

Practice C#

C# and Practice

2025-01-21

Abstract

“C# 환경 설정과 기본 구조를 시작으로, 값 형식과 참조 형식을 포함한 다양한 데이터 형식을 다루며 문자열과 연산자를 활용해 데이터를 가공하는 방법을 익힙니다. 이어서 분기문, 반복문, 패턴 매칭을 통해 로직 제어의 기초를 다지고, 메소드와 클래스, 인터페이스, 추상 클래스를 통해 객체 지향 프로그래밍의 핵심 개념을 학습합니다. 또한 프로퍼티, 배열, 컬렉션, 인덱서, 제네릭, 예외 처리 기법 등을 배우며 견고하고 유연한 코드를 작성할 수 있는 기술을 쌓게 됩니다. 더 나아가 대리자, 이벤트, 람다식, LINQ를 통해 함수형 프로그래밍 및 데이터 쿼리 기법을 학습하고, 리플렉션과 애트리뷰트, dynamic 형식, 파일 입출력, 스레드와 태스크, WinUI, 네트워크 프로그래밍, 가비지 컬렉션까지 폭넓게 살펴봄으로써 C# 개발 전반에 필요한 지식을 고루 익힐 수 있도록 구성하였습니다.”

C# 학습 내용

Syntax Part.1

- 1. C#과 .NET
- 2. Visual Studio와 Solution
- 3. Data, Value 그리고 Ref
- 4.

-
- 분기문
 - if, else, else if
 - switch 문
 - switch 식
 - 반복문
 - while
 - do while
 - for
 - foreach
 - break
 - continue
 - goto

- 패턴 매칭
 - 선언 패턴
 - 형식 패턴
 - 상수 패턴
 - 프로퍼티 패턴
 - 관계 패턴
 - 논리 패턴
 - 괄호 패턴
 - 위치 패턴
 - var 패턴
 - 무시 패턴
 - 목록 패턴
- 메소드
 - return -매개 변수에 대하여
 - 참조에 의한 매개 변수 전달
 - 메소드의 결과를 참조로 반환하기
 - 출력 전용 매개 변수
 - 메소드 오버로딩
 - 가변 개수의 인수
 - 명명된 인수
 - 선택적 인수
 - 로컬 함수
- 클래스
 - 생성자와 종료자
 - * 생성자
 - * 종료자
 - 정적 필드와 메소드
 - 객체 복사하기
 - this 키워드
 - * this() 생성자
 - 접근 한정자로 공개 수준 결정하기
 - 상속으로 코드 재활용하기
 - 기반 클래스와 파생 클래스 사이의 형식 변환
 - 오버라이딩과 다형성
 - 메소드 숨기기
 - 오버라이딩 봉인하기
 - 읽기 전용 필드
 - 중첩 클래스
 - 분할 클래스
 - 확장 메소드
 - 구조체
 - 튜플

Syntax Part.2

- 인터페이스와 추상 클래스

- 프로퍼티
- 배열과 컬렉션 그리고 인덱서
- 일반화 프로그래밍
- 예외 처리하기
- 대리자와 이벤트
- 람다식
- LINQ
- 리플렉션과 애트리뷰트
- dynamic 형식
- 파일 다루기
- 스레드와 태스크
- WinUI
- 네트워크 프로그래밍
- 가비지 컬렉션