



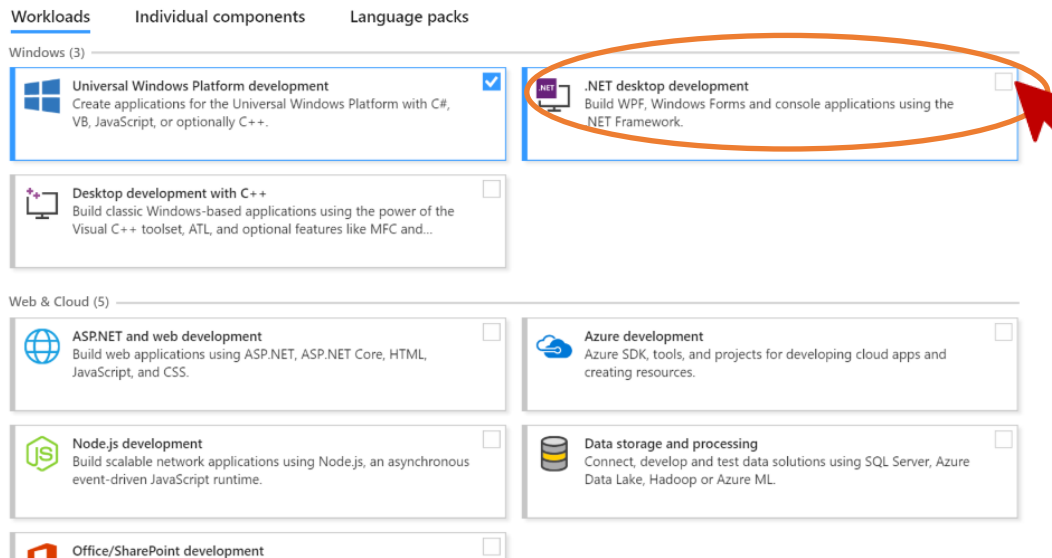
# Windows App 만들기

C# - Day1

천해철 guruchun@gmail.com

# Visual Studio는 설치하셨나요?

- Visual Studio 2022 커뮤니티 버전  
<https://visualstudio.microsoft.com/ko/vs/community/>



이것만 체크해서  
설치하면 됩니다.

# Windows App을 만드는 여러가지 방법

- Microsoft에 의존하는 방식: Windows Only 또는 Mono on Linux
  - Win32, MFC, WinForm, WPF, UWP
- Microsoft에 독립적인 방식: Cross-Platform
  - Qt, PyQt / GTK, PyGTK
  - Electron
  - Swing, JavaFX, Eclipse RCP

## 고려사항

- 프로그래밍 언어
- 프로그래밍 난이도

# 프로그래밍 언어

Javascript > Python > C#

- C#: Basic처럼 쉽고 C처럼 익숙한 문법, 현대 프로그래밍 언어의 장점을 수용하며 발전 중
- Javascript & HTML5: 가장 빠르게 발전하고 가장 널리 쓰이는 방식, 완전 대세
- C, C++: 프로그래머의 능력에 의존함, 높은 자유도로 개발 가능 - 대규모 APP에서 사용 금지
- Python: 배우기 쉽고 잘 설계된 현대 프로그래밍 언어, 고퀄의 GUI Toolkit은 QT가 유일함
- JAVA: Cross-Platform에서 동작하는 Desktop Application 개발 (Legacy Only)

# 프로그래밍 난이도

Winform > WPF > Electron

- WinForm: 대충 만들어도 그럴듯 함, 안이빔, 복잡하고 규모가 커질 수록 고민이 많아짐
- WPF: 코딩의 양을 획기적으로 줄여 줌, UI Designer와 분업할 경우에 좋음, 아키텍처 중요  
대규모 APP에는 Prism같은 별도의 App. Framework 사용
- PyQt: 그럭저럭 다 좋음. Python을 알아도 QT는 추가로 배워야 함.  
Linux를 사용하는 제어패널 HMI용으로 가장 먼저 검토했으나 라이선스가 애매해서 포기  
Windows용 APP이라면 굳이 PyQt를 써야할 필요는 없음
- Electron: Web기술에 대한 Background가 없다면 접근하기 어려움  
하지만 가장 Hot, Popular, Trendy, Innovative하므로 배워두면 피가 되고 살이 됨
- Swing/JavaFX, Eclipse RCP: Cross-Platform but Not QT  
특이한 사람, 특이한 업종(Legacy)에서 쓰이나 신규 APP은 Electron으로 넘어가는 추세  
IntelliJ, Eclipse

Linux용으로 PyGTK도 대안이나 IDE 없음

Teams, VS Code, Notion, Slack

# 답은 이미 정해 놓았음...

## c# & WinForm

- 수소시스템용 관리콘솔을 개발할 때 이미 고민했던 문제, 지금 다시 봐도 ...
- 임시로 만들었지만 이미 쓰이는 프로그램들
  - SOFC 관리콘솔 (WinForm) 관리콘솔 통합 예정
  - H2 관리콘솔 (WPF) PEMFC Gen2용으로 관리콘솔 Renewal 예정
  - 기타 WinFrom APP들
    - MonHMI: PEMFC 전장박스 로그 메시지 출력
    - WinModbus: Modbus/TCP Server/Client APP, XML로 정의한 Modbus MAP 사용 가능
    - CANManger: CAN Message 송신/수신, DBC 파일 로딩
    - 인버터/온도모듈 시뮬레이터, 온습도장치 모니터링통신장치(BOP) 시뮬레이터 통합 예정

## 참고) C# 버전과 닷넷 지원현황

C# 버전	플랫폼	표준	출시일	VS 버전	변경 사항
1.0	.NET Framework 1.0	<a href="#">ECMA-334</a>	2002년 1월	.NET 2002	<a href="#">링크</a>
1.1	.NET Framework 1.1		2003년 4월	.NET 2003	-
2.0	.NET Framework 2.0		2005년 11월	2005	<a href="#">링크[8]</a>
3.0	.NET Framework 2.0~3.5	-	2007년 11월	2008	<a href="#">링크[9]</a>
4.0	.NET Framework 4.0	-	2010년 4월	2010	<a href="#">링크[10]</a>
5.0	.NET Framework 4.5	<a href="#">ECMA-334</a>	2012년 8월	2012	<a href="#">링크[11]</a>
6.0	.NET Framework 4.6 .NET Core 1.0~1.1	-	2015년 7월	2015	<a href="#">링크[12]</a>
7.0	.NET Framework 4.6.2	-	2017년 3월	2017 v15.0	<a href="#">링크[13]</a>
7.1	.NET Framework 4.7 .NET Core 2.0	-	2017년 8월	2017 v15.3	<a href="#">링크[14]</a>
7.2	.NET Framework 4.7.1	-	2017년 11월	2017 v15.5	<a href="#">링크[15]</a>
7.3	.NET Framework 4.7.2 .NET Core 2.1~2.2	-	2018년 5월	2017 v15.7	<a href="#">링크[16]</a>
8.0	.NET Framework 4.8 .NET Core 3.0	-	2019년 9월	2019 v16.3	<a href="#">링크[17]</a>
9.0	.NET 5.0	-	2021년 4월	2019 v16.9	<a href="#">링크[18]</a>
10.0	.NET 6 <a href="#">Introducing .NET 5</a>	-	2021년 11월	2022	<a href="#">링크</a>
11.0	.NET 7	-	-	2022 Preview	<a href="#">링크</a>

# Tutorial – Day1

Day1

파일:

...

열기

파일 내용 출력



# Tutorial – Day1

- Button
- TextBox
- Label
- ComboBox
- PictureBox
- OpenFileDialog

# 참고) C# 문법, 학습 자료

## 웹페이지

- [.NET용 Windows Forms 및 WPF의 데스크톱 가이드](#)
- [C# / 문법 - 나무위키](#)
- [C# 프로그래밍 배우기 \(csharpstudy.com\)](#)

## 동영상

- [\[무료\] \[C# 초보 강좌\] 예제로 배우는 C# \(inflearn.com\)](#)
- [\[무료\] C# 프로그래밍 \(inflearn.com\)](#)
- [C# 교과서 마스터하기 - Microsoft MVP 저자 직강 - YouTube](#)