

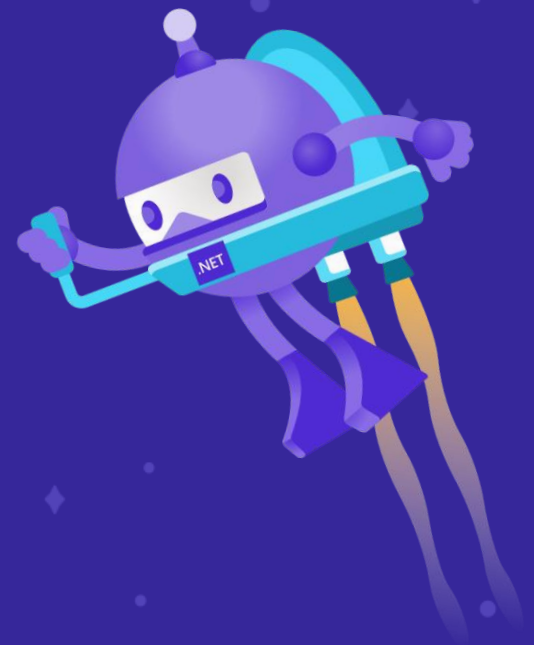
# .NET Conf 2022 x Seoul

---

풀 스택과 **사랑**에 빠질 준비, 되셨나요?

# C# 으로 하는 정규표현식

고요한(kimo87@gmail.com)



# 발표에 앞서

- .NET Interactive를 사용하여 발표를 준비하였습니다.
- 간단한 이론
- 다양한 예제를 해볼 예정입니다.
- 이 발표를 통하여 완전한 정규표현식을 이해할 수 는 없습니다.
- .NET에서 [System.Text.RegularExpressions](#) 를 사용할 예정입니다.

# 정규표현식이란?

- 정규식은 텍스트를 강력하고 유연한 방법으로 처리합니다.
- 정규식의 광범위한 패턴 일치 표기법을 사용하면 많은 양의 텍스트를 빠르게 구문 분석하여 처리 할 수 있습니다.
- 특정 문자 패턴을 찾습니다.
- 텍스트의 유효성을 검사하여 미리 정의된 패턴(예: 전자 메일 주소)과 일치하는지 확인합니다.
- 텍스트 하위 문자열을 추출, 편집, 바꾸기 또는 삭제합니다.

# 정규표현식 역사?

- 정규 표현식은 1950년대 수학 분야 연구에서 유래 되었다. 수년이 지난 후 이 연구에서 파생된 원리와 아이디어는 유닉스 세계에서 펄(Perl) 언어와 grep 같은 유틸리티로 사용되어 왔다. 현재 모든 컴퓨팅 환경에서 정규표현식을 여러 형태로 지원하고 있다.

# 정규표현식을 사용하는 이유?

- 데이터 검증 및 데이터 추출 시 많이 사용하고 있습니다.
- 조건문으로 코드가 길어지고 복잡해 지지만 정규표현식의 패턴으로 코드가 짧아지고 효율적인 코드가 작성이 가능합니다

# Regex class의 메서드

- `IsMatch()` - 주어진 문자열에서 일치하는 부분이 있는지 확인한다.
- `Match()` - 일치하는 부분을 하나 찾아 `Match` 객체로 반환한다.
- `Matches()` - 일치하는 부분을 모두 찾아 `MatchCollection` 객체로 반환한다.
- `Replace()` - 주어진 문자열에서 치환을 수행한다.
- `Split()` - 문자열을 여러 문자열 배열로 나눈다.

# 패턴매칭(Match)

- 표현식을 이용하여 값의 패턴이 일치 하는지 확인 할 수 있습니다.
- 문자열의 한글인지 영어인지 혹은 특수문자인지 검토 및 문자열 사이(A~Z)를 검토 할 수 있다.
- 문자수를 검출 하여 체크를 할 수 있습니다.



# 정규표현식 살펴보기

- 정규표현식은 문자클래스 , 앵커 , 그룹화 구문, 수량자 등 다양한 표현식들이 있습니다.
- [] : 대괄호 안에 있는 문자를 검색합니다. -(하이픈)을 사용하여 범위를 지정할 수 있습니다.  
ex ) [1-9] ->[1,2,3,4,5,6,7,8,9]
- | : or 연산자 ex) te(s|r)t ->test , tert 를 검색
- .(dot) : 마침표(.)는 어떤 문자든지 문자 하나와 일치한다. 숫자 특수 문자 상관없음
- / : 특수 문자가 아닌 문자 기호 표현 ex) / / , / . -> ./ 등을 검색합니다.

# 정규표현식 살펴보기

- `^(캐럿)` : 이 문자 바로 뒤에 있는 문자나 범위에 있는 문자를 제외 시킨다.
- `/ d` : 숫자 하나(`[0-9]`와 같다)
- `/ D` : 숫자를 제외한 문자 하나 (`[^0-9]`와 같다)
- `/ w` : 대소문자와 밑줄을 포함하는 모든 영숫자(`[a-zA-Z0-9_]`와 같다)
- `/ W` : 영문 숫자 및 밑줄이 아닌 모든 문자(`[^a-zA-Z0-9_]`)와 같다.
- `*(반복)` : 0번 이상 반복되는 매치
- `{m,n}` (반복) : 최소 m번 이상 최소 n번 이하 반복을 할수 있다.

# 정규표현식 살펴보기

- / b : 단어 경계를 시작합니다. 단어에 앞에 있는 문자열만 검색  
ex) / b dot : dotnet, homedot -> dotnet, homedot
- + : 문자가 하나 이상 일때  
ex) "a[ / w]+b"는 "ab", "aab", "aaab"를 포함하지만 "b"는 포함하지 않는다.
- ? : 있거나 혹은 없거나 ex) hous?e -> houe , house
- / s : 스페이스 공백
- () : 그룹으로 묶을 수 있습니다.

# 정규표현식의 장 단점

- 장점으로는 코드 작성이 용이하며 정규표현식을 알면 다른 언어에서도 똑같이 사용하실 수 있습니다.
- 단점으로는 표현식을 숙지 하지 않으면 이해 하기 힘들며 성능이 떨어진다.

# 정규표현식 더 알아보기

- MSDN 정규표현식
  - <https://docs.microsoft.com/ko-kr/dotnet/standard/base-types/regular-expressions>

정규표현식 연습사이트

- <https://regexr.com/>

RELEASED

# 발표

- 정규표현식을 왜 써야하는지?
- 간단한 정규표현식 문법을 알았던 시간



고맙습니다!

