

01주차

강의OT

강의개요

- 과목명: IoT소프트웨어응용
- 이수구분: 전공선택
- 학점 : 3학점
- 담당: 박상민(E-mail: smpark@bible.ac.kr / 사무실: 복음관 512호, 02-950-5597)
- 수업시간: 월 09:00-11:40 (1교시)
- 수업방식: 실험/실습
- 질의 응답? 이메일 혹은 LMS쪽지
- 주제어: 아두이노, IoT 프로그래밍
- 수업에 관한 모든 공지는 LMS **공지사항**에 업로드할 예정!

학습개요

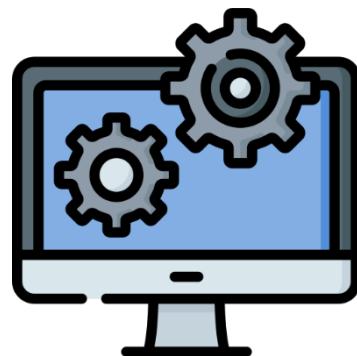
■ 수업 목적

- 사물인터넷 환경을 구성하기 위하여 사물인터넷 개념에 대해 학습
- 아두이노와 다양한 센서를 연결 방법, 아두이노의 통신 기법, 아두이노 프로그래밍 등을 학습
- 라즈베리파이와 아두이노를 연결하는 방법에 대해 학습하고 실제 센서 데이터를 저장하고 실제 사물인터넷 환경을 구현

■ 주요 내용

- 사물인터넷 개념을 학습하고, 사물인터넷에 필요한 통신 방법에 대해 학습
- 아두이노의 기초적인 리눅스 환경을 학습하여 사물인터넷 환경을 구현
 1. 아두이노 회로도 작성방법과 아두이노 프로그래밍 방법에 대해 이해하고 학습
 2. 사물인터넷이 개념과 근거리 통신망으로 분류되는 와이파이 통신의 개념에 대해 이해
 3. 아두이노 보드인 UNO를 사용하여 기초적인 리눅스 환경을 학습하고, 아두이노에서 생성되는 데이터를 UNO에 저장하고 컴파일하는 방법을 학습

학습환경



Server



⚠ 기본적인 프로그래밍 오류

⚠ port, ip 등 네트워크 오류
회로 설계 오류

강의자료

- 강의노트: LMS에 업로드할 예정
- 참고교재
 - 한 권으로 끝내는 아두이노 입문 + 실전, 서민우 , 박준원 지음, 앤써북 (2021)
 - 익스플로링 아두이노, Jeremy Blum 지음, 한빛아카데미 (2021)
 - 모두의 아두이노, 다카모토 다카요리 지음, 모두의 시리즈 (2016)
 - 한 권으로 끝내는 아두이노와 파이썬으로 52개 작품 만들기, 장문철 , 박준원 지음, 앤써북 (2022)

평가방식

■ 출석 (10%) : 만점 10점

- 결석 1번 = -1점
- 지각 3번 = 결석 1번

■ 중간고사 (30%) : 이론

- Half-Open Book 실시: A4 용지 1장(앞/뒤 포함)에 배운 내용 요약하여 가져오기
- 단, 시험지와 함께 요약한 A4용지 제출

■ 기말프로젝트(30%)

- 중간보고서, 발표 및 최종소스코드, 발표자료(25%)
- 학생평가(5%)

■ 과제(20%)

- ✓ 중간고사와 기말 프로젝트중 하나라도 결석시 F
- ✓ 과제 베끼기 적발시 중복된 학생 모두 0점
- ✓ 모든 과제는 만점이 아닐 경우 피드백 반드시 확인!

주차별 수업

| 9. 주차별 수업계획 | | | | |
|-------------|-------|-------------------|---|--|
| 주 | 일자 | 단원명 | 강의내용 | 교수학습방법, 평가방법 및 과제물 (실습, 이론 등) |
| 1주 | 09/01 | 강의 오리엔테이션 IoT란 | - 강의 운영안내 - IoT 개념 - IoT 역사 | - 이론 강의 - 질의 응답 |
| 2주 | 09/08 | 아두이노와 친해지기 | <ul style="list-style-type: none"> ▣ 아두이노의 개념 - 아두이노 보드 종류 - Arduino IDE 설치 및 환경설정 - 아두이노 스케치 구조 이해 ▣ 아두이노 기초 프로그래밍 - Serial.println 실습 - Serial.read 실습 | - 이론 및 실습 강의 - 질의 응답 |
| 3주 | 09/15 | 아두이노 출력 모듈 | <ul style="list-style-type: none"> ▣ LED 모듈 - LED 회로 구성 - LED 깜빡임, 밝기 조절 ▣ RGB 모듈 - RGB 모듈 회로구성 ▣ 3색 RGB LED 모듈 실습 | - 이론 및 실습 강의 - 질의 응답 |
| 4주 | 09/22 | 아두이노 출력 모듈 | <ul style="list-style-type: none"> ▣ 세그먼트 LED 모듈 회로구성 ▣ 0.96인치 OLED I2C 실습 ▣ LCD 모듈 회로구성 | - 이론 및 실습 강의 - 질의 응답 |
| 5주 | 09/29 | 아두이노 출력 모듈 | <ul style="list-style-type: none"> ▣ 수동부저 회로구성 ▣ 능동부저 회로구성 ▣ DS1302 RTC 시계 모듈 실습 | - 이론 및 실습 강의 - 질의 응답 |
| | | 아두이노 입력 모듈 | <ul style="list-style-type: none"> ▣ 택트 스위치 회로구성 ▣ 볼 스위치 LED 실습 | - 이론 및 실습 강의 - 질의 응답 |
| 6주 | 10/06 | 아두이노 입력 모듈 | <ul style="list-style-type: none"> ▣ 초음파 센서 회로구성 ▣ 조도 센서 회로구성 ▣ 온습도 센서 회로구성 ▣ 화재 감지 센서 실습 ▣ MPU6050 자이로 센서 모듈 실습 <p>온라인 보강 10/13 자정까지 시청 완료 * 주석연휴 (10/06)</p> | - 이론 및 실습 강의 - 질의 응답 - 하이브리드러닝 |
| 7주 | 10/13 | 아두이노 모터 실습 | <ul style="list-style-type: none"> ▣ 스텝링 모터 실습 ▣ L298N DC 모터 실습 | - 이론 및 실습 강의 - 질의 응답 - 자기주도적 학습 Homework |
| 8주 | 10/20 | 1교시 : 중간 점검(75분) | ▣ Homework 풀이 및 Summary | - 이론 및 실습 강의 - 질의 응답 |
| | | 2 교시 : 중간고사(75분) | ▣ 지필고사 | - 중간 지필 고사 |

주차별 수업계획

| | | | | |
|-----|-------|-----------------------|--|--|
| 9주 | 10/27 | 아두이노 통신 모듈 | <ul style="list-style-type: none"> ▣ 아두이노 웹서버 구현 | <ul style="list-style-type: none"> - 이론 및 실습 강의 - 질의 응답 - 기말 프로젝트 팀 설정(실제적 과제 적용) |
| 10주 | 11/3 | 아두이노 통신 모듈 | <ul style="list-style-type: none"> ▣ 아두이노 간 통신 ▣ USB 시리얼 통신 실습 ▣ TTL to RS232/485 통신 실습 | <ul style="list-style-type: none"> - 이론 및 실습 강의 - 질의 응답 |
| 11주 | 11/10 | 아두이노 통신 모듈 | <ul style="list-style-type: none"> ▣ WiFi 모듈 회로 구성 ▣ 블루투스 HC-05 모듈 실습 ▣ RFID-RC522 RF 통신 실습 | <ul style="list-style-type: none"> - 이론 및 실습 강의 - 질의 응답 |
| 12주 | 11/17 | 아두이노 통신 모듈 | <ul style="list-style-type: none"> ▣ 적외선 통신 ▣ 적외선 리모컨 실습 | <ul style="list-style-type: none"> - 이론 및 실습 강의 - 질의 응답 |
| 13주 | 11/24 | IoT 프로젝트 (PBL) | <ul style="list-style-type: none"> ▣ 무인점포 만들기 ▣ 환경정보 센서 모니터링 시스템 | <ul style="list-style-type: none"> - 이론 및 실습 강의 - 질의 응답 |
| 14주 | 12/1 | IoT 프로젝트 (PBL) | <ul style="list-style-type: none"> ▣ 기말 프로젝트 및 중간보고서 점검 | <ul style="list-style-type: none"> - 이론 및 실습 강의 - 질의 응답 - 팀 별로 기말 프로젝트 고도화 학습 |
| 15주 | 12/8 | 1교시: 기말 점검(75분) | <ul style="list-style-type: none"> ▣ Summary | <ul style="list-style-type: none"> - 이론 및 실습 강의 - 질의 응답 |
| | | 2교시 : 기말 프로젝트 발표(75분) | <ul style="list-style-type: none"> ▣ 프로젝트 발표 | <ul style="list-style-type: none"> - 교수 및 학생 평가 |



01주차

IoT 소개

IoT란?

- Internet of Thing의 약자 / 사물인터넷
- 개념은 니콜라 테슬라에 의해 처음 등장

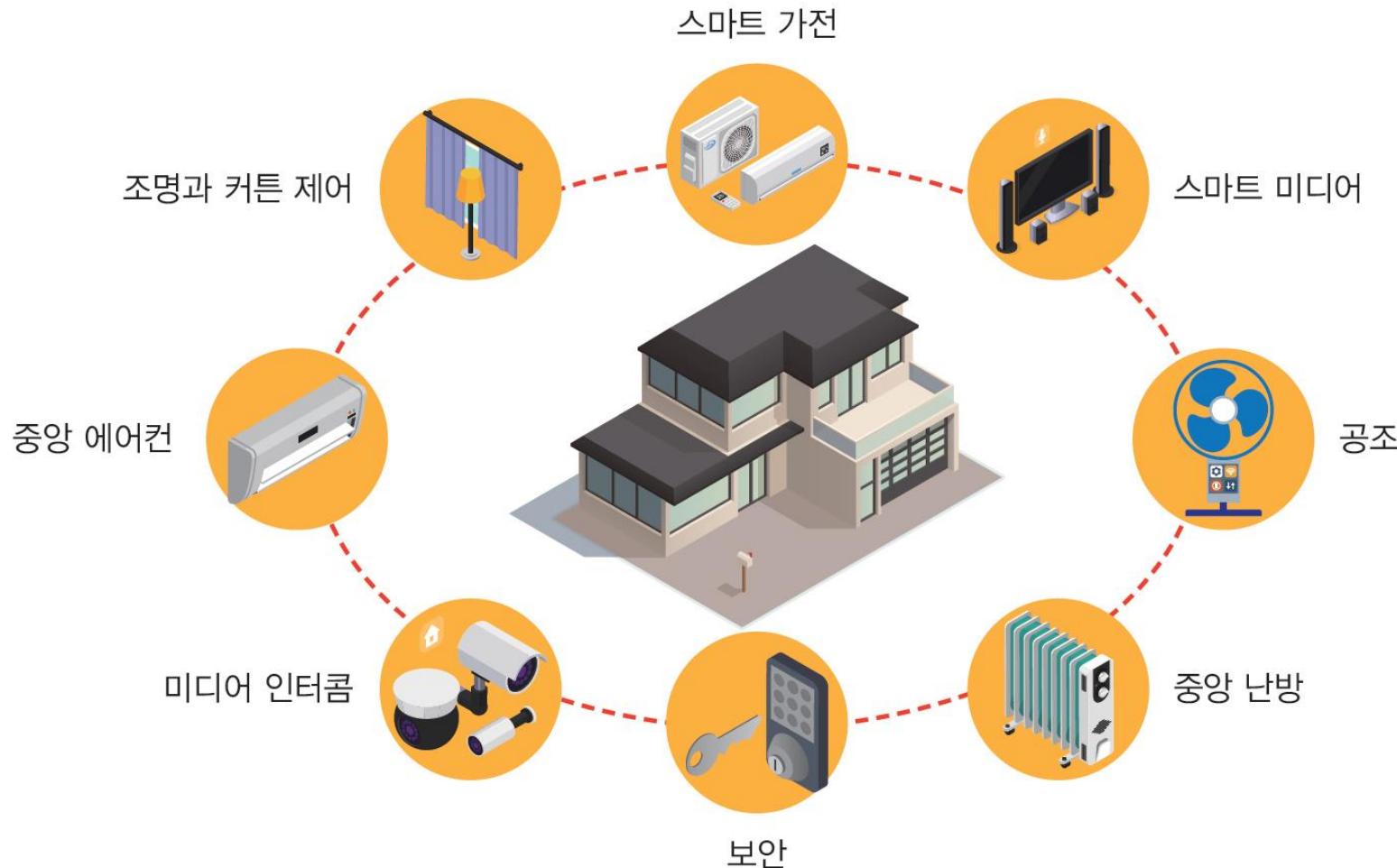
“무선이 완벽하게 적용되면, 지구 전체는 커다란 두뇌로 변모할 것이다. 그래서 모든 사물은 실제적이고 리드미컬한 전체에 대한 구성 요소가 될 것이다.”

-니콜라테슬라, 1926년 Colliers 기고문

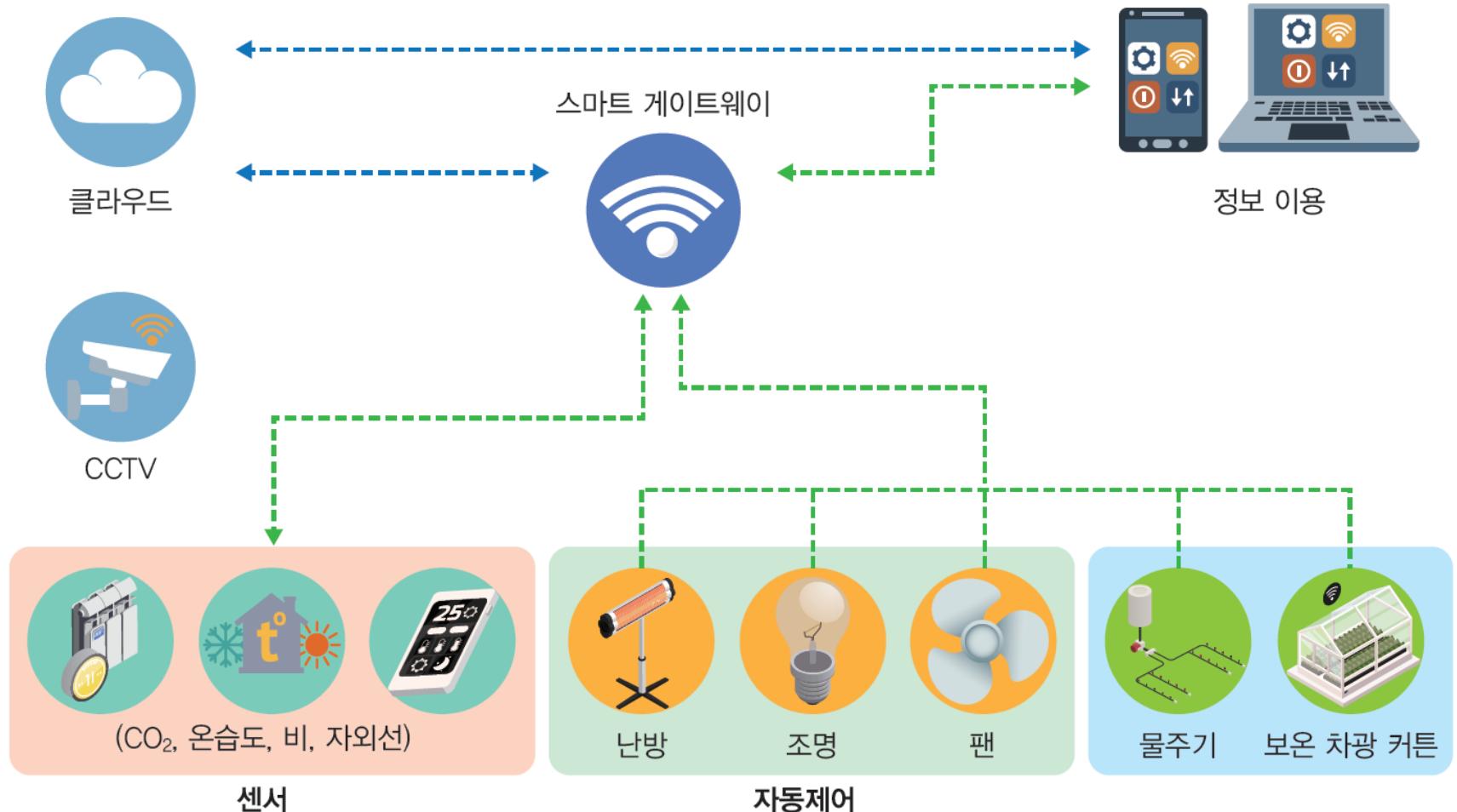
- IoT란 용어는 MIT의 케빈 애쉬튼이 1999년에 최초로 사용
- 2011 가트너 보고서에 게재되면서 미디어가 주목
- 사물인터넷의 정의

“개체나 센서, 일상용품이 네트워크에 연결되고 정보처리 능력을 갖게 되어,
인간의 개입이 거의 없이 정보를 생성, 교환, 소비하여 정해진 기능을 수행하는 것”

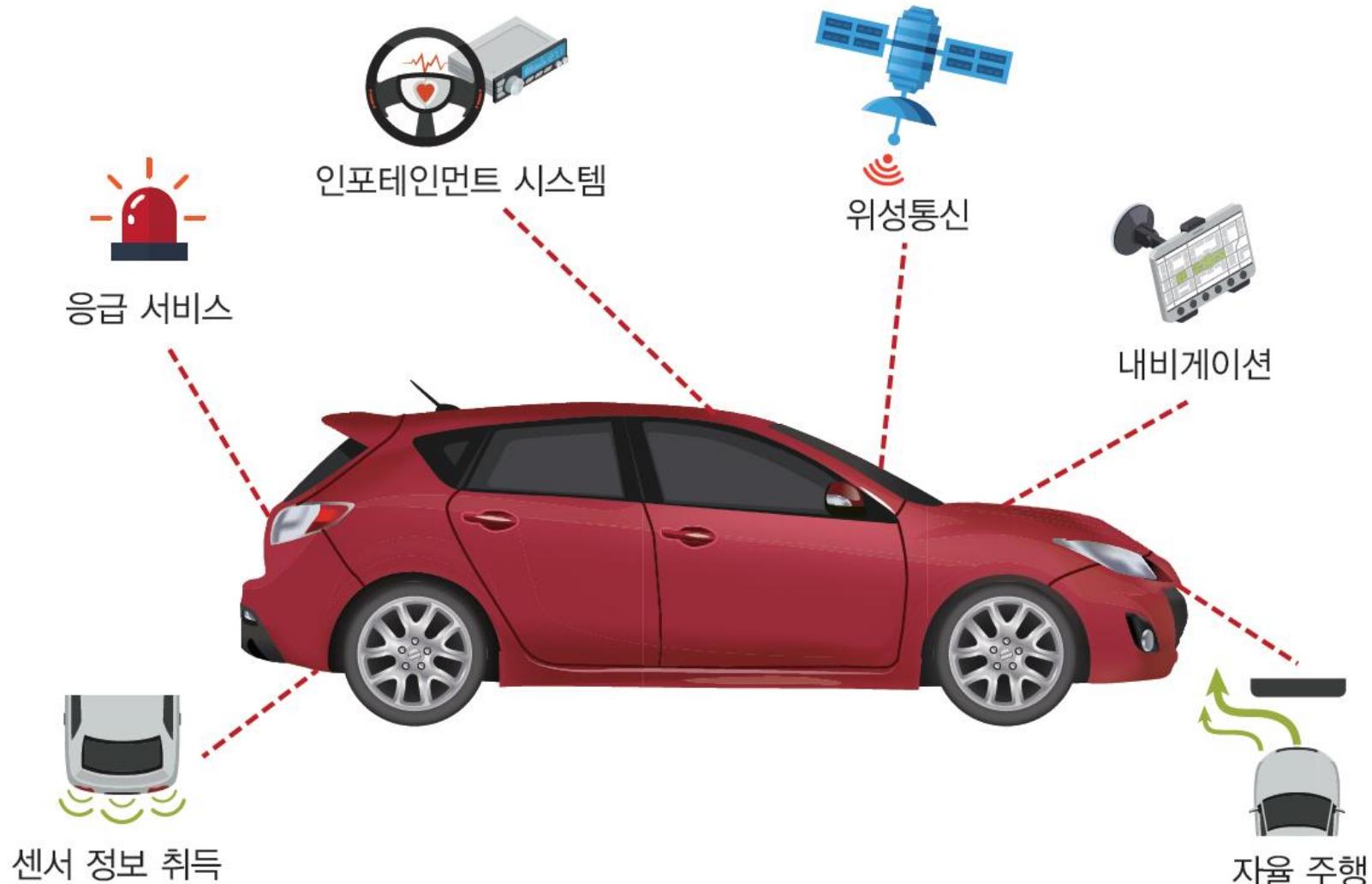
IoT 사례: 스마트 홈(Smart Home)



IoT 사례: 스마트 팜(Smart Farm)



IoT 사례: 스마트 카(Smart Car)

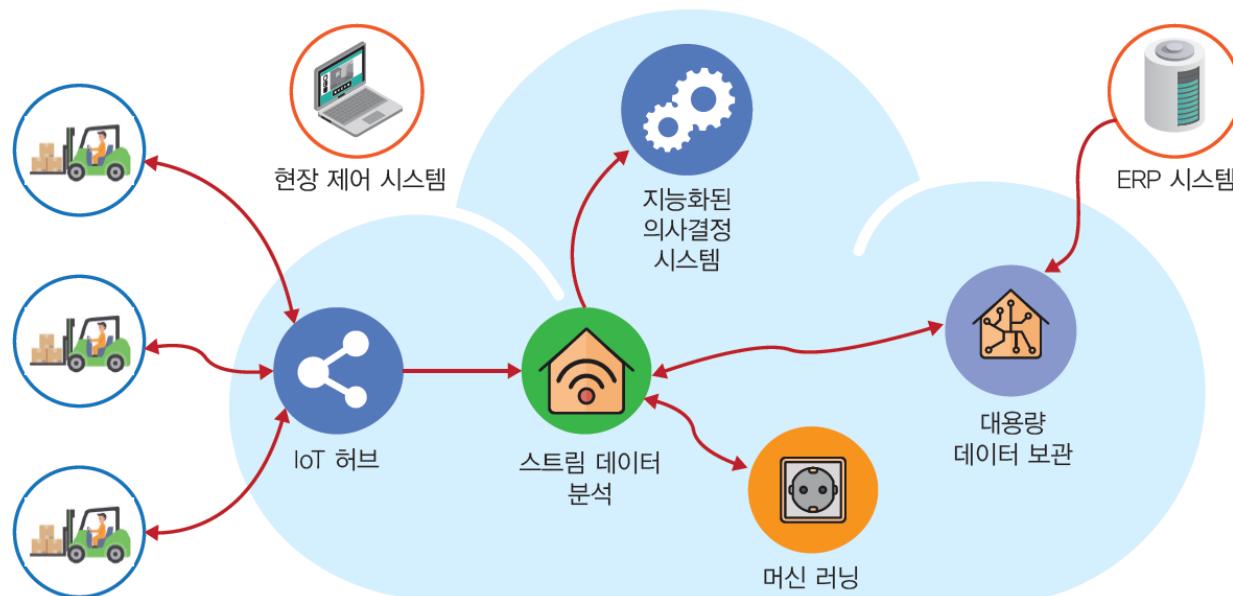


IoT 사례: 스마트 헬스케어(Smart Healthcare)



IoT 구조

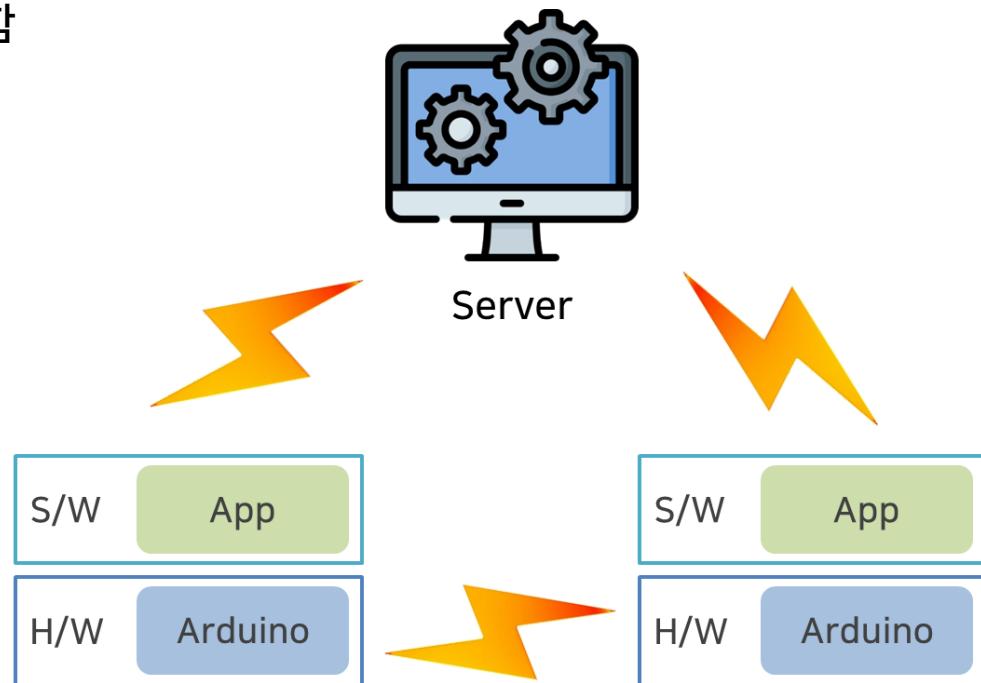
- 편의상 '기업형 IoT 시스템'과 '소규모 IoT 시스템'으로 분리
- 기업형 IoT 시스템의 특징
 - 대기업이나 공공 서비스를 위한 IoT는 기술 난이도가 높으며 개발과 운용 비용이 높음
 - IoT 시스템의 구성은 복잡하여 전문가의 참여가 필수
 - 데이터들은 전사적 자원 관리(ERP)시스템과 연결되어 의사결정에 영향



IoT 구조

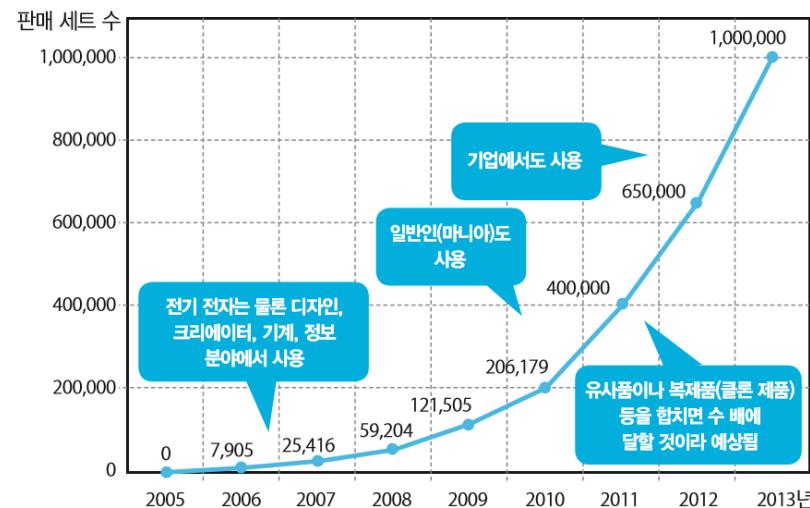
■ 소규모 IoT 시스템의 특징

- 비교적 구조가 간단하기 때문에 한 사람이 전체 시스템을 개발 및 관리
- 디바이스는 센서 정보를 클라우드 상의 브로커(서버)에 전달
- 서버는 수신한 정보를 다른 디바이스나 관련 시스템에 전송
- 본 과목은 소규모의 IoT 시스템을 목표로 함



아두이노 탄생과 배경

- 2005년 전기 전자를 전공하는 학생들이 손쉽게 공부할 수 있도록 값싼 교재용 마이컴 보드로 아두이노가 개발됨
- 아두이노는 기술의 장벽이 매우 낮아서 지금은 정보 처리나 기계, 디자인, 크리에이터 등 문·이과를 막론하고 다양한 학생들이 사용
- 이러한 보급 기세는 세계적으로 퍼져, 학생과 일반인은 물론 여러 기업의 기술자들도 사용
- 아두이노는 마이컴 보드의 표준



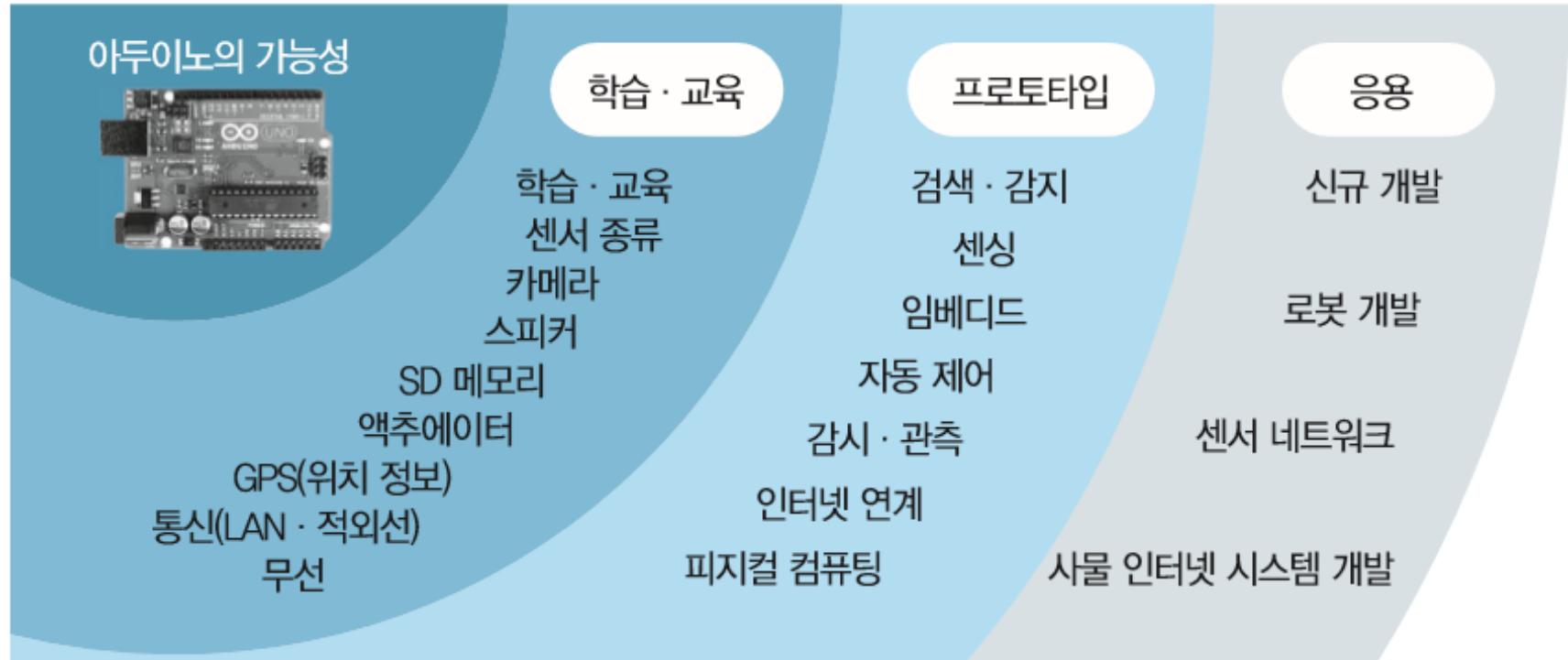
아두이노 탄생과 배경

- 보급이 확대된 이유 중 가장 큰 이유는 아두이노가 **오픈 소스 하드웨어로 개발됐다**는 것
- 오픈 소스 하드웨어로 개발되어 아두이노의 회로도와 기판도 등이 공개됨
- 이를 참고하여 누구나 간단히 클론 제품을 개발하고 판매할 수 있음
- 마이컴 보드를 프로그래밍하기 위한 소프트웨어 개발 환경(IDE, 통합 개발 환경)도 인터넷에서 무료로 내려받아 사용할 수 있음
- 뿐만 아니라 많은 사용자가 직접 만든 예제나 프로그램을 인터넷에 올리기 시작

아두이노 탄생과 배경

- PIC 마이컴이나 H8 마이컴은 전기 전자를 어느 정도 아는 사람만이 능숙하게 사용할 수 있음
- 아두이노를 사용하는 데는 그 정도로 **높은 수준의 전기 전자 지식이 필요하진 않아서**, PIC 마이컴이나 H8 마이컴 이상으로 널리 사용되게 됨
- 하드웨어와는 거리가 멀었던 정보 처리 교육에도 아두이노를 사용하기 시작
- 아두이노는 실제로 손에 쥐고 다룰 수 있는 교육 교재로 높이 평가됨
- 최근에는 일반 기업에도 아두이노를 사용하는 기술자가 늘고 있음
- 기업에서 기술을 연구하는 사람들이 제품 개발을 위한 시제품 제작에 사용

아두이노의 가능성



아두이노의 특징과 장점

- 가격이 싸고 쉽게 구할 수 있다
 - 아두이노 우노는 약 30,000원 정도에 살 수 있음
 - 인터넷 쇼핑몰에서도 살 수 있어서 전국 어디서든 쉽게 손에 넣을 수 있음
- 단기간에 개발할 수 있다
 - 브레드보드와 점퍼 와이어를 사용해 납땜 없이 간단하게 전기회로를 구성할 수 있기 때문
- 기술의 문턱이 낮다
 - 전기 전자 기술을 잘 몰라도 되고, 간단한 프로그래밍(스케치)으로 개발할 수 있으며 확장과 변경도 간단하고 빠르게 할 수 있음

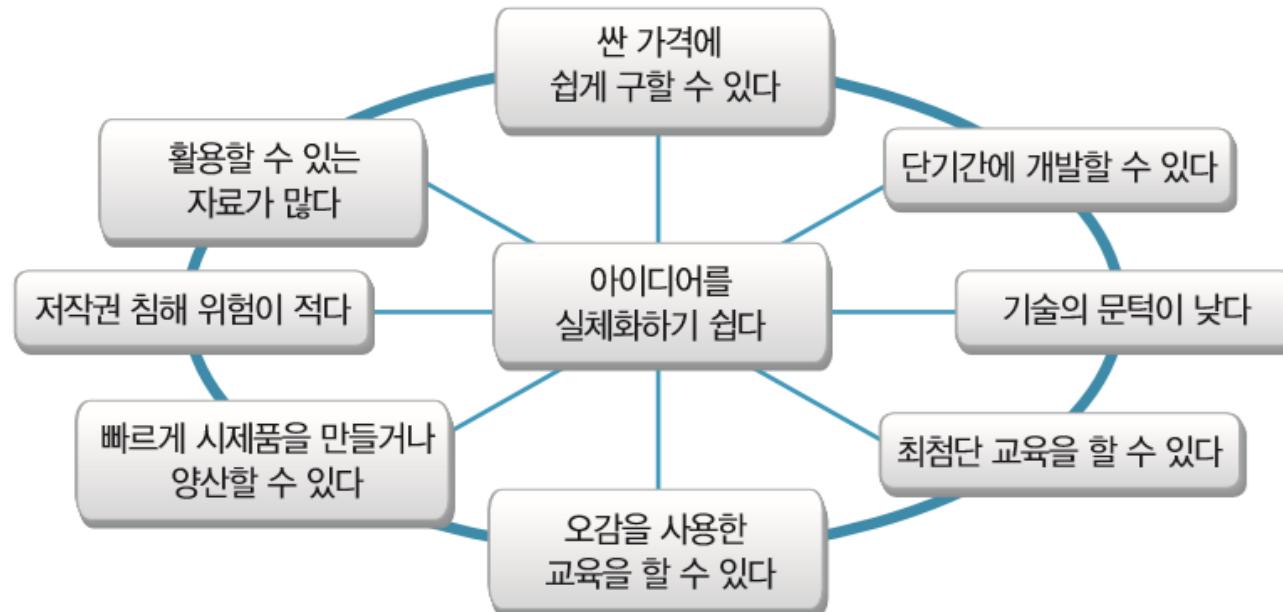
아두이노의 특징과 장점

- 활용할 수 있는 자료가 풍부하다
- 인터넷에는 아두이노에 관한 정보가 풍부함
- 국내뿐만 아니라 해외에서도 누군가가 새로운 부품을 사용해서 연결해 본 사례를 프로그램과 함께 올려놓기도 함
- 어떤 사이트에는 동영상이 있기도 하니 참고하기에 좋음

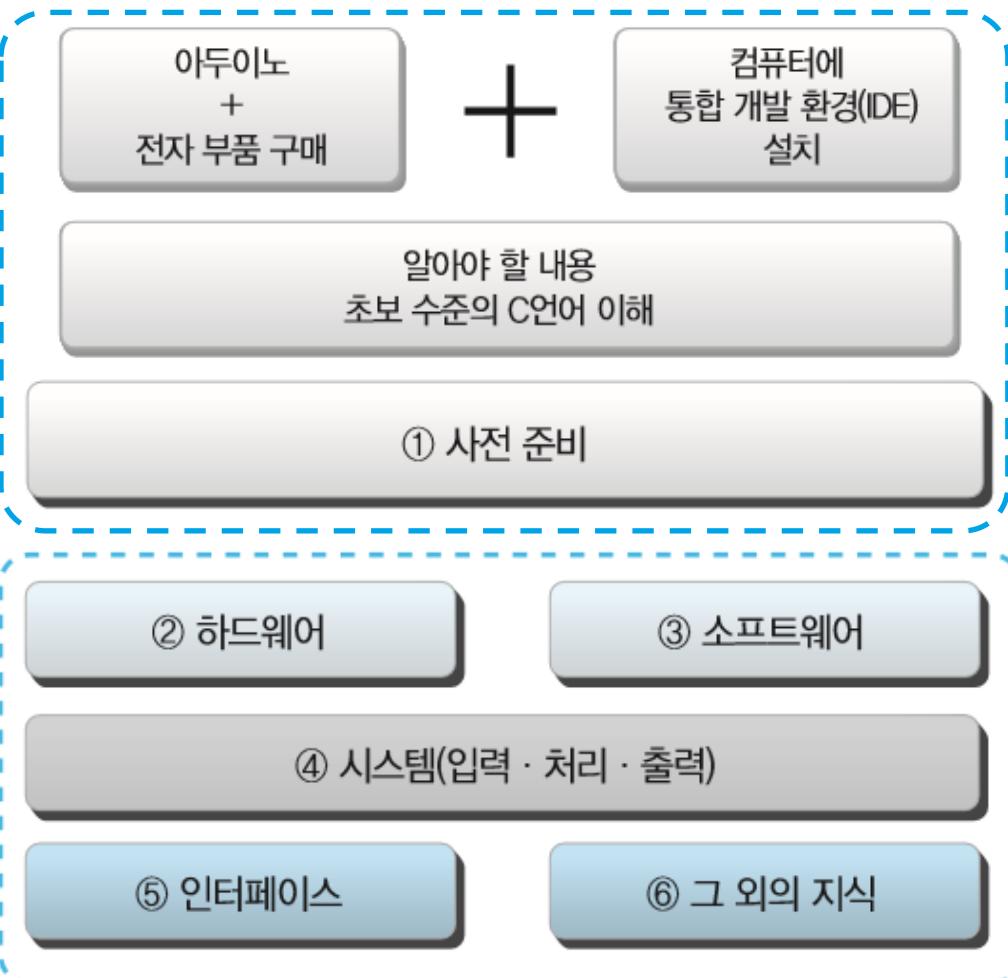
- 저작권 침해 위험이 적다
- 사용할 수 있는 기술이 공개되어 있어서 저작권 침해 걱정이 적음
- 자유롭게 가져다 쓸 수 있음(단, 대량 생산 판매일 때는 예외일 수 있음)

아두이노의 특징과 장점

- 시제품 개발에 효율적이고 빠르게 대응할 수 있다
- 아두이노는 시제품 개발에 사용하기 좋고 적은 수를 개발하거나 단기간만 사용할 제품을 개발하는 데도 빠르게 대응할 수 있음
- 이 외에도 브레드보드와 젠퍼 와이어는 여러 번 재활용할 수 있고, 쉽게 조립하고 응용할 수 있어서 교재로 사용하기 좋다는 점도 특징 중 하나

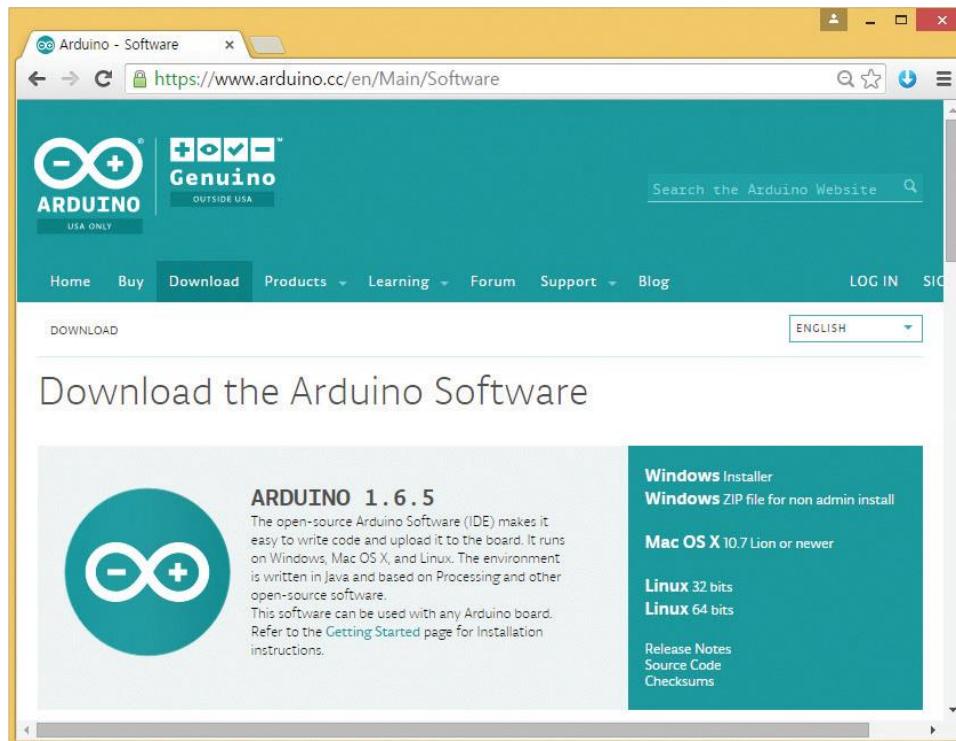


아두이노 학습 준비

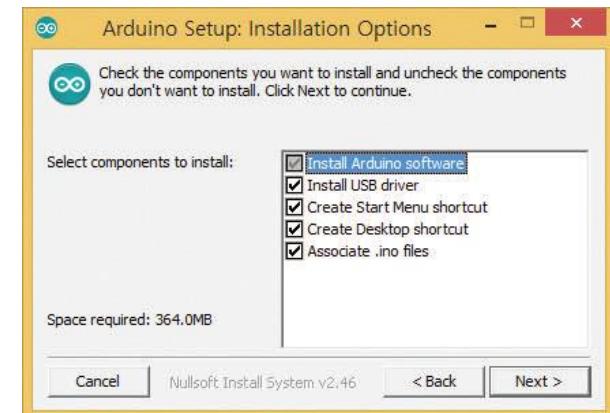


아두이노 IDE 환경 설치 및 설정

아두이노 IDE 설치 - 1



(1) IDE 다운로드 : [www.arduino.cc](https://www.arduino.cc/en/Main/Software)

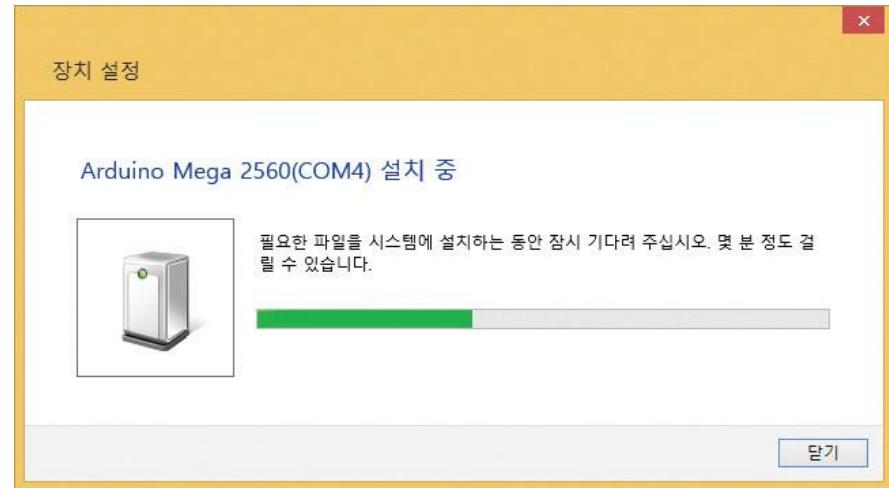


(2) 인스톨러 실행

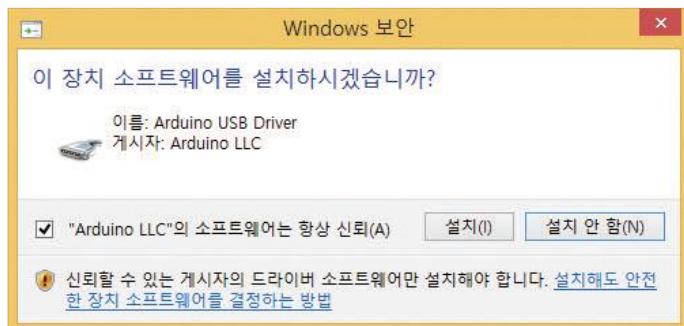
아두이노 IDE 설치 - 2



(3) 설치 디렉터리 지정



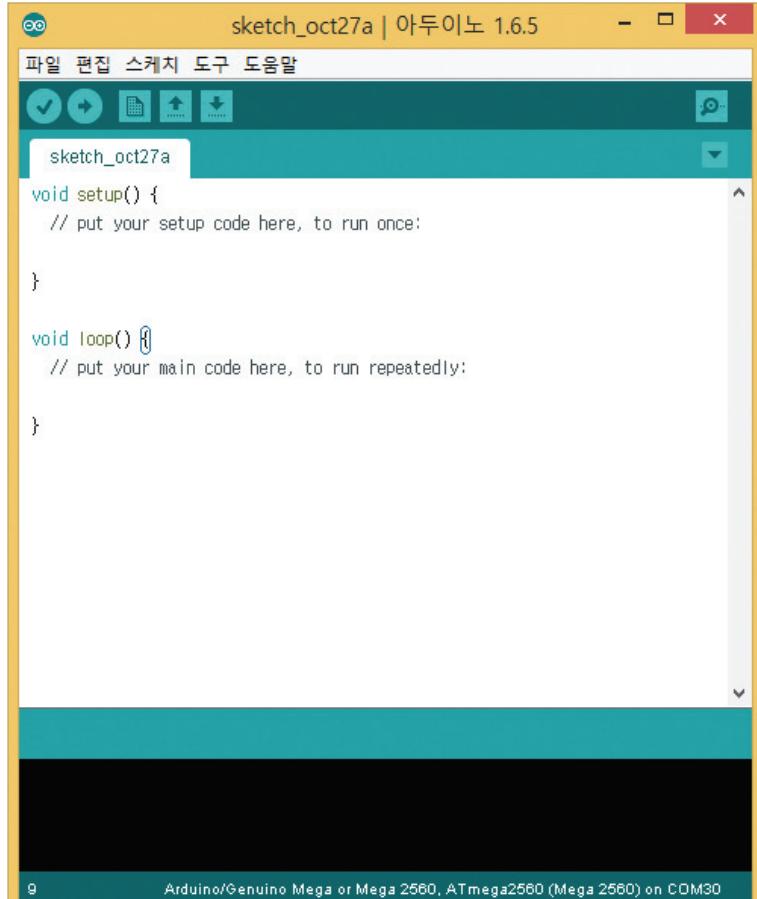
(5) 아두이노 보드 연결 : 가상 COM 포트 자동 설정



(4) 아두이노 보드 드라이버 자동 설치

아두이노 IDE

- 작고 간단한 통합개발환경
- 초보자를 위해 꼭 필요한 기능들만으로 구성
- 한번의 클릭으로 컴파일에서 업로드까지 진행
- C로 구현되어 OS 간 이식성이 뛰어남
- 디버깅 기능은 제공하지 않음



The screenshot shows the Arduino IDE interface with a sketch titled "sketch_oct27a". The code editor contains the following C-like pseudocode:

```
sketch_oct27a
void setup() {
    // put your setup code here, to run once:
}

void loop() {
    // put your main code here, to run repeatedly:
}
```

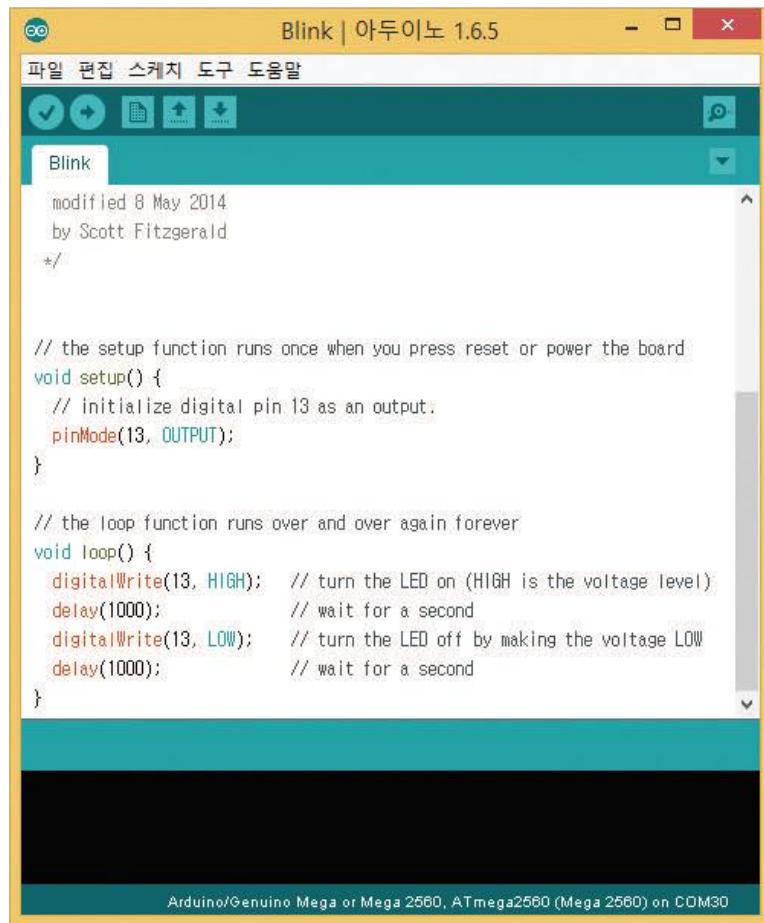
The status bar at the bottom indicates the board is an "Arduino/Genuino Mega or Mega 2560, ATmega2560 (Mega 2560) on COM30".

첫 번째 아두이노 프로그램

- 아두이노에서 프로그램은 그림을 그리듯 쉽게 작성할 수 있다는 의미에서 '스케치(sketch)'라 불림

- 블링크 (Blink) 스케치

- 13번 핀의 LED를 1초 간격으로 점멸하는 스케치
- C 언어에서의 'Hello World'에 해당
- '파일 → 예제 → 01.Basics → Blink' 선택



The screenshot shows the Arduino IDE interface with the title bar "Blink | 아두이노 1.6.5". The menu bar includes "파일", "편집", "스케치", "도구", and "도움말". The toolbar has icons for file operations like open, save, and upload. The main code editor window displays the "Blink" sketch:

```
modified 8 May 2014
by Scott Fitzgerald
/*
 * the setup function runs once when you press reset or power the board
void setup() {
    // initialize digital pin 13 as an output.
    pinMode(13, OUTPUT);
}

/* the loop function runs over and over again forever
void loop() {
    digitalWrite(13, HIGH);      // turn the LED on (HIGH is the voltage level)
    delay(1000);                // wait for a second
    digitalWrite(13, LOW);       // turn the LED off by making the voltage LOW
    delay(1000);                // wait for a second
}
```

At the bottom of the IDE, it says "Arduino/Genuino Mega or Mega 2560, ATmega2560 (Mega 2560) on COM30".

첫 번째 프로그램 자세히 살펴보기

- 줄 1~23 : 주석
 - '/'로 시작하고 '*/'로 끝나는 여러 줄로 이루어진 주석
- 줄 25 : 한 줄 주석
 - 같은 줄에서 '//' 뒤의 내용은 주석으로 처리
- 줄 26~29 : 초기화 setup 함수
 - 프로그램 실행이 시작될 때 한 번만 실행
 - 핀의 입출력 설정, 통신 인터페이스 초기화 등의 작업 수행
- 줄 28 : 핀의 입출력 상태 설정 pinMode 함수
 - 아두이노의 디지털 핀은 입력 또는 출력으로 사용 가능
 - 입력과 출력으로 동시에 사용은 불가능하므로 pinMode 함수로 입력 또는 출력으로 사용하도록 지정
 - 디폴트 상태는 입력 상태

첫 번째 프로그램 자세히 살펴보기

- 줄 28 : pinMode 함수

- /*로 시작하고 */로 끝나는 여러 줄로 이루어진 주석
- 첫 번째는 핀 번호를 지정
 - 아두이노 우노에서 내장 LED는 13번 핀에 연결되어 있음
 - 아두이노 보드에 따라 내장 LED가 연결된 번호는 다를 수 있음
 - 13 대신 내장 LED가 연결된 핀 번호를 나타내는 LED_BUILTIN 사용 가능
- 두 번째는 핀의 상태로 입력(INPUT) 또는 출력(OUTPUT)을 지정

- 줄 32~37 : 반복 수행 loop 함수

- 초기화를 위한 setup 함수 수행 이후 반복해서 loop 함수가 호출됨
- setup과 loop 함수는 아두이노의 기본 함수로 수행할 작업이 없어도 반드시 포함해야 함

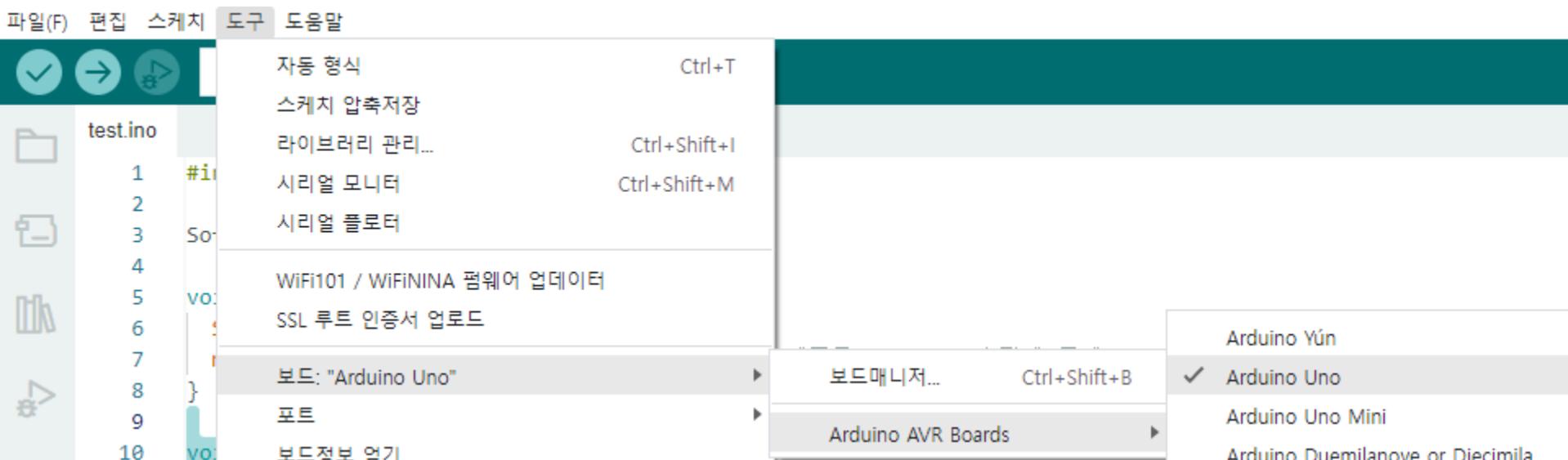
첫 번째 프로그램 자세히 살펴보기

- 줄 33 : 디지털 데이터 출력 digitalWrite 함수
 - 2개의 매개변수를 가짐
 - 첫 번째는 핀 번호를 지정
 - 두 번째 매개변수는 핀으로 출력될 값 지정
 - HIGH(5V) 또는 LOW(0V) 지정
- 줄 34 : 시간 지연 delay 함수
 - 밀리초 단위의 지연 시간을 매개변수로 가짐
 - 블링크 예제에서는 1초, 즉, 1000밀리초 간격으로 점멸 상태가 바뀜

프로그램 업로드를 위한 설정 1

■ 아두이노 보드 선택

- 사용하고자 하는 보드를 선택
- ‘도구 → 보드’ 메뉴에서 ‘Arduino/Arduino Uno’ 선택



프로그램 업로드를 위한 설정 3

■ 포트 선택

- 아두이노 보드에 할당된 가상 COM 포트 선택
- ‘도구 → 포트’ 메뉴에서 해당 포트 선택
- 아두이노 보드에 할당된 포트는
장치관리자에서 확인

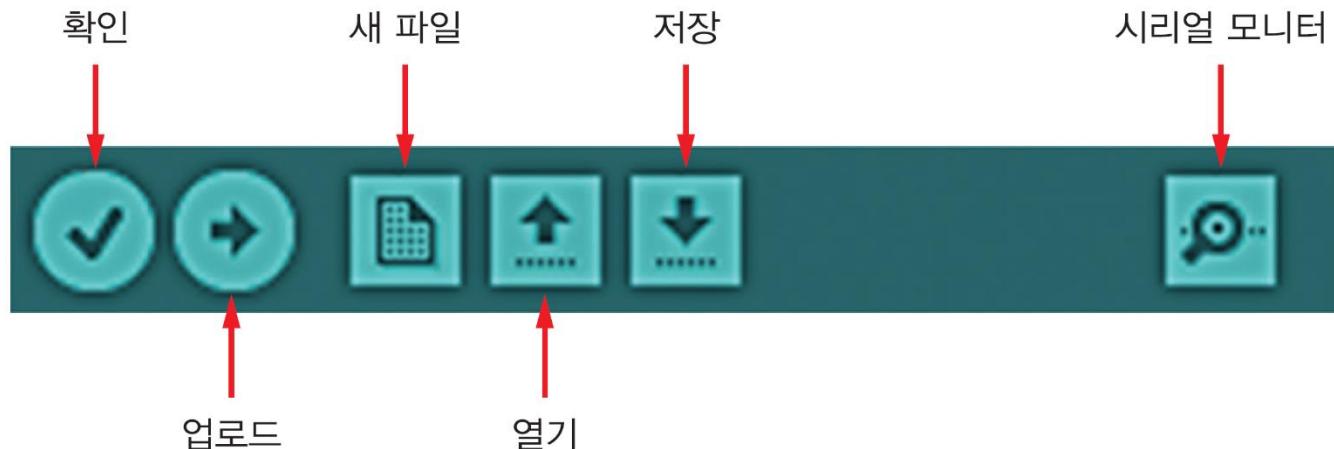


* 이 과목에서는 “Arduino Uno”선택

프로그램 업로드

■ 툴바 사용

- 확인 : 스케치 컴파일
- 업로드 : 스케치 컴파일 및 생성된 기계어 파일을 아두이노로 업로드
→ 한 번의 클릭으로 업로드까지 진행
- 시리얼 모니터 : 컴퓨터와의 데이터 송수신을 위한 프로그램 실행



스케치의 구조

- 스케치는 C/C++을 기반으로 함
- main 함수는 존재하지 않음
 - main 함수는 숨겨져 있으므로 신경 쓰지 않아도 됨
- 2개의 기본 함수로 구성
 - setup 함수
 - 초기화 함수
 - 스케치 실행이 시작될 때 한 번만 실행
 - loop 함수

- 반복 실행 함수
- 아두이노를 위한 프로그램에서 메인/이벤트 루프에 해당

스케치의 구조 비교

전처리

```
int main(void)
{
    초기화
    while(1)
    {
        데이터 처리
    }
    return 1;
}
```

기존 C 프로그램의 구조

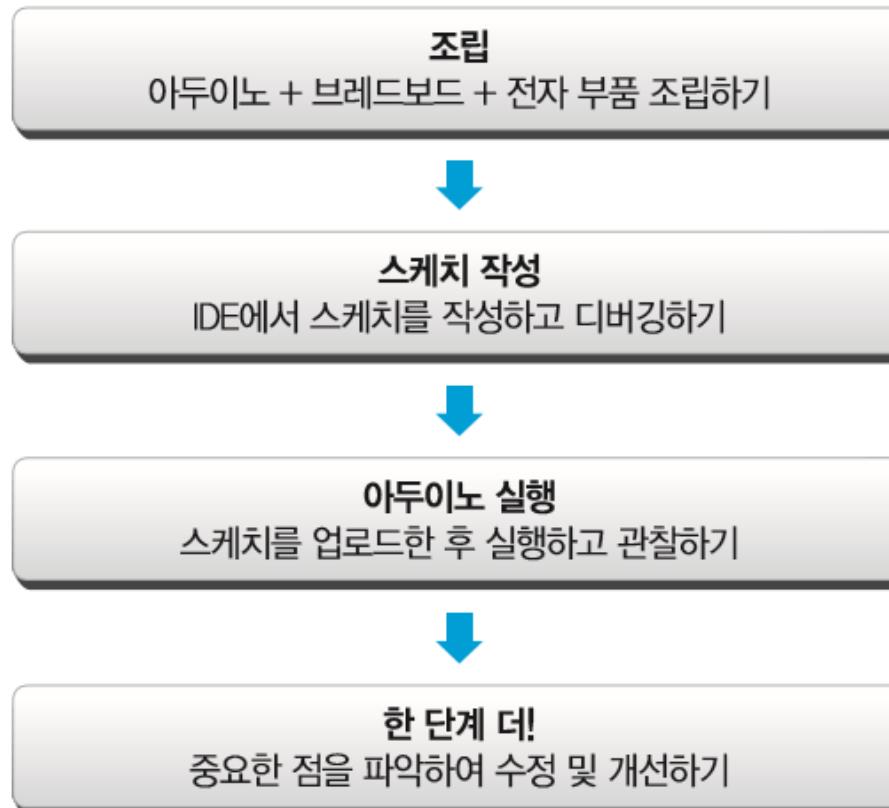
변수선언

전처리

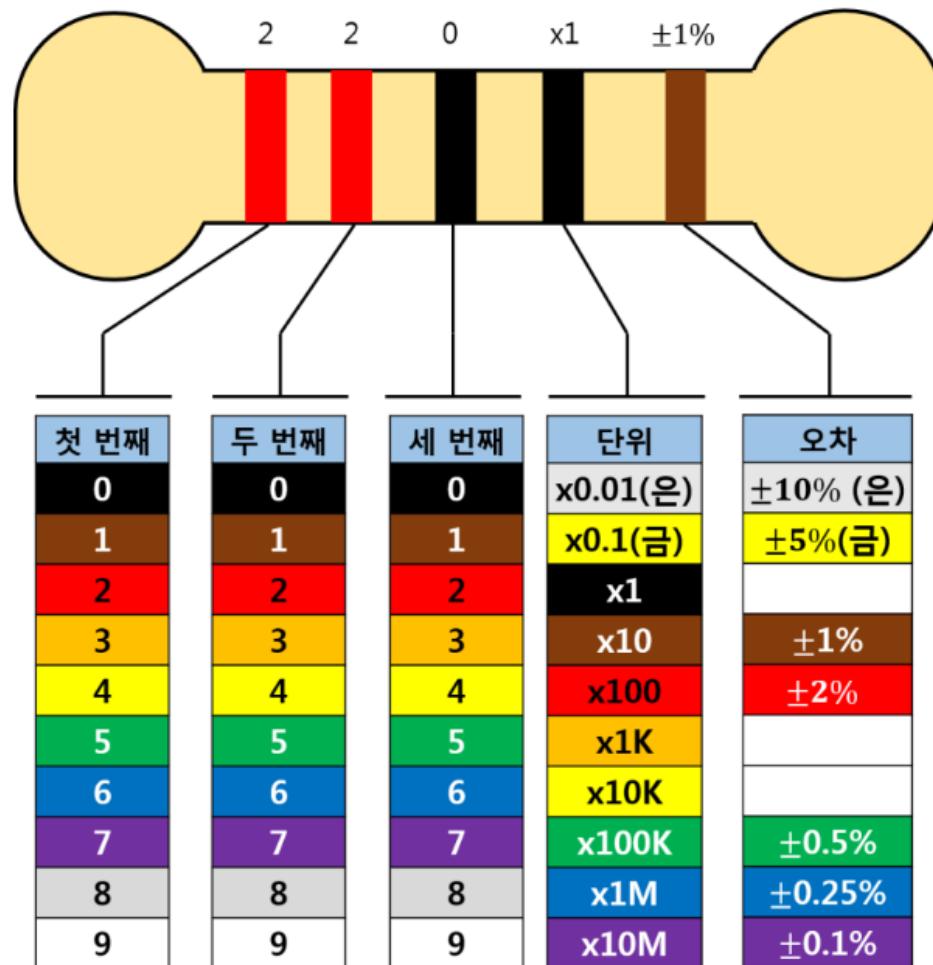
```
void setup(void)
{
    초기화
}
void loop(void)
{
    데이터 처리
}
```

아두이노를 위한 스케치의 구조

아두이노 실행 단계



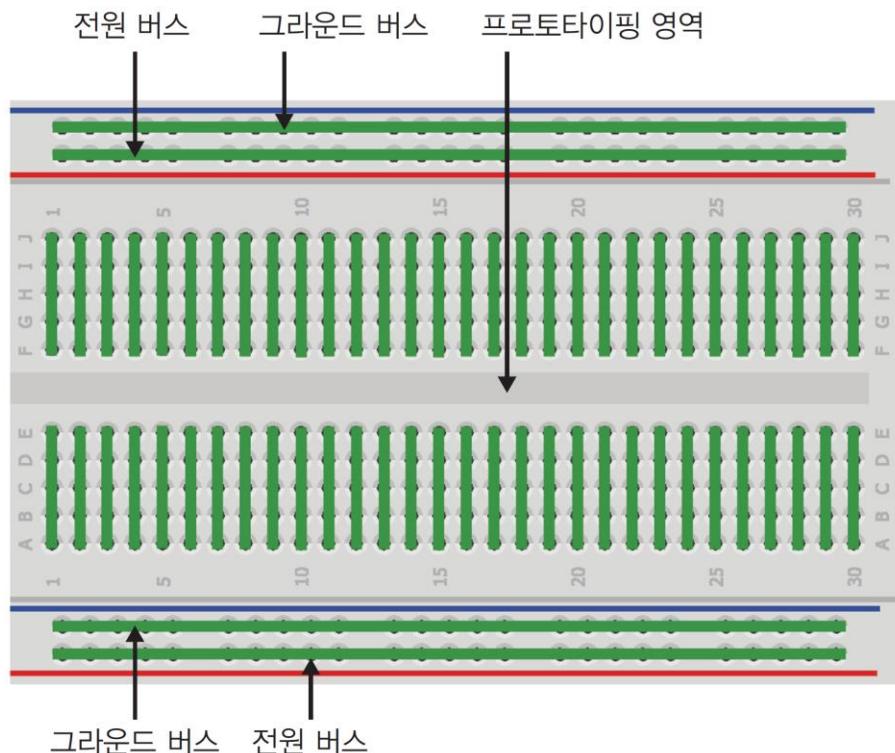
4. 저항 읽는 법



LED 모듈 연결

필요한 도구 : 브레드 보드

- 부품을 납땜하지 않고도 회로 구성을 가능하게 하는 프로토타이핑 도구
- 양쪽에 파란색(그라운드)과 빨간색(전원 버스) 훌은 연결된 상태
- 수직 훌은 5개 그룹으로 연결된 상태
- 각 핀이 내부적으로 연결되어 있어 각 전기/전자 부품을 연결하기 편리함



필요한 도구 : LED(Light Emitting Diode, 발광 다이오드)

- 양극(+, 애노드)에서 음극(-, 캐소드)방향으로만 전류가 흐름

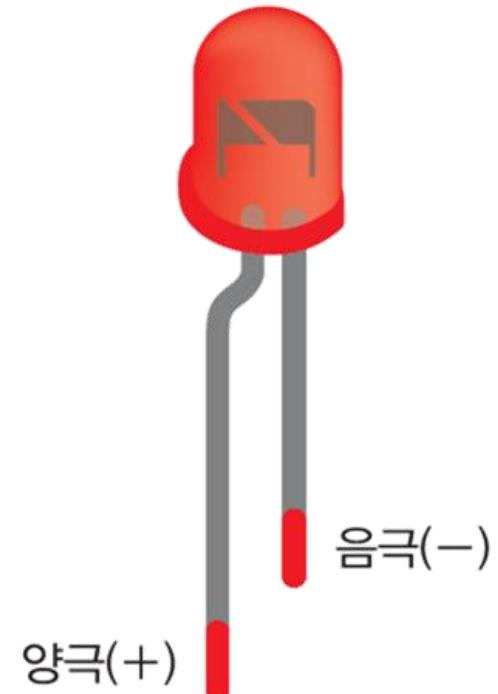
- 따라서 +, - 방향을 고려하여 연결해야 한다.

- 양극과 음극의 2개 다리를 가짐

- 다리 길이가 긴 쪽이 양극

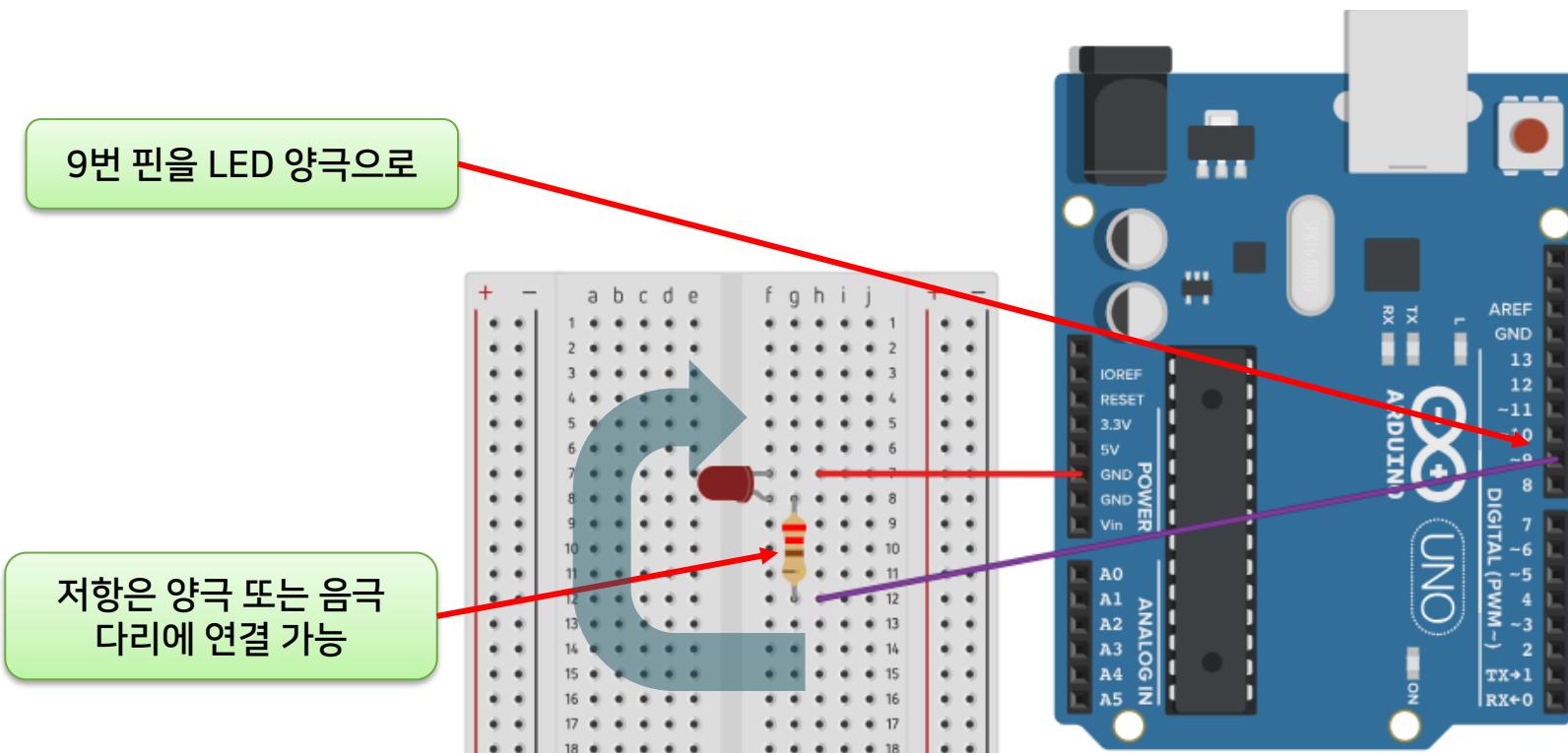
- 과도한 전류가 흐르지 않도록 전류 제한 필요

- 200Ω 전후 저항이 일반적으로 사용됨



아두이노에 LED 연결하기

- 아래 그림과 같이 LED를 아두이노에 연결한다.
 - 일반적인 LED는 2V의 전압과 2mA의 전류를 소모하기 때문에 5V의 아두이노 전압을 낮추기 위해 저항을 사용해야함
 - 220Ω 저항을 LED에 직렬로 연결
- 저항을 사용하지 않고 LED를 직접 연결하면 LED가 타버리거나 손상될 수 있음



LED 예제 함수

void pinMode(uint8_t pin, uint8_t mode)

- 매개변수
 - pin : 설정하고자 하는 핀 번호
 - mode : INPUT, OUTPUT, INPUT_PULLUP 중 하나
- 반환값 : 없음

void digitalWrite(uint8_t pin, uint8_t value)

- 매개변수
 - pin : 핀 번호
 - value : HIGH(1) 또는 LOW(0) => 디지털 출력: 1 또는 0으로만 표현
- 반환값 : 없음

void delay(unsigned long ms)

- 매개변수
 - ms : 밀리초 단위의 지연 시간
- 반환값 : 없음

uno_ex3-1: LED 모듈 예제

- 아두이노의 9번 핀에 연결된 LED를 켜기 위해, 9번 핀을 출력 모드로 설정

```
1 const int LED=9; //9번 핀을 사용하는 LED 상수 정의
2
3 ● void setup()
4 {
5     pinMode(LED, OUTPUT); //LED(9번 핀)를 출력으로 지정
6     digitalWrite(LED, HIGH); //LED를 HIGH로 지정      → 9번 핀을 5V로 설정 → LED 켜짐
7 }
8
9 void loop()
10{
11    //이 실습에서는 loop문을 비움
12}
```

디지털 9번 핀 (5V) → 저항 (220Ω) → LED (+극 → -극) → GND → 아두이노 내부 회로 → 다시 전원 공급 회로로 반환

Thank You