Part 2

Object-Oriented Programming

C# is an object-oriented programming language, meaning that the code we write is typically organized into little blocks, each responsible for a small slice of the whole program. Each object has its own data (variables) and capabilities (methods), and the objects all work together to form a cohesive system. Without an understanding of object-oriented programming in C#, our knowledge of the language is far from complete. This is the topic of Part 2.

We will look at the following:

- Introduce what object-oriented programming is about (Level 15).
- Discuss the many ways C# lets you create custom types: enumerations (Level 16), tuples (Level 17), classes (Level 18), interfaces (Level 27), structs (Level 28), records (Level 29), and generics (Level 30).
- Discuss the key points of object-oriented programming: information hiding (Level 19), properties (Level 20), static members (Level 21), null references (Level 22), inheritance (Level 25), and polymorphism (Level 26).
- Get some practice designing and building larger object-oriented programs (Levels 23, 24, and 31).
- A final level describing some common types that come with .NET's Base Class Library, including **`Random`**, **`DateTime`**, **`TimeSpan`**, lists, and dictionaries (Level 32).