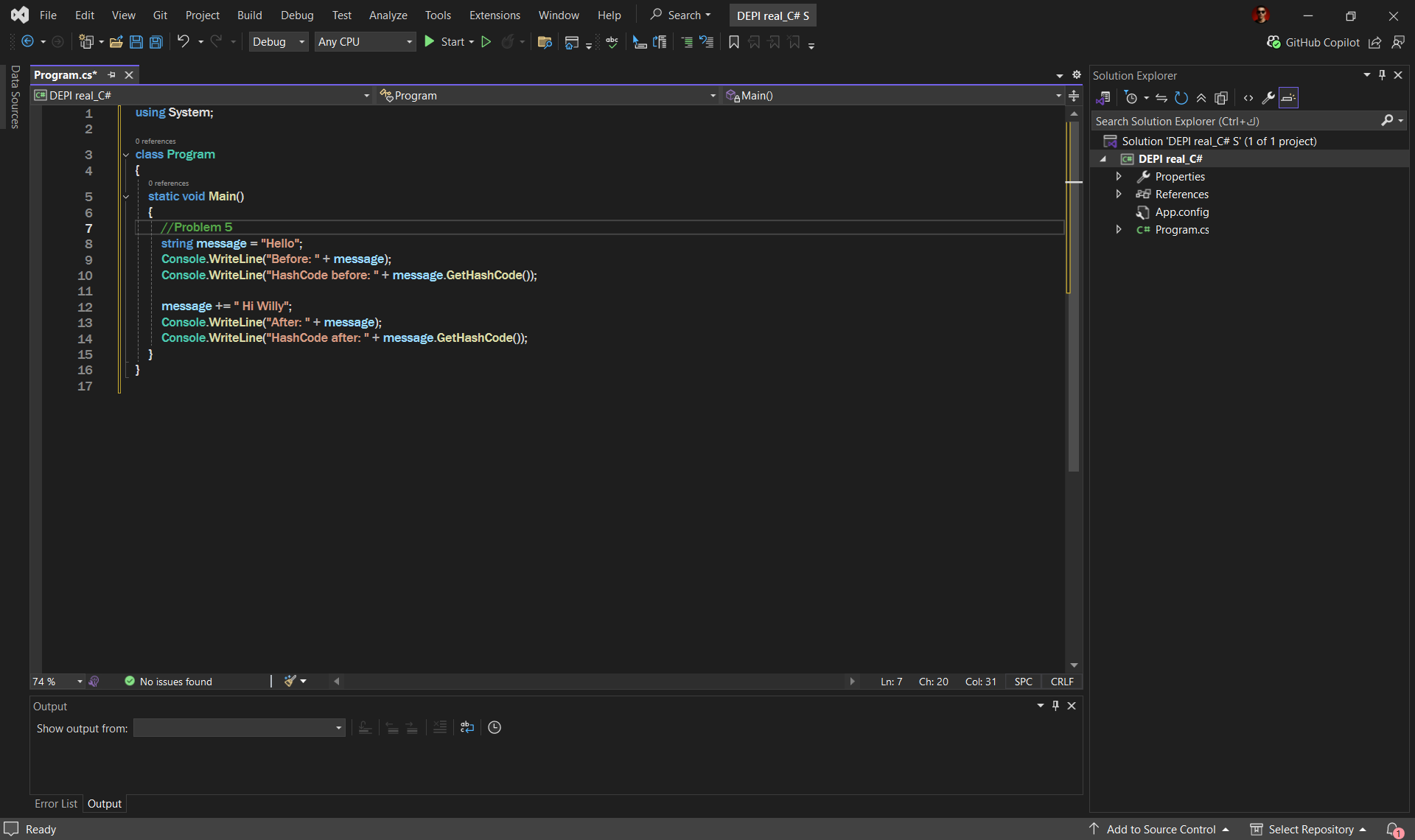
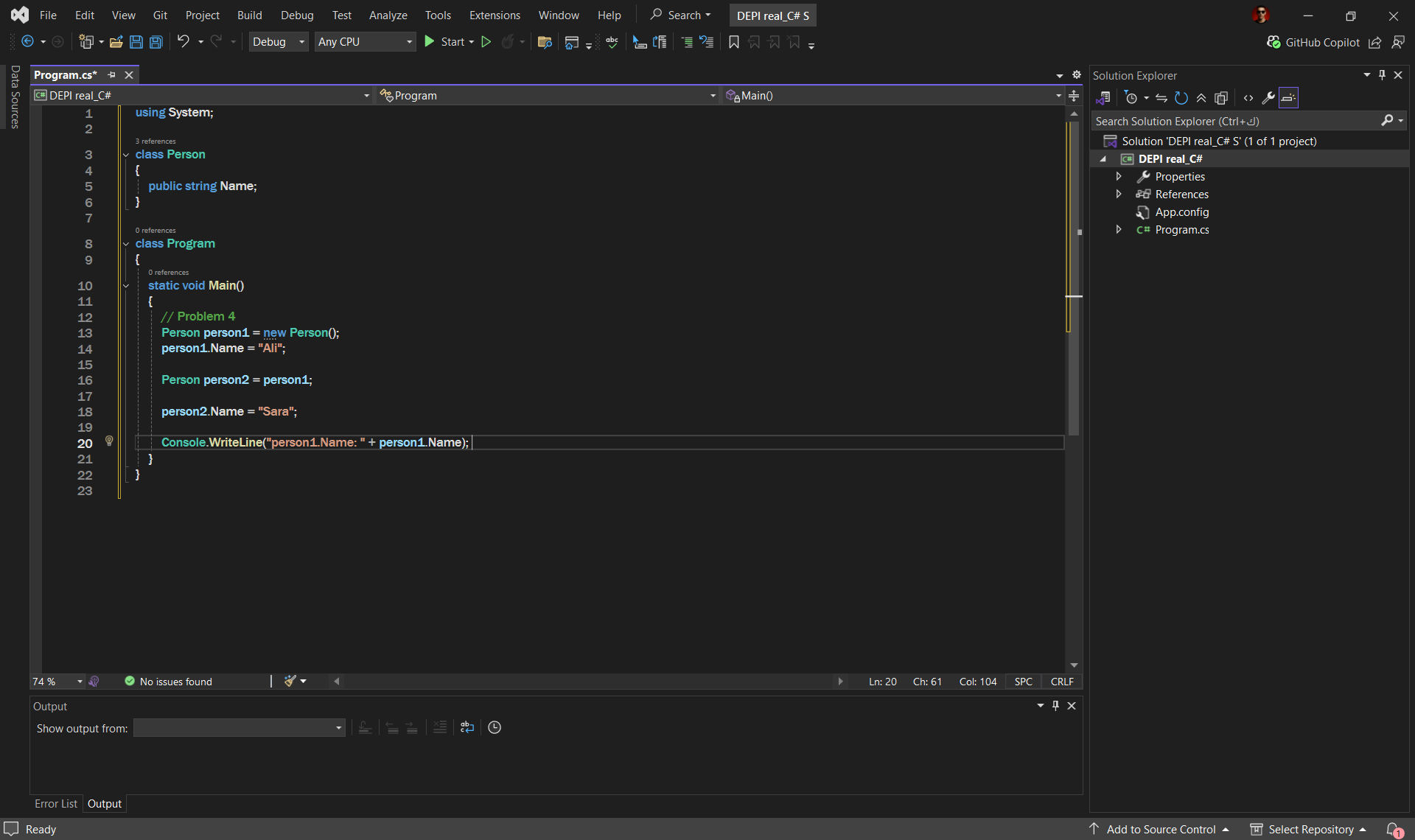
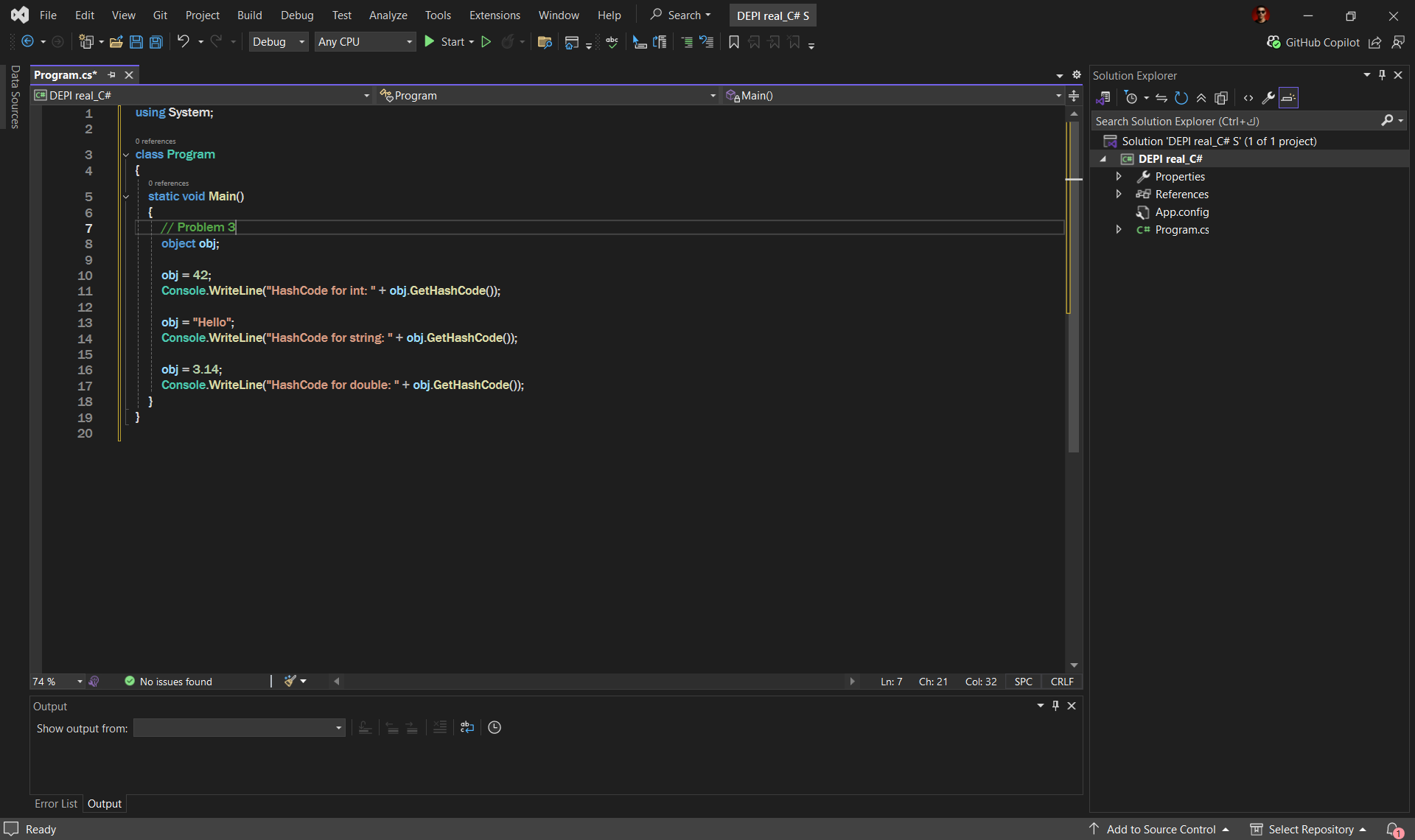
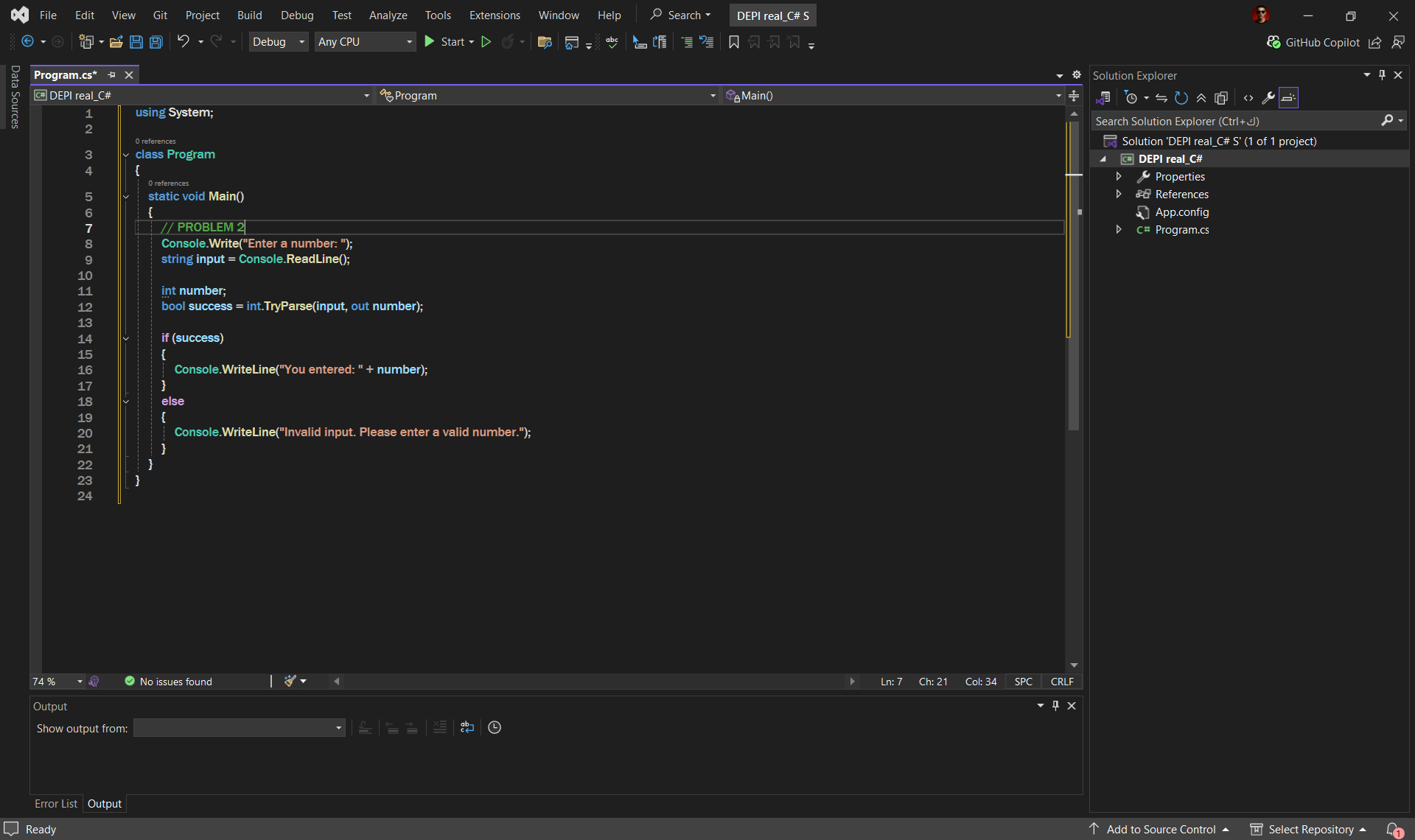
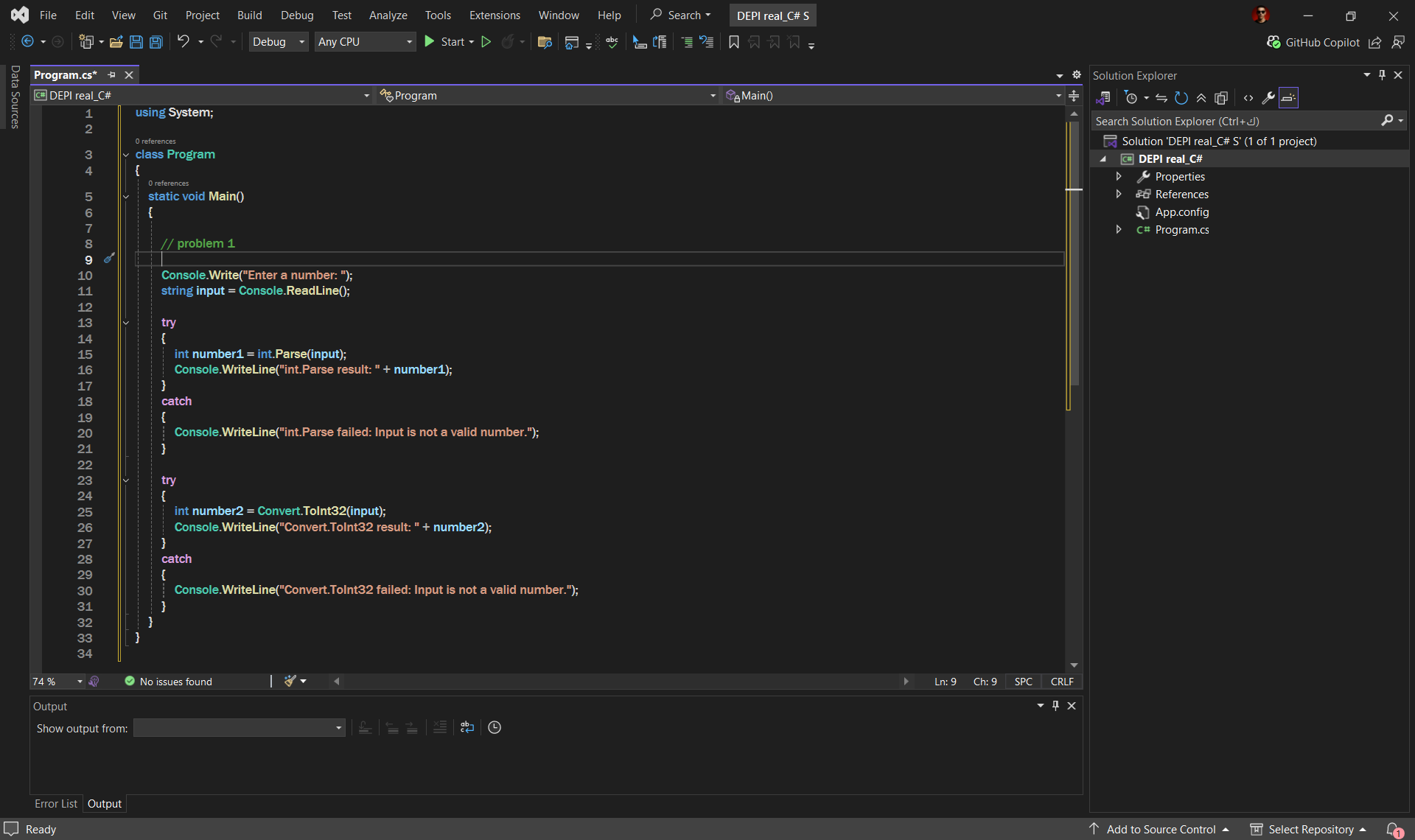
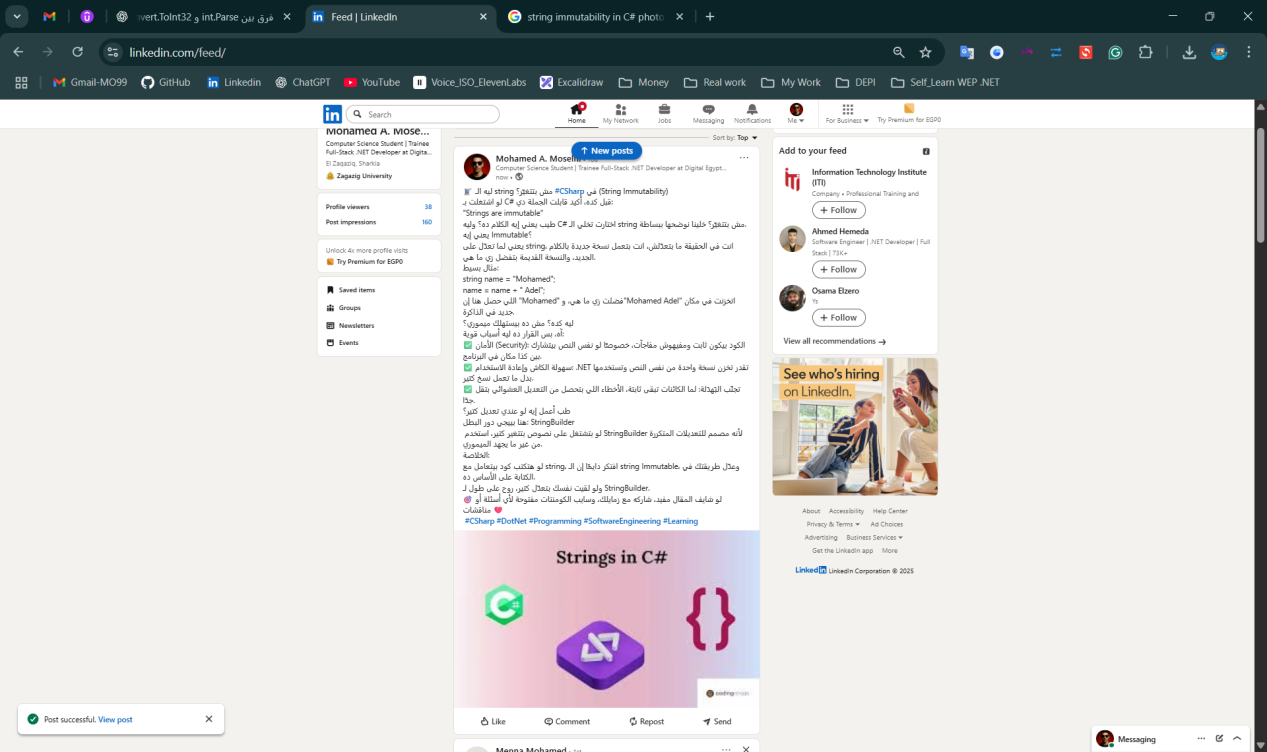
****

**Part01**

**Question: What is the difference between int.Parse and Convert.ToInt32 when**

**handling null inputs?**

**الفرق الرئيسي بين int.Parse و Convert.ToInt32 لما تتعامل مع null هو إن:**

**int.Parse(null) هيعمل Error (Exception) ويقولك فيه مشكلة لأنه مش بيعرف يتعامل مع null خالص.**

**لكن Convert.ToInt32(null) هيكون أذكى شوية، هيرجع 0 من غير ما يطلع Error.**

**يعني لو فيه احتمال إن القيمة تكون null، الأفضل تستخدم Convert.ToInt32 عشان ما يكسرش البرنامج.**

**Question: Why is TryParse recommended over Parse in user-facing applications?**

**TryParse بيتفضل على Parse في البرامج اللي الناس هتستخدمها (user-facing) لأنه أأمن ومش بيوقع البرنامج.**

**Parse لو المستخدم دخل قيمة غلط زي كلمة بدل رقم)، البرنامج هيطلع Error وهيقف).**

**لكن TryParse بيحاول يحوّل، ولو معرفش، ما بيعملش Error، بس بيرجع false وانت تتصرف.**

**يعني تقدر تقول للمستخدم "دخل رقم صحيح" بدل ما البرنامج يهنّج.**

**Question: Explain the real purpose of the GetHashCode() method.**

**GetHashCode() وظيفته يطلع رقم يعبر عن الكائن، عشان يسهل على البرنامج يلاقيه بسرعة في الحاجات اللي شبه القوائم الكبيرة، زي Dictionary أو HashSet.**

**يعني بدل ما يدور على الكائن نفسه، بيدور بالرقم اللي رجعه GetHashCode، وده أسرع بكتير.**

**Question: What is the significance of reference equality in .NET?**

**Reference Equality معناها تسأل:**

**هل الاتنين دول بيشاوروا على نفس الكائن بالضبط في الذاكرة؟**

**يعني مش مهم لو عندهم نفس القيم، المهم هل هما نسخة واحدة ولا لأ.**

**Question: Why string is immutable in C# ?**

**الـ string في C# immutable يعني ماينفعش يتغير بعد ما يتعمل.**

**الـ string في C# مش بتتغير بعد ما تتعمل، يعني لو عدلت عليها، بيعمل نسخة جديدة ومش بيغيّر القديمة.**

**ليه كده؟**

**علشان C# تبقى أأمن وأرتب. لو كذا جزء في البرنامج بيستخدم نفس الكلمة، التعديلات ما تبوظش الدنيا. وكمان ده بيسهّل على السيستم إنه يحتفظ بنسخة واحدة من كل كلمة بدل ما يكررها**

**Question: How does StringBuilder address the inefficiencies of string concatenation?**

**لما تربط string كتير بـ +، كل مرة بيعمل نسخة جديدة وده بيبطّأ البرنامج.**

**لكن StringBuilder بيجمع الكلام كله في نفس المكان من غير ما يكرر النسخ، فبيكون أسرع وأخف.**

**Question: Why is StringBuilder faster for large-scale string modifications?**

**StringBuilder أسرع لما بتعدل كتير في النص لأنه بيشتغل على نفس الحاجة،**

**لكن الـ string العادي كل مرة بيعمل نسخة جديدة، وده بيبطّأ الدنيا.**

**Question: Which string formatting method is most used and why?**

**الطريقة اللي أغلب الناس بتستخدمها في تنسيق الكلام في C# هي اللي بنكتب فيها المتغير جوا الجملة كده على طول، من غير لف ولا دوران. ليه؟ لأنها أسهل، شكلها مفهوم، ومبتتعبكش وانت بتقرأ أو تكتب الكود.**

**يعني بدل ما تقعد تكتب حاجات معقدة، بتكتب الجملة عادي كأنك بتكلم حد، وتحط المتغير في نص الكلام.**

**Question: Explain how StringBuilder is designed to handle frequent modifications**

**compared to strings.**

**الـ string في C# ما بيتغيرش، يعني كل مرة تعدّل عليه (تزود حرف، تشيل، تدمج...) بيعمل نسخة جديدة، وده بيتعب الجهاز وبيبطّأ البرنامج.**

**لكن StringBuilder معمول مخصوص للتعديلات الكتير، بيشتغل على نفس النسخة من الكلام، من غير ما يعيد إنشاء كل حاجة من أول وجديد.**

**يعني تقدر تضيف، تمسح، تعدل براحتك من غير ما السيستم يدوّخ.**

**2- What’s Enum data type, when is it used? And name three common built\_in enums used**

**frequently?**

**1 . يعني إيه Enum؟**

**الـ Enum نوع بيانات بنستخدمه لما يكون عندنا مجموعة قيم ثابتة ومعروفة، وكل قيمة ليها اسم ومعنى.**

**زي مثلًا أيام الأسبوع، أو مستويات الصوت، أو حالة الطلب (تم، جارٍ، ملغي...).**

**2 . إمتى بنستخدمه؟**

**بنستخدم الـ Enum لما نحب نوضّح المعنى بدل ما نكتب أرقام.**

**يعني بدل ما تكتب 0 أو 1، تكتب Low أو High، وده بيخلي الكود أوضح وأسهل في الفهم.**

**3 . أمثلة على Enums جاهزة في C#:**

**فيه Enums كتير C# بتوفرها جاهزة، زي:**

**DayOfWeek بيمثل أيام الأسبوع.**

**ConsoleColor علشان تغير لون الكلام في الكونسول.**

**FileAccess بتستخدمه لتحديد صلاحيات التعامل مع الملفات (قراءة، كتابة...).**

**3- what are scenarios to use string Vs StringBuilder?**

**استخدم string في الحالات دي:**

* **لو النص قصير**
* **مفيش تعديلات كتير عليه**
* **زي جملة بسيطة أو اسم ثابت**

**استخدم StringBuilder في الحالات دي:**

* **لو هتضيف أو تمسح من النص كتير**
* **لو فيه تكرار أو حلقة بتشتغل على النص**
* **لو النص كبير وبيتغيّر باستمرار**

**4- self study report**

**5- what meant by user defined constructor and its role in initialization**

**الـ user-defined constructor هو طريقة إنت بتكتبها بنفسك عشان أول ما الكائن يتعمل، يبدأ وهو جاهز بالقيم اللي إنت محددها.**

**يعني بدل ما تعمل كائن وبعد كده تملاله البيانات واحدة واحدة، لا، تحط القيم من الأول وهو بيتكوّن.**

**زي إيه؟**

**لو عندك طالب، ممكن تخليه يتكوّن بالاسم والسن على طول، بدل ما تعمله وبعد كده تضيف الاسم والسن يدوي.**

1. **compare between Array and Linked List**

**Array:**

* **بتحدد عدد العناصر من الأول**
* **الوصول لأي عنصر سريع**
* **الإضافة والحذف صعبين شوية**

**Linked List:**

* **تقدر تزود أو تشيل عناصر بسهولة**
* **بطيئة في الوصول للعناصر**
* **مش لازم تكون العناصر جنب بعض في الذاكرة**