DAy 9 Question

1. Why is it recommended to explicitly assign values to enum members in some cases?

* Default value of Enums =0
* enhances readability and reduces ambiguity.

1. What happens if you assign a value to an enum member that exceeds the underlying type's range?

* the compiler will throw an error.

1. What is the purpose of the virtual keyword when used with properties?

* derived classes to provide a custom implementation for a property and can override in base class.

1. Why can’t you override a sealed property or method?

* Method cannot be overridden by derived classes , method is explicitly marked as final.

1. What is the key difference between static and object members?
2. Can you overload all operators in C#? Explain why or why not.
3. When should you consider changing the underlying type of an enum?
4. Why can't a static class have instance constructors?
5. What are the advantages of using Enum.TryParse over direct parsing with int.Parse?
6. What is the difference between overriding Equals and == for object comparison in C# struct and class ?
7. Why is overriding ToString beneficial when working with custom classes?
8. Can generics be constrained to specific types in C#? Provide an example
9. What are the key differences between generic methods and generic classes?
10. Why might using a generic swap method be preferable to implementing custom methods for each type?
11. How can overriding Equals for the Department class improve the accuracy of searches?
12. Why is == not implemented by default for structs?