Major Features of C#

The programming language C# has many important major features, some being similar to those of other languages and some being unique to C#. It has features similar to object-oriented languages such as the ability to create objects by using classes or structs, the allowance of inheritance and polymorphism, and its support for record types. Along with these features, C# draws similarities in features from functional languages such as the use of pattern matching. Despite C#'s similarities to functional and object-oriented languages, it also created features that are exclusive to itself. Two of these features are 'async'/'await' keywords and language-integrated query (LINQ).

C# draws many similarities to object-oriented languages, and it can be seen in some of its features. One place it can be seen is with the ability to create objects. In Microsoft Learn, Objects – create instances of types (2021), Bill Wagner describes how objects in C# are created. Wagner says that object creation in C# can be through instances of classes or instances of structs by using the 'new' operator (optional for structs). Object creation is also a feature that is found in Java, and this is where the origins of objects in C# could be. This could be due to the popularity that Java was experiencing at the time. In order for Microsoft and C# to compete, they needed to add object-oriented features similar to Java, and this included object creation. The origins of objects could have also been inspired by C++ since it utilizes object creation as well. Hejlsberg also said he designed C# with many similarities to C++.

Inheritance and polymorphism are two object-oriented features that are utilized in C#. The origins of these features could also be traced back to Java due to the language's popularity. The support for record types is another feature that C# includes. According to Microsoft Learn, Objects – create instances of types (2021), Bill Wagner says "records are types with built-in

behavior for value-based equality". The origins for record types in C# could be traced to C++. C++ is a language well known for its support for records, and C# was designed in similarity to C++. Along with object-oriented features, C# uses features from functional programming languages.

C#'s similarities to functional programming languages can be seen through similar features. In Microsoft Learn, Pattern matching overview (2022), Bill Wagner explains how pattern matching works in C#. Wagner says, "C# pattern matching provides more concise syntax for testing expressions and taking action when an expression matches". In Pattern matching overview, Wagner goes over scenarios where pattern matching is used in C#. This includes null checks, type tests, comparing discrete values, etc. C#'s origins for pattern matching could be drawn back to the first developments of this feature in the languages COMIT and SNOBOL. Along with similarities to other languages, C# includes features that are unique to itself.

One feature that is unique to C# is asynchronous programming and the key words 'async' and 'await'. Asynchronous programming allows the user to write non-blocking code, which is useful when dealing with operations that take more time to complete. According to Microsoft Learn, Asynchronous programming with async and await (2023), Bill Wagner says, "You add the 'async' modifier to a method declaration to declare the method is asynchronous. The 'await' keyword provides a non-blocking way to start a task, then continue execution when the task completes.". In Asynchronous programming with async and await, Bill Wagner gives an example by modeling asynchronous programming after a list of steps to make breakfast.

Asynchronous programming is a major feature of C# that was important to its development. It was one of the first languages to use the 'async' and 'await' commands, and this influenced other programming languages to follow such as JavaScript, Python, and Haskell. The

origins of asynchronous programming in C# could be traced back to F#, which was the first language to use asynchronous programming.

Another powerful feature that is unique to C# is the language-integrated query, or LINQ. According to Microsoft Learn, Language Integrated Query (LINQ) (2023), Bill Wagner says, “LINQ is the name for a set of technologies based on the integration of query capabilities directly into the C# language.”. LINQ enables querying in C#, and it makes a query a first-class language construct. Users can directly query and manipulate data from different sources using a similar syntax to SQL all within C# programs. Ben Wagner goes into more depth about LINQ in Microsoft Learn, Language Integrated Query (LINQ) (2023).

LINQ is one of C#’s most powerful major features and its development marked big advancements in data querying and manipulation. Although languages before had ways to query and manipulate data in their own language, they weren’t as integrated and powerful as C# and LINQ. The origins of LINQ could have come from languages like SQL (mainly used for querying) or the programming language Haskell, which had a similar querying mechanism developed in its code.

The programming language C# has various different major functions that make it similar and different from other programming languages. It has features similar to object-oriented languages, such as the ability to create objects and the support for record types. It has features similar to functional programming languages, such as the use of pattern matching. Although these similarities draw ties to other languages, the ‘async’/’await’ keywords for asynchronous programming and LINQ make C# unique.