

IoT 장치를 위한 Azure Certification 프로그램

(주)한컴MDS
GB사업총괄 SE팀
essupport@hancommds.com



목차

1. IoT 장치와 Starter Kit 을 인증해야 하는 이유	2
2. IoT 장치와 Starter Kit 을 인증하는 방법	2
3. Azure Certified for IoT 프로그램 혜택	3
4. Azure Certified for IoT 프로그램 진행 절차	4
5. Azure IoT SDK 를 이용한 디바이스와 Azure IoT Hub 간 연결 테스트	5
Step 1. 개발 시스템 환경 구성	5
Step 2. Azure IoT Hub 구성	10
Step 3. IoT 장치 등록	13
Step 4. 장치에 Azure IoT SDK 빌드 및 배포	16
Step 5. Azure IoT Client SDK 샘플 실행 및 유효성 검사	18
Step 6. 로그 패키징 및 공유	23
Step 7. Microsoft Azure Certification 진행 완료 메일 수신	26

1. IoT 장치와 Starter Kit 을 인증해야 하는 이유



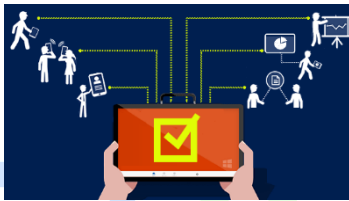
신규 고객 발굴

Azure Certified for IoT Device Catalog 사이트를 비즈니스 온라인 채널로 활용하십시오. 고객이 직접 귀하의 영업 부서에 연락하여 더 많은 비즈니스 기회를 창출 할 수 있습니다.



Azure 와의 연결

고객이 장치 및 키트 별 Azure 자습서를 통해 빠른 시간 안에 Azure 에 연결할 수 있습니다. Microsoft 템플릿을 활용하여 고객이 단기간에 귀하의 기술로 시작할 수 있도록 하십시오.



구매자와의 신뢰

인증된 장치는 IoT 고객이 프로젝트 진행을 안심하게 할 수 있도록 도와줍니다.



IoT 네트워크 성장

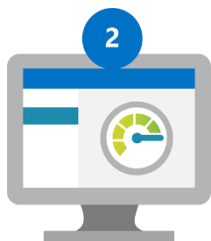
다른 Microsoft IoT 파트너와 공동 작업 및 협력하여 더 많은 성과를 거두십시오.

2. IoT 장치와 Starter Kit 을 인증하는 방법



프로필 만들기

Catalog.azureiotsuite.com 에서 기본 회사 정보를 작성하면 파트너 대시보드를 통해 장치를 추가하거나 관리할 수 있습니다.



장치 추가

인증 받을 장치의 OS 플랫폼 및 프로그래밍 언어를 선택하십시오. Microsoft는 단계별 지침을 제공합니다.



테스트 및 검증

호환성 테스트를 실행하고 다음을 제공하십시오.

- 장치에 에이전트 설치하기 위한 패키징
- Azure IoT 기기로 사용하는 예



등록

혜택을 누리십시오! 장치가 자동으로 카탈로그에 등록되고 게재됩니다.



새 키트 인증

각 키트 인증은 인증된 장치로 시작됩니다. 기기가 이미 인증된 경우 2~4단계에 따라 키트를 추가하고 테스트 및 검증한 다음 등록하십시오! 그렇지 않으면 1단계부터 시작하여 먼저 장치를 인증하십시오.

3. Azure Certified for IoT 프로그램 혜택

PR 기회와 로고 추천	기업과 파트너 간의 연결	검색 기능 및 Azure 인증 사용 권한
<ul style="list-style-type: none"> • 소셜 미디어에 로고 노출 • Microsoft 경영진 인용 지원 • 모든 IoT 인증 웹 사이트에 접근 가능한 로고 	<ul style="list-style-type: none"> • 기기 목록을 통해 고객과 직접 연락 및 연결할 수 있는 기능 • 새로운 비즈니스 리드 발굴 • 등록된 하드웨어 목록에서의 비즈니스 통찰력 제공 	<ul style="list-style-type: none"> • 고객에게 신뢰를 보여줄 수 있는 Azure 공인 배지 • 향상된 기기 및 키트의 검색 기능 • 특정 비즈니스 및 기술 요구 사항 별로 하드웨어 나열

4. Azure Certified for IoT 프로그램 진행 절차

- ① IoT용 Azure Certified 장치 카탈로그 사이트 (<https://catalog.azureiotsuite.com>) 접속 후, 왼쪽 상단의 "Become a Partner" 버튼 클릭
- ② 해당 페이지에서 왼쪽 상단의 "Get Started" 버튼 클릭
- ③ "파트너 등록 양식 페이지" 에서 파트너 프로필 작성 후 "제출" 버튼 클릭
 - ※ 회사 전자 메일은 반드시 Microsoft 계정 또는 조직 계정 (Active Directory 에 링크) 이어야 함. Microsoft 계정은 <https://www.microsoft.com/en-us/account> 사이트에서 생성 가능
 - ※ 회사 로고는 정사각형 비율로, 지원되는 형식은 .png, .jpg, .jpeg 이며 최대 파일 크기는 300KB 임
- ④ 가입 후 IoT용 인증 지원 팀으로부터 계정 활성화를 위한 메일을 수신 받으면, 메일 내용에 따라 절차 진행하여 파트너 가입 완료
 - ※ 5분 안에 메일이 수신되지 않을 경우, 정크 (스팸) 메일 확인
- ⑤ 계정 활성화가 되면, 오른쪽 상단의 "파트너 대시보드" 클릭하여 로그인 진행
- ⑥ "파트너 대시보드" 페이지에서 Device(s) 탭의 "+" 눌러 Azure Certification 받을 디바이스 등록 진행
- ⑦ 장치 등록 양식 작성 후 "저장" 버튼 클릭하여 디바이스 정보 저장
 - ※ 최종 제출하기 전까지는 언제든지 장치 등록 정보 수정 및 저장 가능
- ⑧ 디바이스 등록하면 IoT용 인증 지원 팀에서 해당 요청을 검토하고 사용자에게 이 메일로 해당 지침을 전송
- ⑨ 해당 지침에 따라 Azure IoT SDK 를 이용하여 해당 디바이스와 Azure IoT Hub 와의 연결이 유효한지 통신 테스트 진행
 - ※ Azure IoT SDK 이용한 테스트 진행 방법은 아래 "섹션 5" 에서 자세하게 설명
- ⑩ 디바이스와 Azure IoT Hub 와의 통신 테스트 로그 수집 후, 해당 로그를 IoT용 인증 지원 팀에게 iotcert@microsoft.com 이메일 발송
- ⑪ IoT용 인증 지원 팀에서 48시간 이내에 제출한 로그 확인 후 최종 인증 완료 메일 전송
- ⑫ 모든 인증 단계가 완료되면 IoT용 Azure Certified 장치 카탈로그 사이트에 자동으로 디바이스 게재되고 확인 메일 수신

5. Azure IoT SDK 를 이용한 디바이스와 Azure IoT Hub 간 연결 테스트

Windows 기반 디바이스와 Azure IoT Hub 와의 연결 유효성 테스트 절차

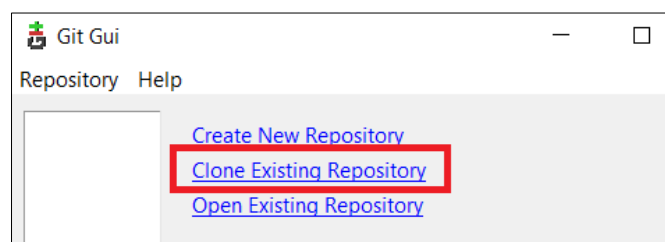
- Step 1. 개발 시스템 환경 구성
- Step 2. Azure IoT Hub 구성
- Step 3. IoT 장치 등록
- Step 4. 장치에 Azure IoT SDK 빌드 및 배포
- Step 5. Azure IoT Client SDK 샘플 실행 및 유효성 검사
- Step 6. 로그 패키징 및 공유
- Step 7. Microsoft Azure Certification 진행 완료 메일 수신

Step 1. 개발 시스템 환경 구성

1. Visual Studio 2017 (VS 2015 이상 설치, Community Edition도 무관)
- Visual Studio Community 무료 다운로드:
<https://www.visualstudio.com/ko-kr/downloads/download-visual-studio-vs.aspx>
2. Git GUI Clients 다운로드 및 설치 <https://git-scm.com/downloads>
 - ① 페이지에서 Downloads for Windows 를 클릭하여 Windows용 Git 설치
 - ② Windows 용 Git 설치 시 기본 설정 값으로 설치 진행
3. 로컬 시스템에 Azure IoT SDK 복사 (GUI 방식과 Bash 명령어 방식이 있음)

<GUI 방식>

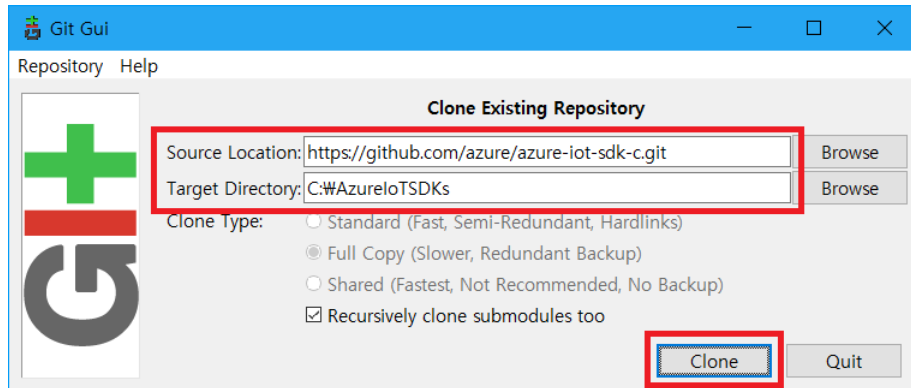
- ① Git GUI 실행하여 'Clone Existing Repository' 선택



- ② Source Location 에 <https://github.com/Azure/azure-iot-sdk-c.git> 입력
- ③ Target Directory 에 Azure IoT SDK 를 복사할 폴더 지정

(예: C:\AzureIoTSDKs)

- ④ 'Clone' 을 눌러 웹으로부터 로컬시스템에 Azure IoT SDK 다운로드 진행



<Bash 명령어 방식>

- ① GitHub 로 부터 Azure IoT SDK 를 복사할 폴더 생성 (예: C:\AzureIoTSDKs)

- ② Git 설치 후, Git Bash 실행한 다음, C:\AzureIoTSDKs 폴더로 이동

- ③ Azure IoT SDK 복사 명령어 실행

```
git clone --recursive https://github.com/azure/azure-iot-sdk-c.git
```

4. CMake 다운로드 및 설치

- ① CMake 다운로드: <https://cmake.org/download/>

※ Windows Installer 용 파일 다운로드 시 개발 시스템의 프로세서에 맞게 다운로드!

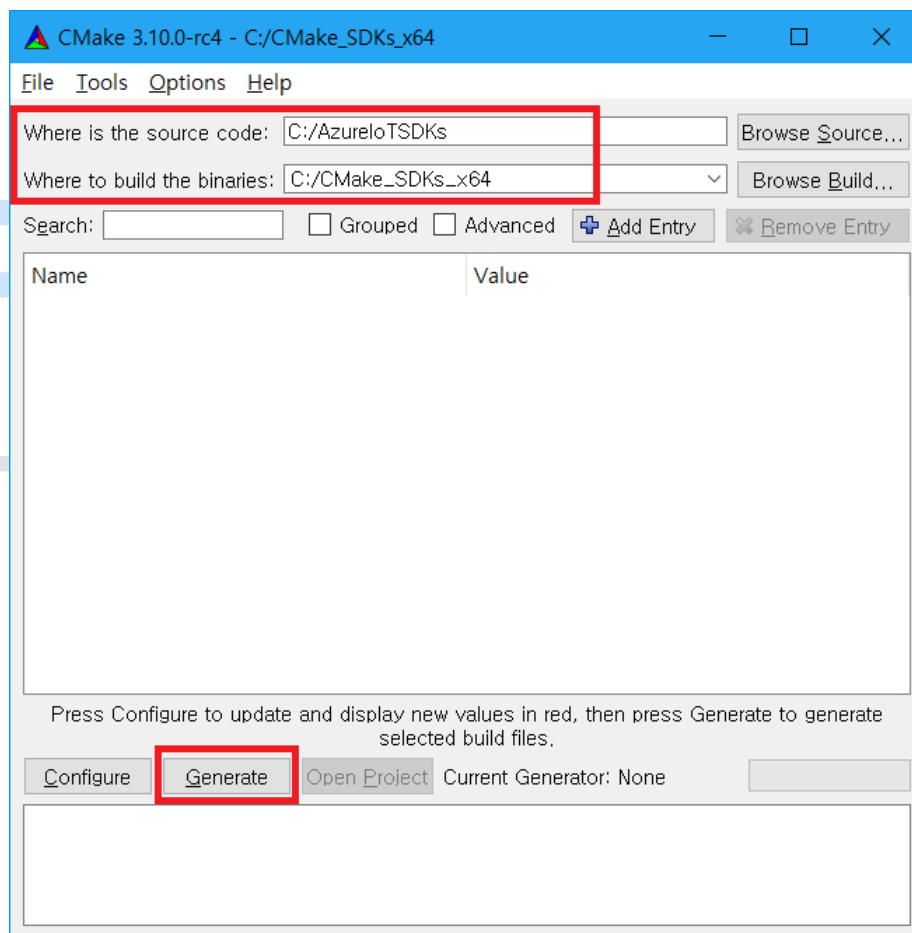
- ② 본문에서는 64bit 용 cmake-3.10.0-rc5-win64-x64.msi 파일 다운로드 및 설치

Platform	Files
Windows win64-x64 Installer: Installer tool has changed. Uninstall CMake 3.4 or lower first!	cmake-3.10.0-rc5-win64-x64.msi
Windows win64-x64 ZIP	cmake-3.10.0-rc5-win64-x64.zip
Windows win32-x86 Installer: Installer tool has changed. Uninstall CMake 3.4 or lower first!	cmake-3.10.0-rc5-win32-x86.msi
Windows win32-x86 ZIP	cmake-3.10.0-rc5-win32-x86.zip
Mac OSX 10.6 or later	cmake-3.10.0-rc5-Darwin-x86_64.dmg
	cmake-3.10.0-rc5-Darwin-x86_64.tar.gz
Linux x86_64	cmake-3.10.0-rc5-Linux-x86_64.sh
	cmake-3.10.0-rc5-Linux-x86_64.tar.gz

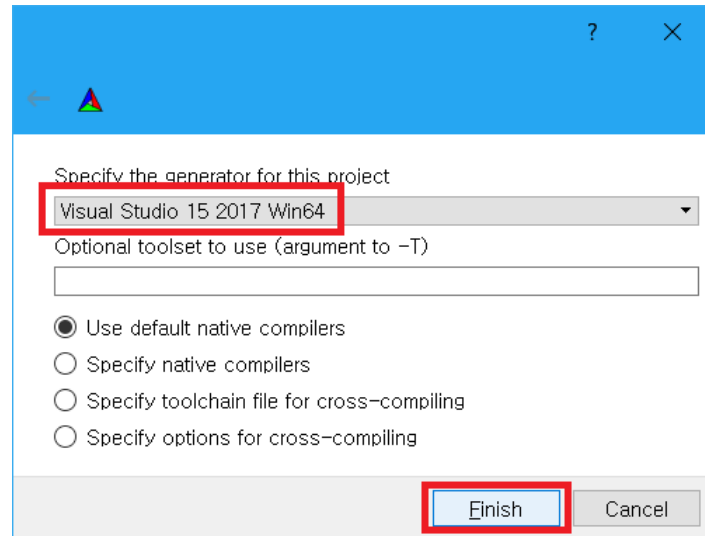
5. CMake 를 이용하여 IoT SDK 프로젝트 생성 (GUI 방식과 CMD 명령어 방식이 있음)

<GUI 방식>

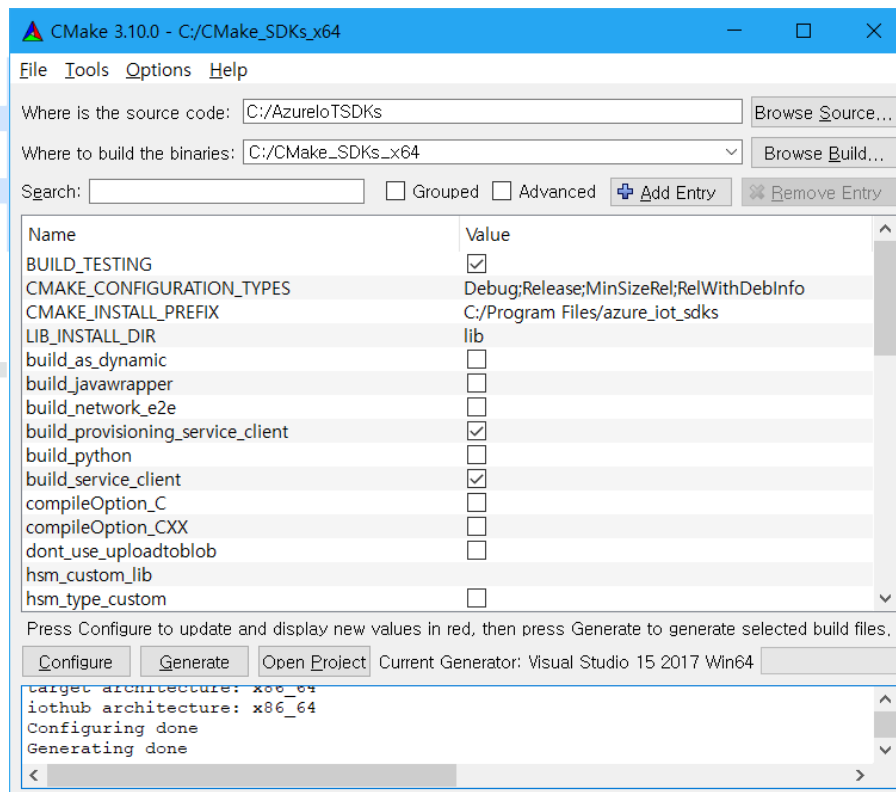
- ① CMake GUI 버전 (cmake-gui) 실행
 - ② 'Where is the source code' 항목에는 이전 단계에서 Git 으로부터 복사한 Azure IoT SDK 폴더 안의 C 폴더 값 선택 (예: C:\AzureIoTSdks)
 - ③ 'Where to build the binaries' 항목에는 CMake 프로젝트가 생성될 경로 지정 후, 'Generate' 클릭 (예: C:\CMake_SDKs_x64)
- ✓ 프로젝트 생성 버전은 Azure Certification 을 받을 시스템 운영 환경 (x86 or x64) 에 맞게 생성



- ④ 시스템에 설치된 Visual Studio 환경에 맞는 항목 선택 후 'Finish' 를 클릭
- ✓ 해당 예시는 Visual Studio 2017 x64 버전을 선택하여 진행함



⑤ 지정한 'C:\CMake_SDKs_x64' 폴더에 x64용 C 프로젝트 파일 생성됨



⑥ 생성된 x64용프로젝트 확인 (예: C:\CMake_SDKs_x64)

이름	수정된 날짜	유형	크기
CMakeFiles	2017-11-22 오전 9...	파일 폴더	
c-utility	2017-11-22 오전 9...	파일 폴더	
deps	2017-11-22 오전 9...	파일 폴더	
iothub_client	2017-11-22 오전 9...	파일 폴더	
iothub_service_client	2017-11-22 오전 9...	파일 폴더	
serializer	2017-11-22 오전 9...	파일 폴더	
Testing	2017-11-22 오전 9...	파일 폴더	
uamqp	2017-11-22 오전 9...	파일 폴더	
umqtt	2017-11-22 오전 9...	파일 폴더	
ALL_BUILD.vcxproj	2017-11-22 오전 9...	VC++ Project	68KB
ALL_BUILD.vcxproj.filters	2017-11-22 오전 9...	VC++ Project Fil...	1KB
azure_iot_sdks.sln	2017-11-22 오전 9...	Microsoft Visual S...	91KB
cmake_install.cmake	2017-11-22 오전 9...	CMAKE 파일	2KB
CMakeCache.txt	2017-11-22 오전 9...	텍스트 문서	28KB
Continuous.vcxproj	2017-11-22 오전 9...	VC++ Project	58KB
Continuous.vcxproj.filters	2017-11-22 오전 9...	VC++ Project Fil...	1KB
CTestTestfile.cmake	2017-11-22 오전 9...	CMAKE 파일	1KB
DartConfiguration.tcl	2017-11-22 오전 9...	TCL 파일	3KB
Experimental.vcxproj	2017-11-22 오전 9...	VC++ Project	58KB
Experimental.vcxproj.filters	2017-11-22 오전 9...	VC++ Project Fil...	1KB
INSTALL.vcxproj	2017-11-22 오전 9...	VC++ Project	11KB
INSTALL.vcxproj.filters	2017-11-22 오전 9...	VC++ Project Fil...	1KB
Nightly.vcxproj	2017-11-22 오전 9...	VC++ Project	58KB
Nightly.vcxproj.filters	2017-11-22 오전 9...	VC++ Project Fil...	1KB
NightlyMemoryCheck.vcxproj	2017-11-22 오전 9...	VC++ Project	58KB
NightlyMemoryCheck.vcxproj.filters	2017-11-22 오전 9...	VC++ Project Fil...	1KB
RUN_TESTS.vcxproj	2017-11-22 오전 9...	VC++ Project	11KB

<CMD 명령어 방식>

① CMake 설치 후, CMake 환경변수 등록

✓ 설치한 Visual Studio 2017 관련 프로그램 중 'Developer Command Prompt for VS 2017' 실행 (시작 – Visual Studio 2017 경로에서 확인 가능)

✓ set path="C:\Program Files\CMake\bin";%PATH% 명령 실행하여 등록

② 다운로드 받은 Azure IoT SDK 프로젝트 폴더로 이동 (C:\AzureIoTSDKs)

③ 해당 폴더에서 아래 명령어를 실행하여 프로젝트 생성

✓ Azure IoT Certification 을 받을 장비가 x86인 경우,

```
mkdir CMake_SDKs_x86
cd CMake_SDKs_x86
cmake -G "Visual Studio 15 2017" ..
```

✓ Azure IoT Certification 을 받을 장비가 x64인 경우,

```
mkdir CMake_SDKs_x64
```

cd CMake_SDKs_x64

cmake -G "Visual Studio 15 2017 Win64" ..

※ 명령어 옵션 중 .. 은 상위 경로에 있는 CMakeLists.txt 파일저장위치를 의미

Step 2. Azure IoT Hub 구성

1. IoT Hub 생성

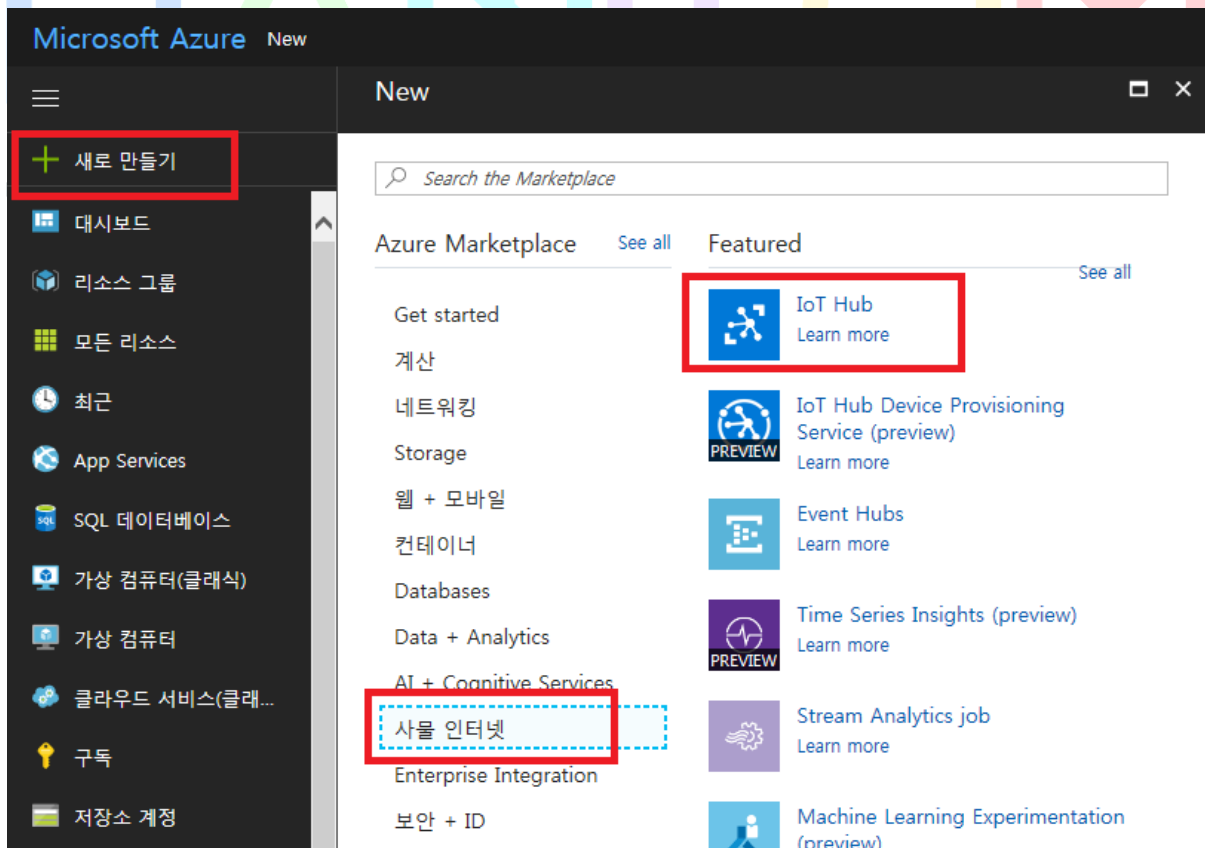
- ① Microsoft Azure 계정 생성 (<https://azure.microsoft.com/ko-kr/>)

※ 웹 페이지 오른쪽 상단에 "무료 계정" 버튼 클릭하여 계정 생성

- ② Azure 계정 생성 후, Azure 포탈 사이트 접속 (<https://portal.azure.com>)

※ Azure 포탈 접속 시 생성한 Azure 계정으로 로그인 필요

- ③ '+ 새로 만들기' - '사물 인터넷' - 'IoT Hub' 클릭



- ④ IoT Hub 이름, 가격 및 크기 계층, 리소스 그룹, 위치 등의 필드값 기입 후 '만들기' 실행

Microsoft Azure

New > IoT 허브

IoT 허브

Microsoft

+

 새로 만들기

대시보드

리소스 그룹

모든 리소스

최근

App Services

SQL 데이터베이스

가상 컴퓨터(클래식)

가상 컴퓨터

클라우드 서비스(클래...

구독

저장소 계정

가상 네트워크

보안 센터

가상 컴퓨터 확장 집합

스트림 분석 작업

Azure Active Directory

모니터

비용 관리 + 청구

추가 서비스 >

* 이름

AzureIoTCertiWin

* 가격 및 크기 계층

F1 - 무료

i

무료 계층을 선택하는 경우 나중에 유료 계층으로 이동하려면 전체 IoT Hub를 삭제하거나 다시 만들어야 합니다.

* IoT Hub 단위 ⓘ

1

* 장치-클라우드 파티션 ⓘ

2개 파티션

* 구독

Visual Studio Enterprise with MSDN

* 리소스 그룹 ⓘ

● 새로 만들기

○ 기존 그룹 사용

AzureIoTCerti

* 위치

일본 서부

☐ 대시보드에 고정

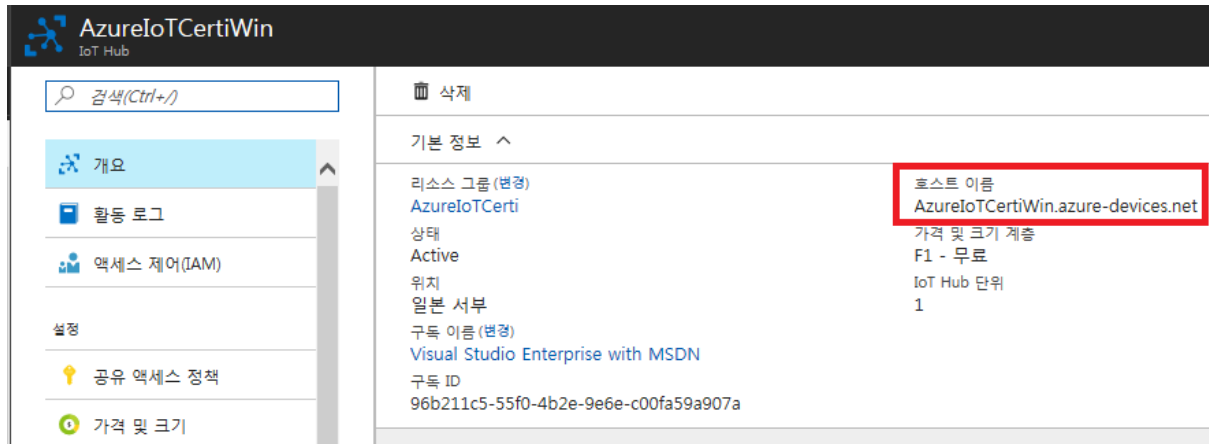
만들기

자동화 옵션

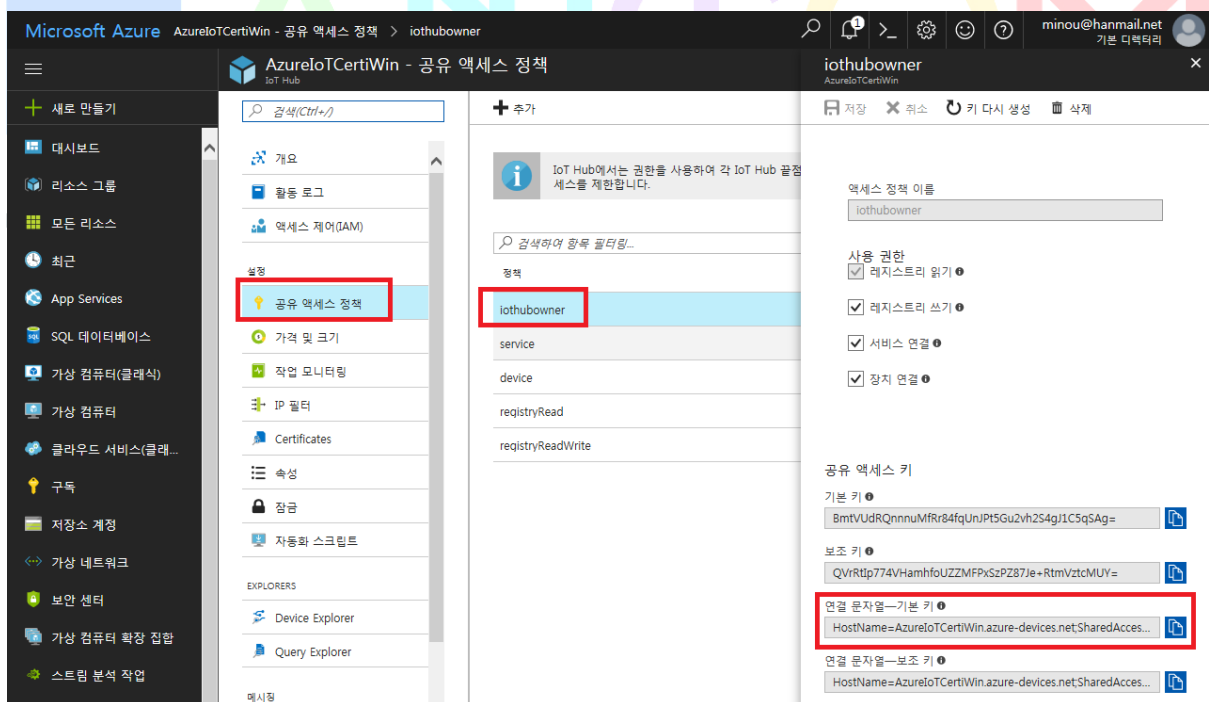
페이지 11 / 26

2. IoT Hub 의 Hostname 과 Connection String 복사

① 생성한 IoT Hub 의 기본 정보에서 '호스트 이름' 복사하여 메모장에 저장

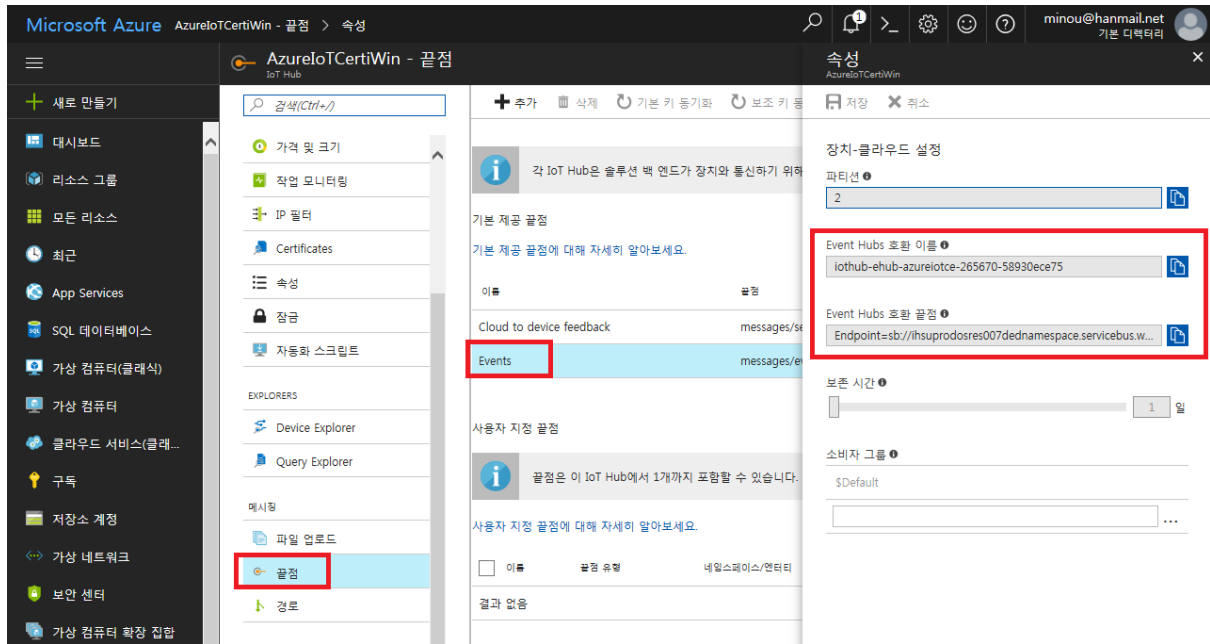


② 왼쪽 메뉴에서 '공유 액세스 정책' - 'iothubowner' - '연결 문자열 - 기본 키' 복사하여 메모장에 저장



③ 왼쪽 메뉴에서 '끝점' 클릭 후, 기본 제공 끝점 항목에서 'Events' 클릭, 'Event Hubs' 호환 이름, 'Event Hubs 호환 끝점' 두 항목 값 복사하여 메모장에 저장

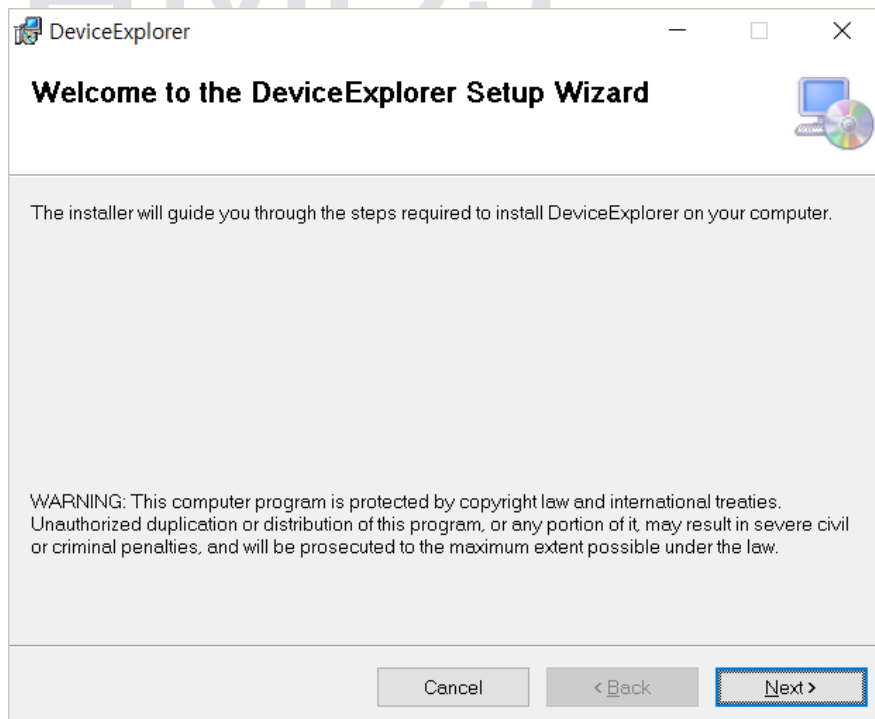
※ 'read-d2c-messages' 앱을 생성 시 위 세 가지 값 필요



Step 3. IoT 장치 등록

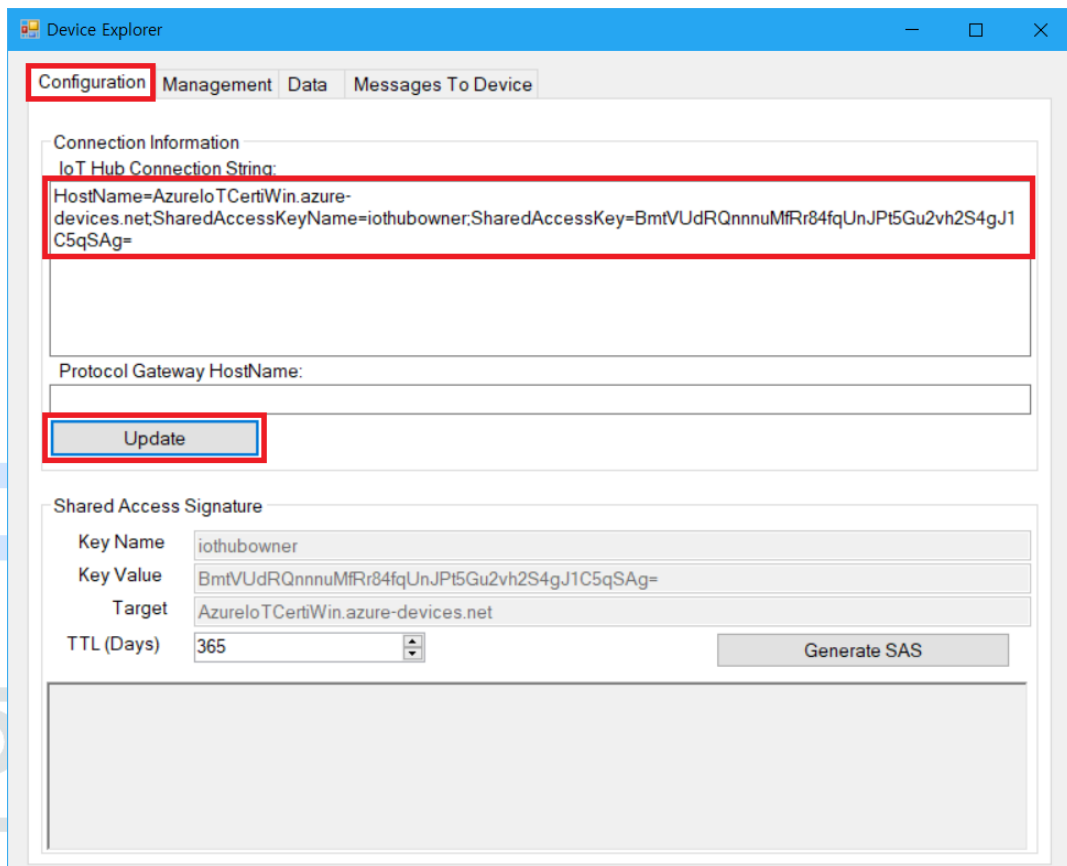
1. 개발 시스템에서 Device Explorer 다운로드 및 설치

- ① Device Explorer 다운로드: <https://github.com/Azure/azure-iot-sdks/releases>
- ② SetupDeviceExplorer.msi 파일 다운로드 및 설치



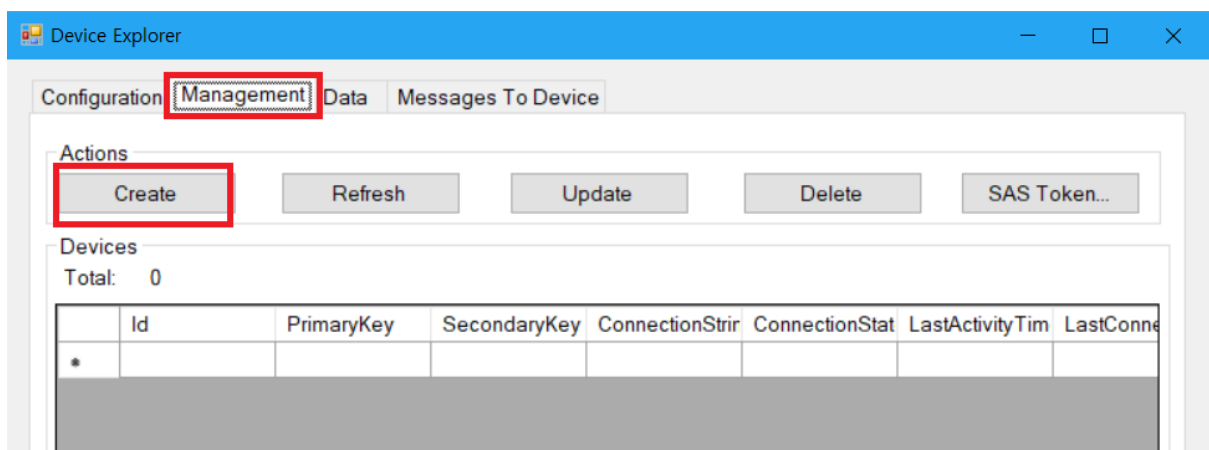
2. Connection 정보 추가

- ① Device Explorer 실행 – ‘Configuration’ 탭 선택 – ‘IoT Hub Connection String’ 항목에 이전 IoT Hub 생성 단계에서 복사한 IoT Hub Connection String 값 복사
- ② 값 복사 후 ‘Update’ 클릭하여 연결 정보 추가



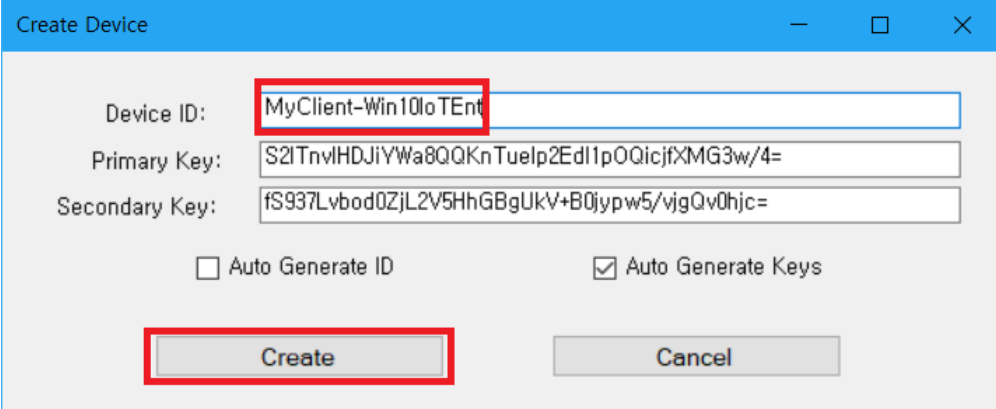
3. IoT Hub 에 연결할 장치 생성 및 등록

- ① Device Explorer 의 ‘Management’ 탭 선택 후 ‘Create’ 클릭

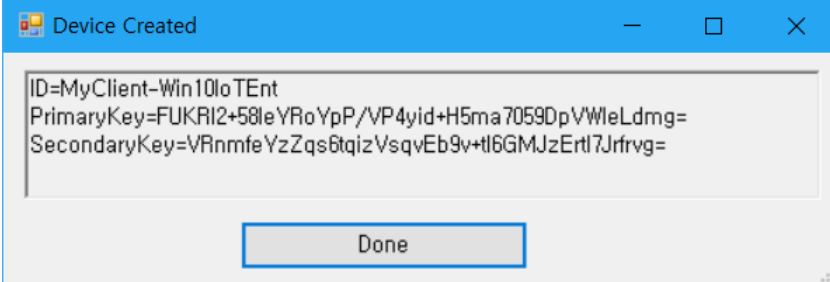


- ② Device ID 항목에 추가할 장치 ID (Azure IoT Certification 을 받을 장비) 입력 후 'Create' 클릭하여 장치 추가

※ Azure IoT Certification 을 받을 시스템의 컴퓨터 이름 입력

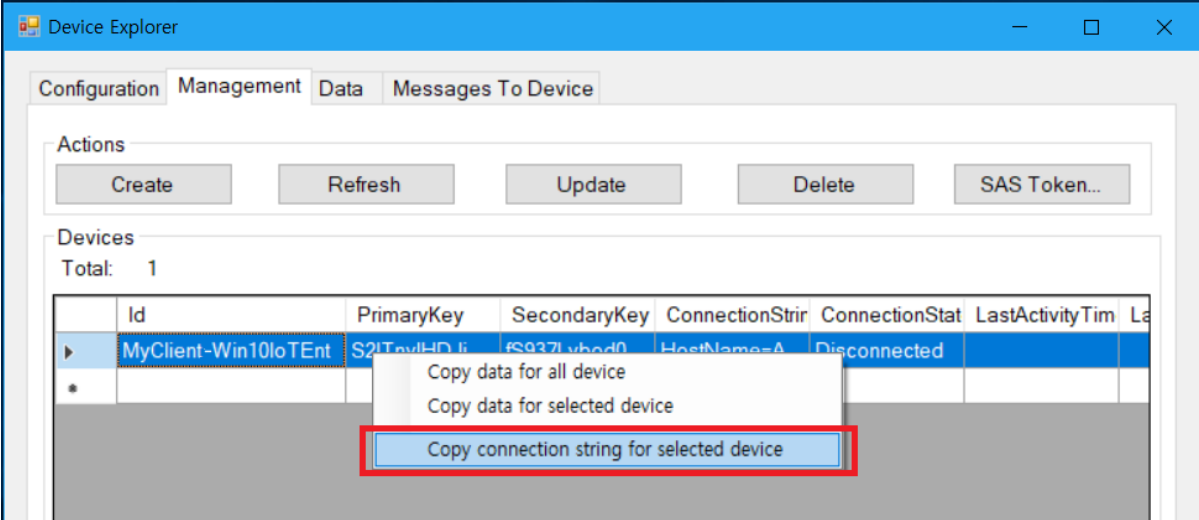


The 'Create Device' dialog box is shown. It has three input fields: 'Device ID' (containing 'MyClient-Win10IoTEnt'), 'Primary Key' (containing 'S2ITnvIHDJiYWa8QQKnTuelp2Edl1p0QicjfXMG3w/4='), and 'Secondary Key' (containing 'fS937Lvbod0ZjL2V5HhGBgUkV+B0jypw5/vjgQv0hjc='). Below these fields are two checkboxes: 'Auto Generate ID' (unchecked) and 'Auto Generate Keys' (checked). At the bottom are 'Create' and 'Cancel' buttons. The 'Device ID' field and the 'Create' button are highlighted with red rectangles.



The 'Device Created' dialog box is shown. It displays the generated information: ID=MyClient-Win10IoTEnt, PrimaryKey=FUKRI2+58leYRoYpP/VP4yid+H5ma7059DpVWleLdmg=, and SecondaryKey=VRnmfeYzZqs6tqizVsqvEb9v+tl6GMJzErtl7Jfrvrg=. A 'Done' button is at the bottom.

- ③ 추가한 장치에서 마우스 우클릭하여 'Copy connection string for selected device' 클릭하여 해당 장치의 Connection String 값을 메모장에 복사



The 'Device Explorer' window is shown. It has tabs for 'Configuration', 'Management', 'Data', and 'Messages To Device'. Under 'Management', there are buttons for 'Create', 'Refresh', 'Update', 'Delete', and 'SAS Token...'. Below these is a 'Devices' section showing 'Total: 1'. A table lists the device with columns: Id, PrimaryKey, SecondaryKey, ConnectionStr, ConnectionStat, LastActivityTim, and La. The device 'MyClient-Win10IoTEnt' is selected. A right-click context menu is open over the selected device, showing options: 'Copy data for all device', 'Copy data for selected device', and 'Copy connection string for selected device'. The last option is highlighted with a red rectangle.

Id	PrimaryKey	SecondaryKey	ConnectionStr	ConnectionStat	LastActivityTim	La
MyClient-Win10IoTEnt	S2ITnvIHDJiYWa8QQKnTuelp2Edl1p0QicjfXMG3w/4=	fS937Lvbod0ZjL2V5HhGBgUkV+B0jypw5/vjgQv0hjc=	HostName=A	Disconnected		

Step 4. 장치에 Azure IoT SDK 빌드 및 배포

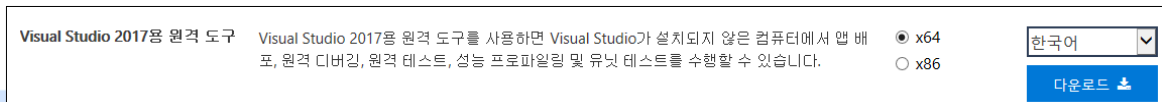
1. 장치 연결

① Azure IoT Certification 을 받을 장비(Windows 기반)를 인터넷 접속 가능하도록 네트워크에 연결 (유/무선 모두 가능)

② Visual Studio 2017 가 설치된 개발 시스템과 해당 장비를 연결
*해당 가이드에서는 원격 도구를 이용한 장비 연결 방법을 설명함

✓ Visual Studio 2017용 원격 도구 다운로드

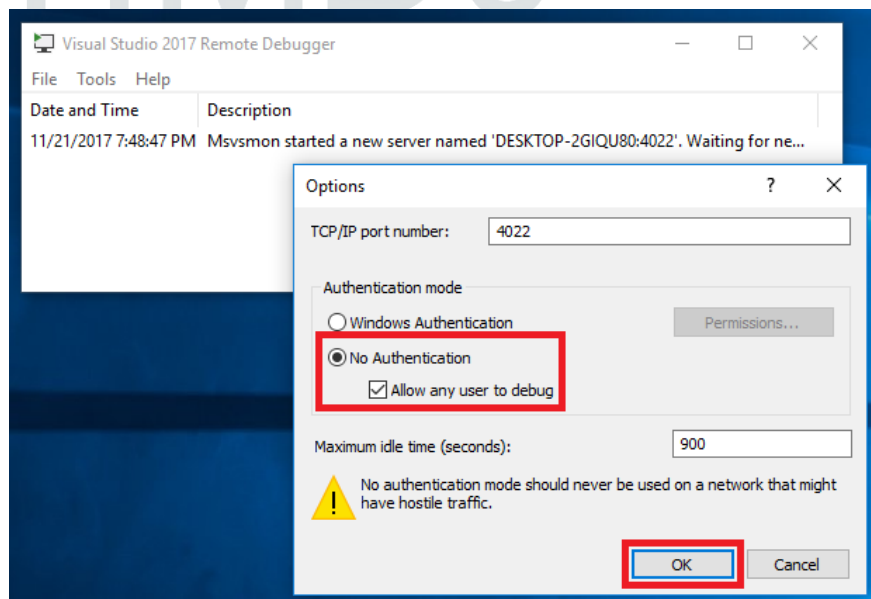
<https://www.visualstudio.com/ko/downloads/?rr=https%3A%2F%2Fdeveloper.microsoft.com%2Fko-kr%2Fwindows%2Fdownloads>



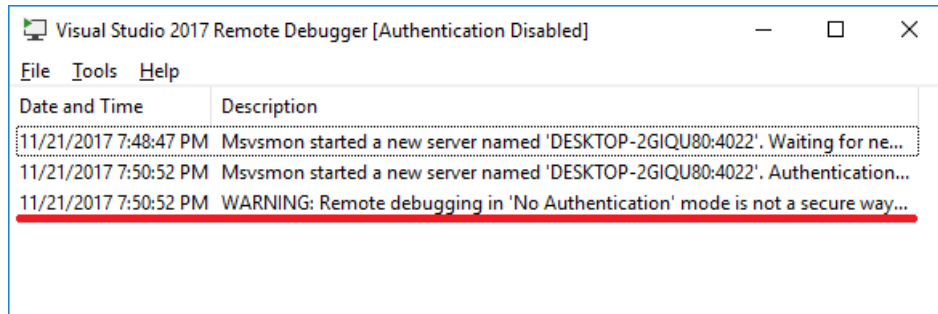
✓ Azure IoT Certification 을 받을 장비에 다운로드 받은 원격 도구 설치 및 실행

✓ Visual Studio 2017 원격 디버거 메뉴에서 '도구' - '옵션' 항목으로 이동

✓ 인증 모드를 '인증 안함', '모든 사용자가 디버깅할 수 있도록 허용' 값 선택 후 '확인' 클릭

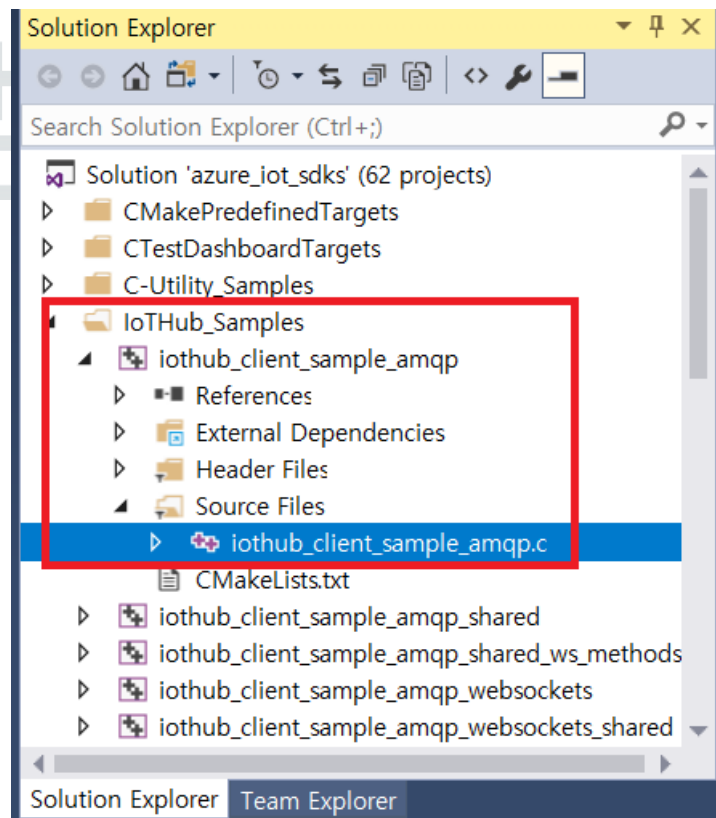


✓ '인증 안 함' 모드로 프로그램 실행 대기



2. IoT SDK Sample 빌드

- ① 개발 시스템에서 이전에 생성한 CMake_SDKs_x64 프로젝트 폴더로 이동
- ② azure_iot_sdks.sln 솔루션 파일 실행
- ③ 오른쪽 Solution Explorer 에서 사용하고자 하는 Protocol 에 맞는 샘플 선택
 - ✓ AMQP / HTTP / MQTT 중 원하는 Protocol 선택
(본 문서에서 AMQP 프로토콜 사용)
- ④ 'IoTHub_Samples' - 'iothub_client_sample_amqp' - 'Source Files' 항목으로 이동 후 'iothub_client_sample_amqp.c' 선택하여 소스코드 열기



3. Device Connection String 입력 및 프로젝트 빌드

- ① 선택한 'iothub_client_sample_amqp.c' 소스 파일의 'connectionString' 변수에 Device Explorer 에서 복사한 'Connection String for Selected Device' 값 입력 후 프로젝트 빌드

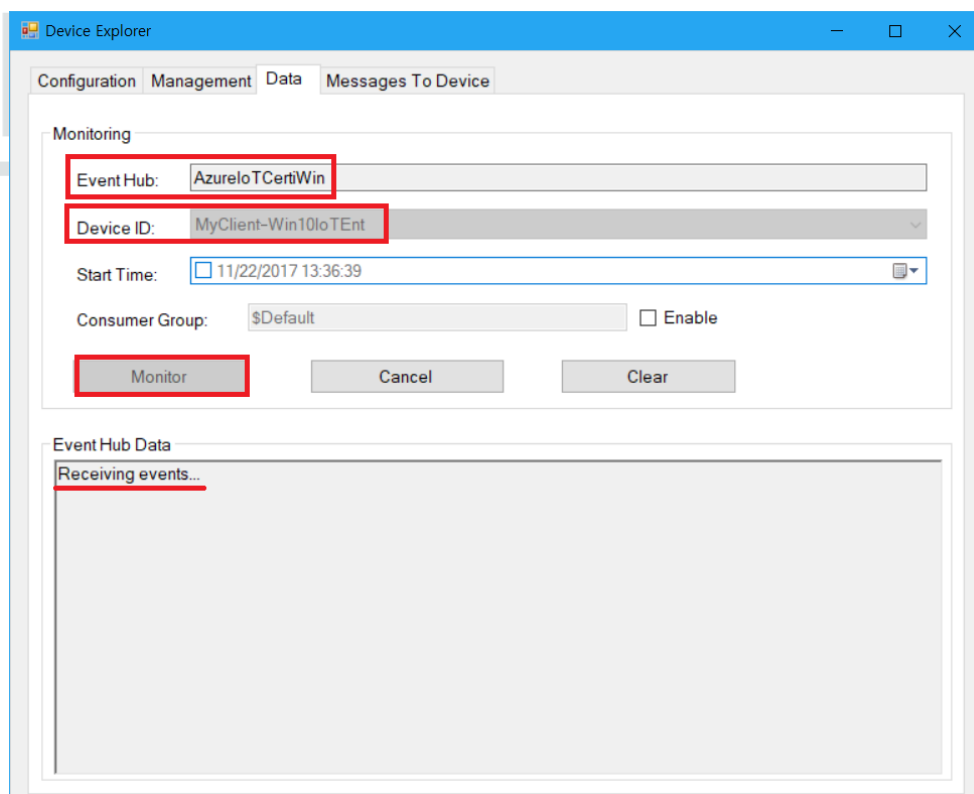
```
/*String containing Hostname, Device Id & Device Key in the format: */
/* "HostName=<host_name>;DeviceId=<device_id>;SharedAccessKey=<device_key>" */
/* "HostName=<host_name>;DeviceId=<device_id>;SharedAccessSignature=<device_sas_token>" */
static const char* connectionString = "[device connection string]";
```

```
/*String containing Hostname, Device Id & Device Key in the format: */
/* "HostName=<host_name>;DeviceId=<device_id>;SharedAccessKey=<device_key>" */
/* "HostName=<host_name>;DeviceId=<device_id>;SharedAccessSignature=<device_sas_token>" */
static const char* connectionString = "HostName=azureiotcentral.azure-devices.net;DeviceId=MyClient-Win10IoTEnt
```

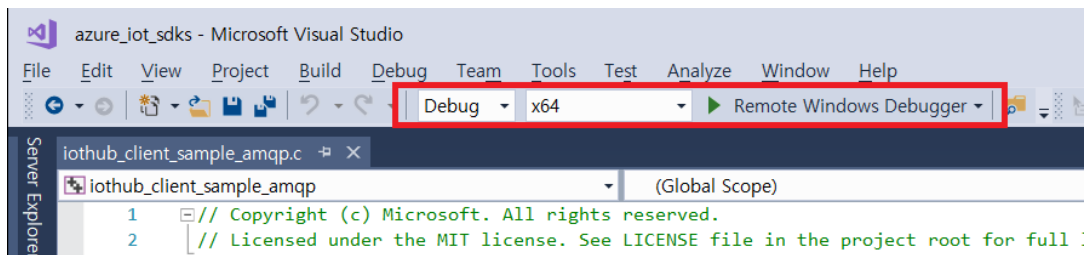
Step 5. Azure IoT Client SDK 샘플 실행 및 유효성 검사

1. 장치와 Azure IoT Hub 간의 통신 유효성 검사 단계 진행

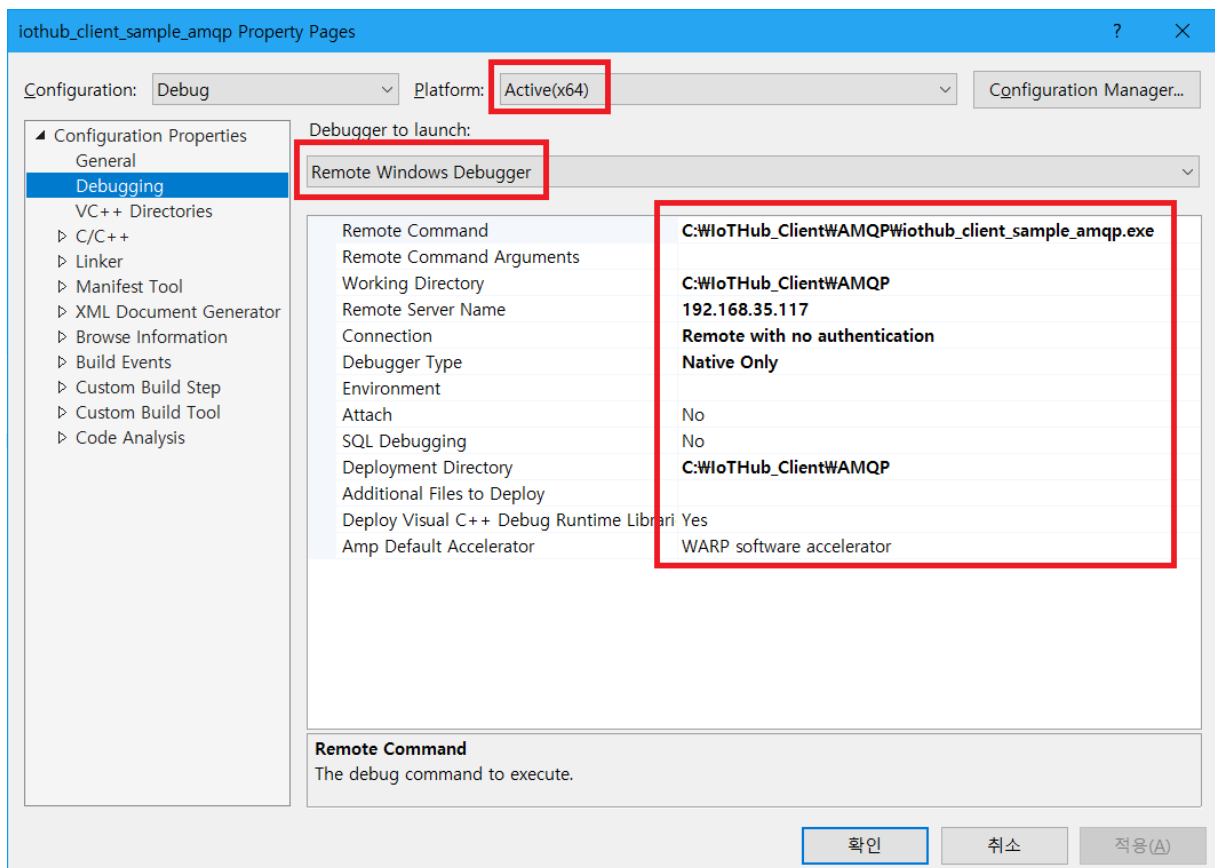
- ① 개발 시스템에서 Device Explorer 실행 후 'Data' 탭의 'Monitor' 버튼 클릭



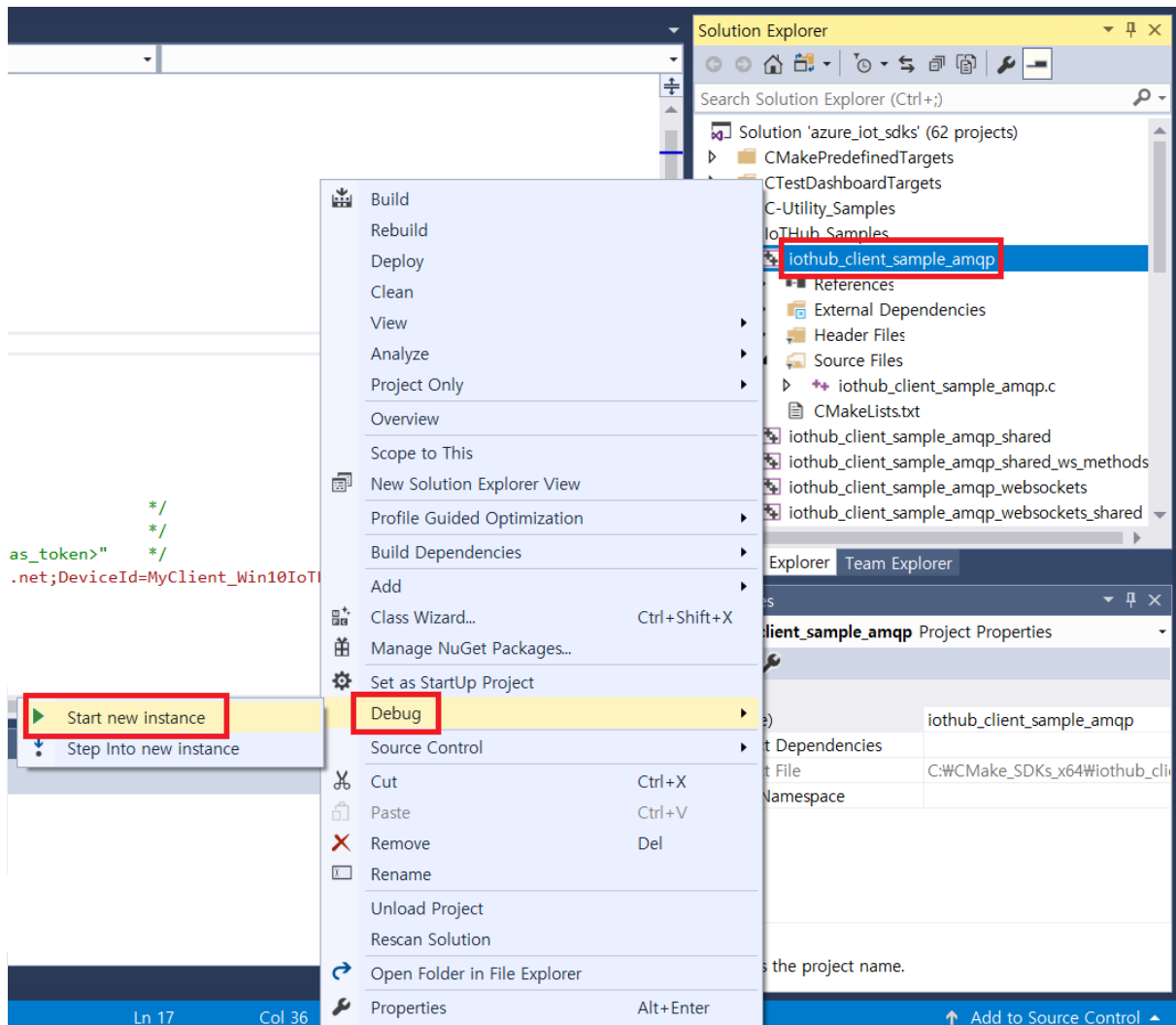
- ② 수정한 SDK Sample 프로젝트의 디버깅 모드를 '원격 윈도우즈 디버거' 모드로 변경



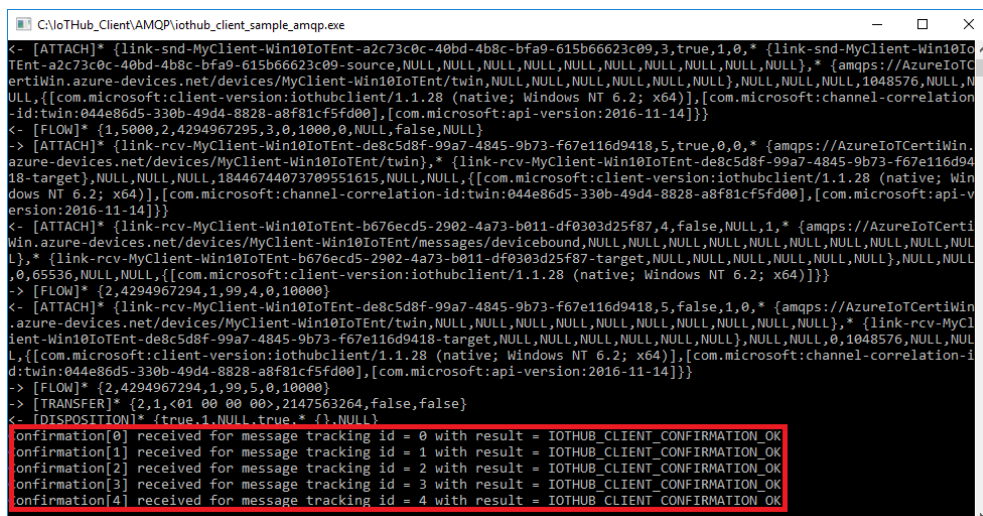
- ③ Azure IoT Certification 을 받을 장치에 대한 정보 기입하여 '원격 윈도우 디버거' 연결 설정



- ④ Solution Explorer 상에서 수정한 SDK Sample 프로젝트 선택 후, (iothub_client_sample_amqp) 마우스 우클릭하여 '디버그' - '새 인스턴스 시작' 실행

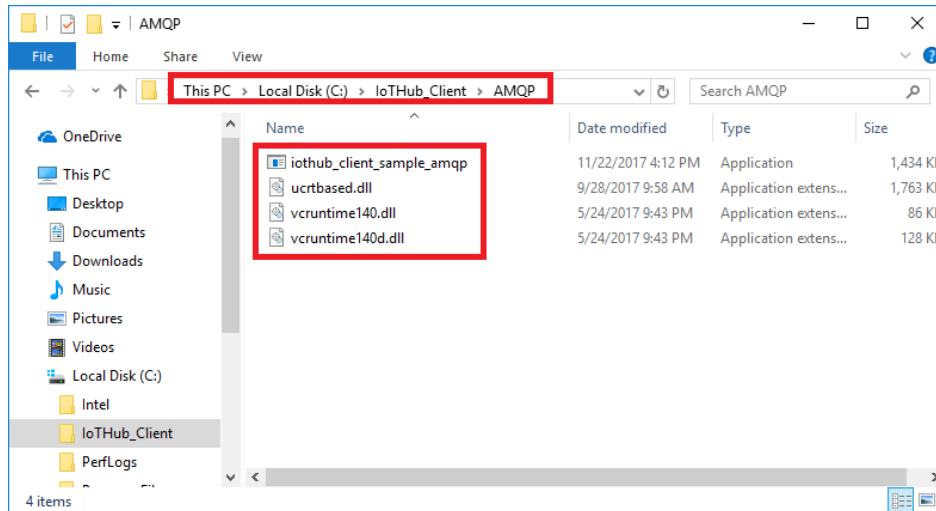


- ⑤ 해당 장비에 원격 빌드가 완료되면, 해당 장비에서 CMD 명령창 기반으로 `iothub_client_sample_amqp.exe` 가 실행되며, 장비와 IoT Hub 간의 연결 성공 메시지를 'IOTHUB_CLIENT_CONFIRMATION_OK' 출력 (본 예시: AMQP Protocol)

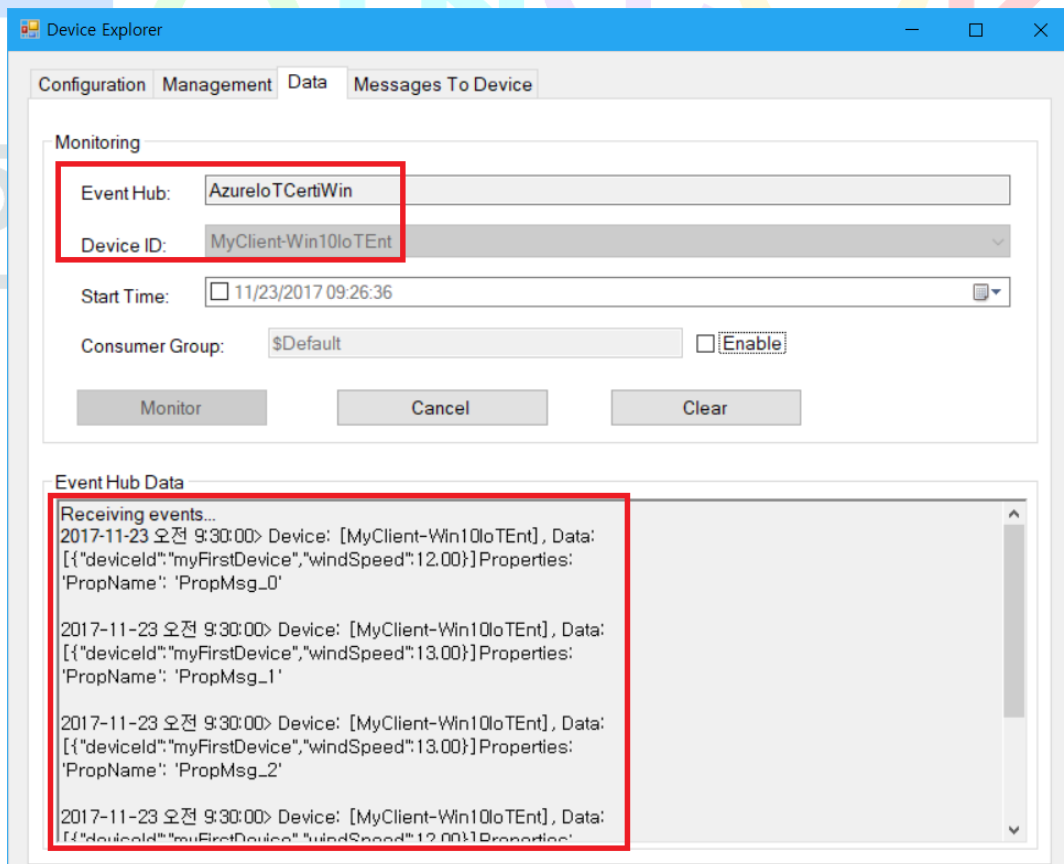


※ HTTP 와 MQTT Protocol 인 경우도 연결 성공 메시지 출력

⑥ 해당 장비에 빌드한 프로젝트가 지정한 경로로 배포된 것을 확인

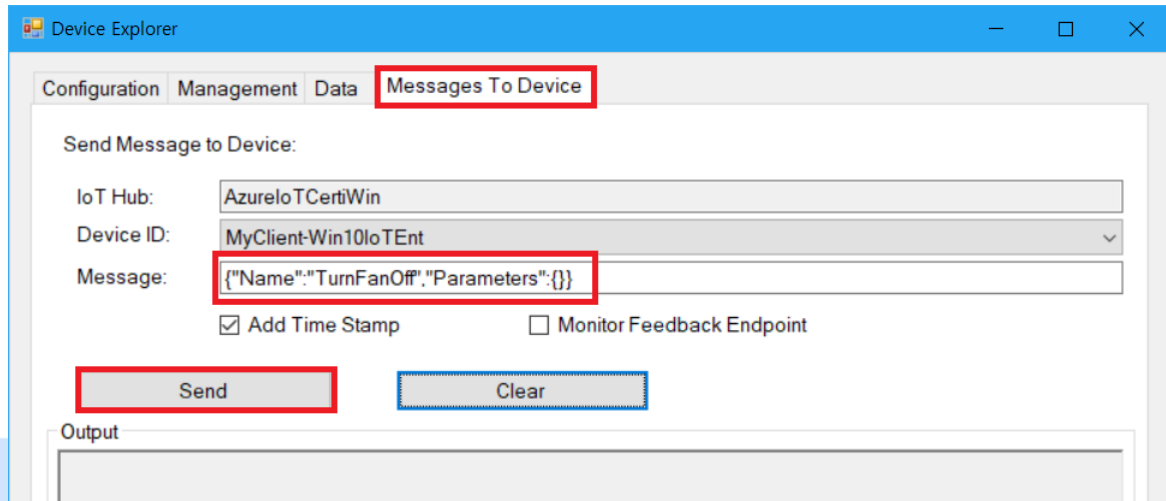


⑦ 해당 장비에서 IoT Hub 로 데이터가 정상적으로 보내진 경우 개발 시스템의 Device Explorer 에 다음과 같이 표시 (본 예시: AMQP Protocol)

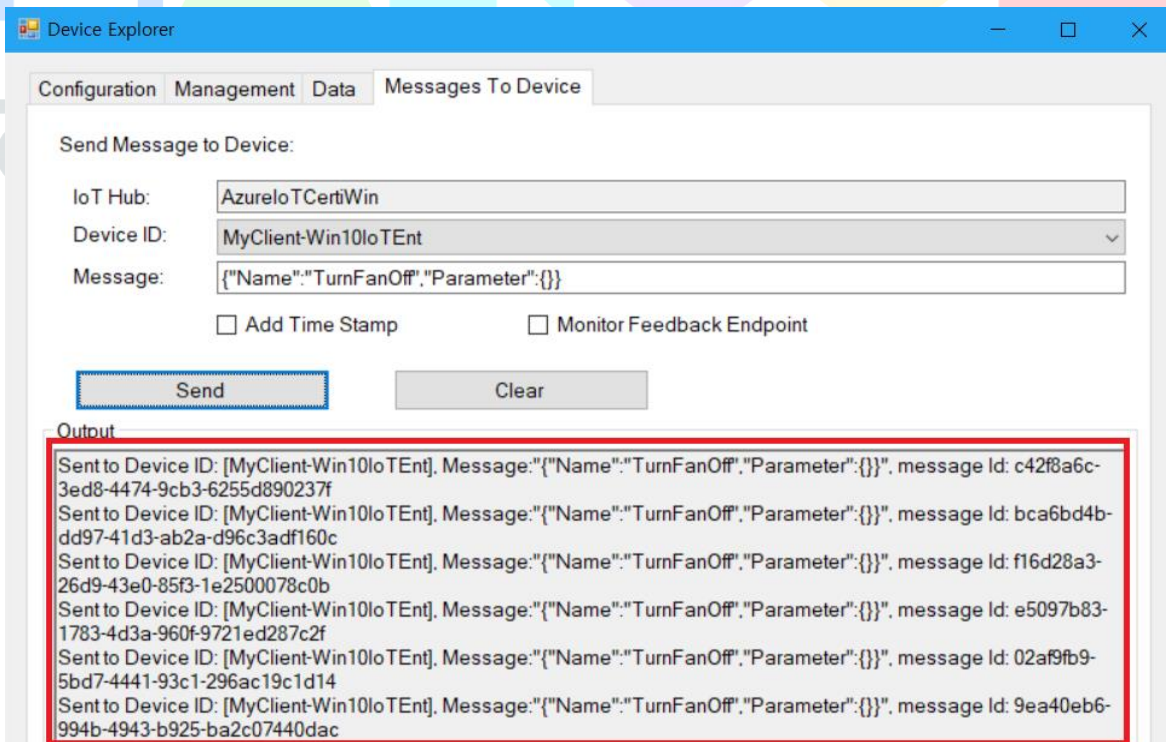


※ HTTP 와 MQTT Protocol 인 경우도 데이터 송신 성공 메시지 출력

- ⑧ 반대로, IoT Hub 에서 장비로 메시지 전송을 확인하기 위해 Device Explorer 의 'Messages to Device' 탭 클릭
- ⑨ Device ID 항목에 해당 장치를 선택한 후 'Message' 항목에 전송할 메시지를 입력 후, 'Send' 클릭 (예: {"Name":"TurnFanOff","Parameters":{}})



- ⑩ Device Explorer 에서 IoT Hub 에서 장비로의 메시지 전송 결과 확인



- ⑪ 장비에서 메시지 정상 수신 시, 아래와 같이 콘솔 창에 수신 메시지 출력 (본 예시: AMQP)

```
C:\IoT_Hub_Client\AMQP\iothub_client_sample_amqp.exe
Content-Type: <null>
Content-Encoding: <null>
BINARY Data: <<<2017-11-23 ?? 9:39:27 - {"Name":"TurnFanOff","Parameters":{}}>>> & Size=61
Message Properties:
    Key: iothub-ack Value: full

> [DISPOSITION]* {true,4,4,true,* {}}
- [TRANSFER]* {4,5,<25 58 E3 37 74 93 B1 4B A9 64 43 0F CE 69 8A 98>,0,NULL,false,NULL,NULL,NULL,NULL,false}
received Message [4]
Message ID: 37f65f6b-a63a-4e73-b051-8ff06db4deae
Correlation ID: <null>
Content-Type: <null>
Content-Encoding: <null>
BINARY Data: <<<2017-11-23 ?? 9:39:33 - {"Name":"TurnFanOff","Parameters":{}}>>> & Size=61
Message Properties:
    Key: iothub-ack Value: full

> [DISPOSITION]* {true,5,5,true,* {}}
- [TRANSFER]* {4,6,<2B 96 5A 49 54 53 D1 41 9C 21 9B 07 BC B7 DA 35>,0,NULL,false,NULL,NULL,NULL,NULL,false}
received Message [5]
Message ID: a75f3384-103d-4d45-9872-ba1e75848859
Correlation ID: <null>
Content-Type: <null>
Content-Encoding: <null>
BINARY Data: <<<2017-11-23 ?? 9:39:37 - {"Name":"TurnFanOff","Parameters":{}}>>> & Size=61
Message Properties:
    Key: iothub-ack Value: full
```

※ HTTP 와 MQTT Protocol 인 경우도 데이터 수신 성공 메시지 출력

Step 6. 로그 패키징 및 공유

1. Azure IoT SDK 프로젝트 빌드 로그와 Sample 테스트 결과 패키징

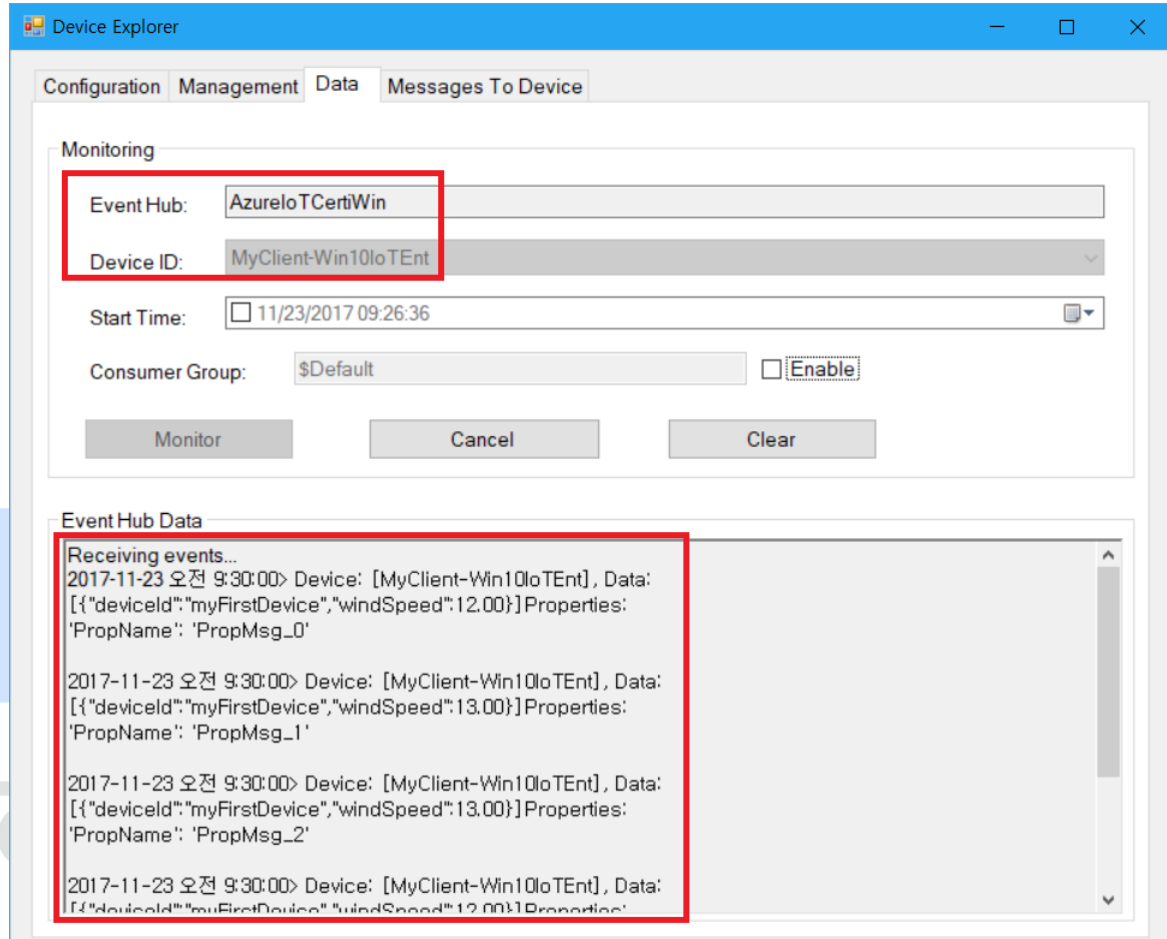
① Visual Studio 에서 SDK Sample 프로젝트를 빌드한 로그 수집 (Step 4 - 5)

```
1 build log from section.txt - 메모장
파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)
6> ----- Rebuild All started: Project: iothub_client_amqp_transport, Configuration: Release Win32 -----
6> Building Custom Rule C:/AzureIoTSDKs/c/iothub_client/CMakeLists.txt
6> Building Custom Rule C:/AzureIoTSDKs/c/iothub_client/CMakeLists.txt
6> CMake does not need to re-run because C:/WCMake_SDKs/Wiothub_client/WCMakeFiles/Wgenerate.stamp is up-to-date.
6> CMake does not need to re-run because C:/WCMake_SDKs/Wiothub_client/WCMakeFiles/Wgenerate.stamp is up-to-date.
6> iothub_client.c
6> version.c
6> iothub_message.c
6> version.c
6> iothub_client_ll.c
6> iothubtransport.c
6> blob.c
6> Generating Code...
6> parson.c
6> iothub_client_ll_uploadtoblob.c
6> iothub_client.vcxproj -> C:/WCMake_SDKs/Wiothub_client/WRelease/Wiothub_client.lib
6> iothubtransportamqp.c
6> C:/WCMake_SDKs/Wiothub_client/src/Wiothubtransportamqp.c : warning C4819: The file contains a character that cannot be represented in the current code page: UTF-8.
6> C:/WCMake_SDKs/Wiothub_client/src/Wiothubtransportamqp.c(566): warning C4819: The file contains a character that cannot be represented in the current code page: UTF-8.
6> Generating Code...
6> iothub_client_amqp_transport.vcxproj -> C:/WCMake_SDKs/Wiothub_client/WRelease/Wiothub_client_amqp_transport.lib
7> ----- Rebuild All started: Project: simplesample_amqp, Configuration: Release Win32 -----
7> Building Custom Rule C:/AzureIoTSDKs/c/serializer/samples/simplesample_amqp/CMakeLists.txt
7> CMake does not need to re-run because C:/WCMake_SDKs/Wserializer/Wsamples/Wsimplesample_amqp/WCMakeFiles/Wgenerate.stamp is up-to-date.
7> simplesample_amqp.c
7> main.c
7> Generating Code...
7> simplesample_amqp.vcxproj -> C:/WCMake_SDKs/Wserializer/Wsamples/Wsimplesample_amqp/WRelease/Wsimplesample_amqp.exe
***** Rebuild All: 7 succeeded, 0 failed, 0 skipped *****
```

② 장비에서 IoT Hub 로 메시지를 보내는 과정 스크린샷 수집

✓ Send Device Events to IoT Hub (콘솔 창과 Device Explorer 모니터링 화면)

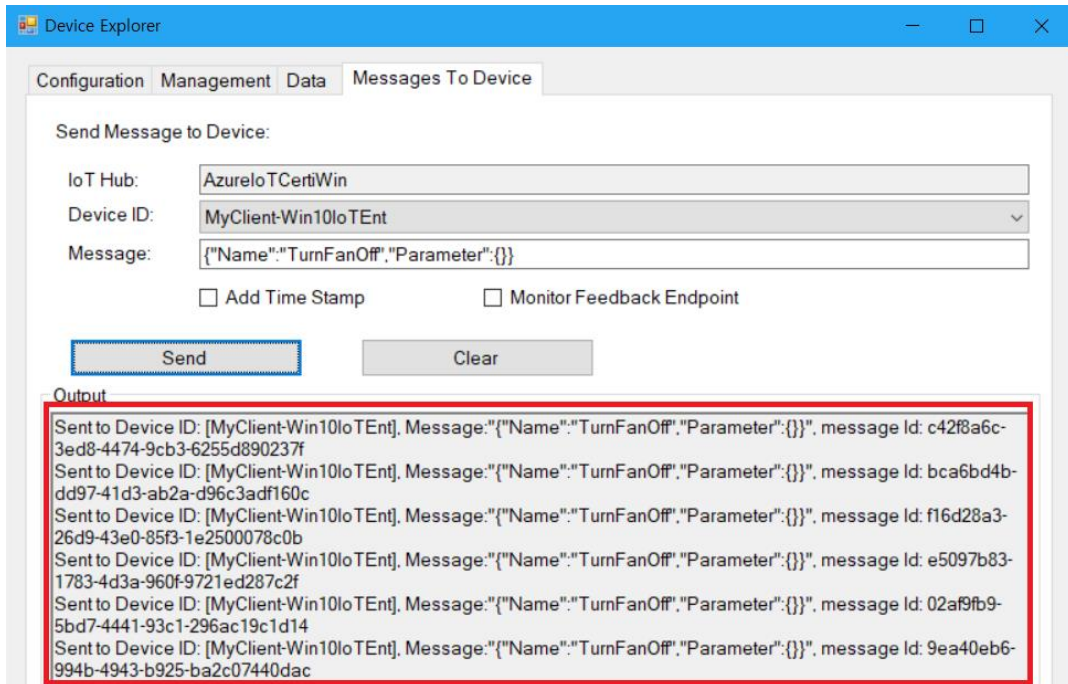

```
C:\IoT_Hub_Client\AMQP\iothub_client_sample_amqp.exe
confirmation[0] received for message tracking id = 0 with result = IOTHUB_CLIENT_CONFIRMATION_OK
confirmation[1] received for message tracking id = 1 with result = IOTHUB_CLIENT_CONFIRMATION_OK
confirmation[2] received for message tracking id = 2 with result = IOTHUB_CLIENT_CONFIRMATION_OK
confirmation[3] received for message tracking id = 3 with result = IOTHUB_CLIENT_CONFIRMATION_OK
confirmation[4] received for message tracking id = 4 with result = IOTHUB_CLIENT_CONFIRMATION_OK
```



③ 장비가 IoT Hub 로부터 메시지를 받는 과정 스크린샷 수집

- ✓ Receive messages from IoT Hub (콘솔 창과 Device Explorer 모니터링 화면)

```
C:\IoT_Hub_Client\AMQP\iothub_client_sample_amqp.exe
> [DISPOSITION]* {true,4,4,true,* {} }
- [TRANSFER]* {4,5,<25 58 E3 37 74 93 B1 4B A9 64 43 0F CE 69 8A 98>,0,NULL,false,NULL,NULL,NULL,NULL,false}
Received Message [4]
Message ID: 37f65f6b-a63a-4e73-b051-8ff06db4deae
Correlation ID: <null>
Content-Type: <null>
Content-Encoding: <null>
BINARY Data: <<<2017-11-23 ?? 9:39:33 - {"Name":"TurnFanOff","Parameters":{}}>>> & Size=61
Message Properties:
Key: iothub-ack Value: full
> [DISPOSITION]* {true,5,5,true,* {} }
- [TRANSFER]* {4,6,<2B 96 5A 49 54 53 D1 41 9C 21 9B 07 BC B7 DA 35>,0,NULL,false,NULL,NULL,NULL,NULL,false}
Received Message [5]
Message ID: a75f3384-103d-4d45-9872-ba1e75848859
Correlation ID: <null>
Content-Type: <null>
Content-Encoding: <null>
BINARY Data: <<<2017-11-23 ?? 9:39:37 - {"Name":"TurnFanOff","Parameters":{}}>>> & Size=61
Message Properties:
Key: iothub-ack Value: full
```

④ 장비에서의 해당 Sample SDK 실행 방법에 대한 명확한 지침 작성

✓ 지침 작성 템플릿

2. Azure IoT Certification Team 에게 생성한 로그 패키징 및 지침 파일을 압축하여 메일에 첨부하여 전송

① 수신자: Azure IoT Certification Team, iotcert@microsoft.com

② 인증 요청 메일 내용 예시

Dear Azure IoT Certification Team,

Hello,

Our company name is Hancom MDS Inc..

This e-mail is to request your team that azure iot certificate.

In response to your request, we are attaching a result log file.

We look forward to your reply.

Thanks
Darby

[Device Information]

1. Device : ABCD-1234 (Intel XXX XXXX)
2. Platform : Windows 10 IoT Enterprise 2016 LTSB (Korean)

3. 다음 단계 진행 관련하여 Azure IoT Certification Team 으로부터 48~72시간 내에 연락이 옵니다.

Step 7. Microsoft Azure Certification 진행 완료 메일 수신

1. Azure IoT Certified for IoT Support Team 으로부터 하기와 같은 메일 수신

✓ 메일 본문 내용 및 첨부 파일

Dear Darby,

Congratulations on successfully completing Microsoft Azure Certification testing!

...

....

.....

Regards,

The Azure Certified for IoT Support Team

메시지

Azure Certified for IoT Badge Guidelines.zip (403 KB)

Microsoft Azure Certified for IoT_Partner Press Release and Blog Template_January 19 2016.docx (42 KB)

끝.

한컴MDS

국내 1위의 임베디드 솔루션 전문기업을 넘어 **인텔리전트 융합 솔루션 전문 기업**으로

(주) 한컴MDS (구. MDS테크놀로지) <http://www.hancommds.com>

13493 경기도 성남시 분당구 대왕판교로 644번길 49 한컴타워 3, 4층

전화 031-627-3000 팩스 031-627-3100