



Software Version: 1.9.3

Installation Guide v1.9

Document Release Date: December 2016

Software Release Date: December 2016



저작권 및 면책조항

본 매뉴얼은 기술상의 부정확함 또는 글자상의 오류를 포함하고 있으며, 저자는 여기에 관한 어떠한 책임도 지지 않습니다.

본 매뉴얼의 내용은 정기적으로 변경 또는 추가 될 수 있으며, 그에 따른 변경본은 본 매뉴얼에 "New Edition"의 형태로 순차적으로 추가될 것입니다. 사전 공지 없이 언제든지 본 매뉴얼에 기재된 제품 및 프로그램을 변경 또는 수정할 수 있습니다.

Mobius 서버 플랫폼 소스코드는 하기의 라이선스 정책에 따라 배포됩니다.

- OCEAN 기반 오픈 소스코드는 3-clause BSD-style 라이선스를 기반으로 배포한다. 3-clause BSD-style 라이선스에 따라, 소스코드 파일에 저작권 헤더를 유지하는 동안 OCEAN 기반 오픈 소스 코드는 상업용 또는 비 상업용 시스템에 자유롭게 사용할 수 있다.
- OCEAN 기반 오픈 소스 라이선스는 개발된 소스코드를 다른 사람과 공유하는 것을 요구하지는 않는다. OCEAN기반 오픈 소스를 활용하여 개발한 소스코드는 개발자의 것이고 누구와 공유할 의무는 없다.
- 누구나 OCEAN의 오픈 소스 발전에 공헌할 수 있다. 그러나, 개발된 소스코드가 OCEAN에 공헌되기를 원한다면 이 코드는 OCEAN의 라이선스 정책을 따르도록 요구된다.

Copyright (c) 2014, OCEAN
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

내 용

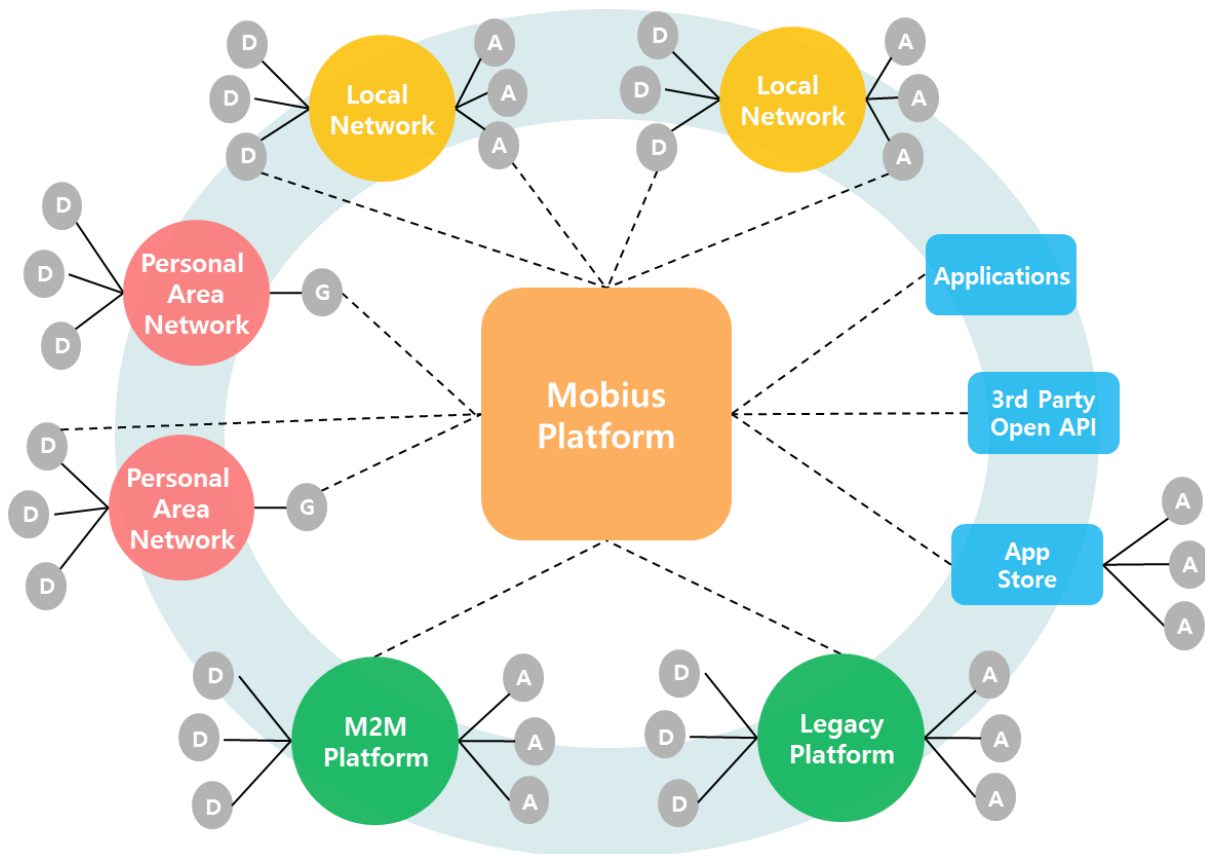
1. Mobius 소개	5
1.1. Mobius 란?.....	5
1.1.1. Mobius 정의.....	5
1.2. Mobius Yellow Turtle 플랫폼	6
1.2.1. Mobius Yellow Turtle 서버 플랫폼 개요.....	6
1.2.2. Mobius Yellow Turtle 서버 플랫폼 주요 기능	7
1.2.3. Mobius Yellow Turtle 서버 플랫폼 구성.....	7
1.2.4. Mobius Yellow Turtle 서버 플랫폼 S/W 아키텍처	8
2. Mobius Yellow Turtle 서버 플랫폼 설치	11
2.1. 설치 개요	11
2.2. Mobius Yellow Turtle 서버 설치 (Windows).....	11
2.2.1. MySQL 설치.....	11
2.2.2. MQTT 서버 설치.....	26
2.2.4. Mobius Yellow Turtle 서버 설치.....	33

1. Mobius 소개

1.1. Mobius란?

1.1.1. Mobius 정의

Mobius 플랫폼은 IoT(Internet of Things) 서비스 제공을 위해 다양한 IoT Device 정보를 관리하고, 이들 IoT Device의 접근 제어, 인증, 사용자 관리, 복수의 IoT 서비스 조합을 제공하여 어플리케이션을 통해 서비스하기 위한 플랫폼이다.



[Mobius(Internet of Things) 플랫폼 개념도]

1.2. Mobius Yellow Turtle 플랫폼

1.2.1. Mobius Yellow Turtle 서버 플랫폼 개요

Mobius Platform은 물리적 기반인 통신 수단을 사용하여 인터넷에 연결된 사물들이 원활하게 작동하도록 관리하고 사물들의 데이터를 저장하는 서버 소프트웨어 플랫폼으로 일종의 미들웨어라고 말할 수 있다. 이처럼 Mobius Server Platform은 디바이스와 어플리케이션의 통신 연결을 쉽고 편리하게 하며, 개방형 개발 환경에서 누구든지 IoT 서비스를 만들어 내고 사용할 수 있는 생태계를 구축한다.



[Mobius Platform 개요]

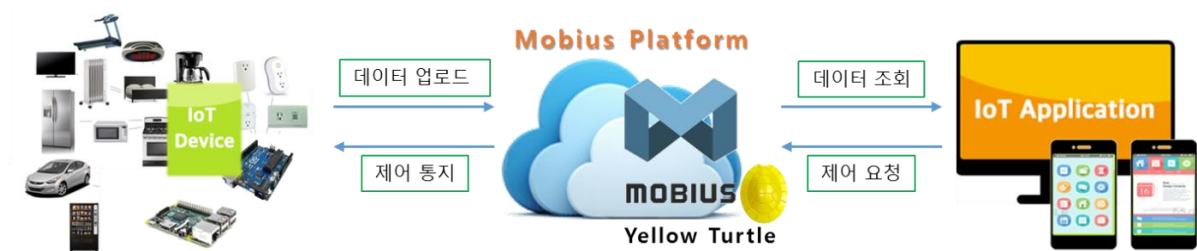
이러한 Mobius를 구현하기 위해 현재 다양한 언어로 개발될 수 있고 개발하고 있다. Mobius Yellow Turtle은 node js 를 사용하여 개발된 Mobius 서버 플랫폼이다. 이전부터 개발되어온 Java Spring Framework 버전은 Blue Octopus 라고 한다.

Mobius Yellow Turtle은 node js의 express framework을 사용하지 않고 express 모듈을 사용하여 개발되었고, http, xml 등 다양한 모듈을 사용하고 있다.

Mobius Platform은 oneM2M 표준을 준수하여 Http, MQTT, CoAP 등 다양한 프로토콜과 Open API를 제공한다. 계층적인 리소스 구조를 가지고 있으며, 기본적으로 Http RESTful 방식을 제공합니다. 현재 공개된 Mobius Yellow Turtle 버전 Http, MQTT를 지원한다. Mobius Yellow Turtle Platform은 Http Open API를 위한 Http 서버와 MQTT 지원을 위한 MQTT 서버로 구성되어 있으며, 리소스 저장을 위한 DB는 MySQL DBMS를 사용한다.

1.2.2. Mobius Yellow Turtle 서버 플랫폼 주요 기능

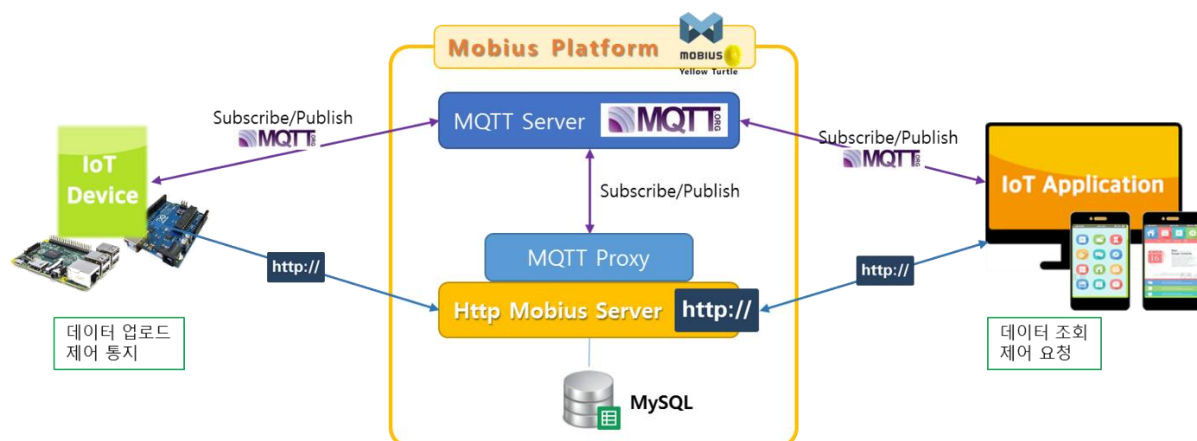
Mobius Yellow Turtle Platform은 디바이스와 어플리케이션을 연결하는 중간 매개체로서 디바이스는 서버 플랫폼으로 데이터를 전송하고, 플랫폼은 데이터를 저장한다. 어플리케이션은 Open API를 통해 플랫폼에 저장된 데이터를 조회하고 플랫폼으로 제어 요청을 전송한다. 플랫폼은 설정된 디바이스에 따라 어플리케이션의 제어요청을 다시 디바이스로 전달한다.



[Mobius 서버 플랫폼 주요 기능]

1.2.3. Mobius Yellow Turtle 서버 플랫폼 구성

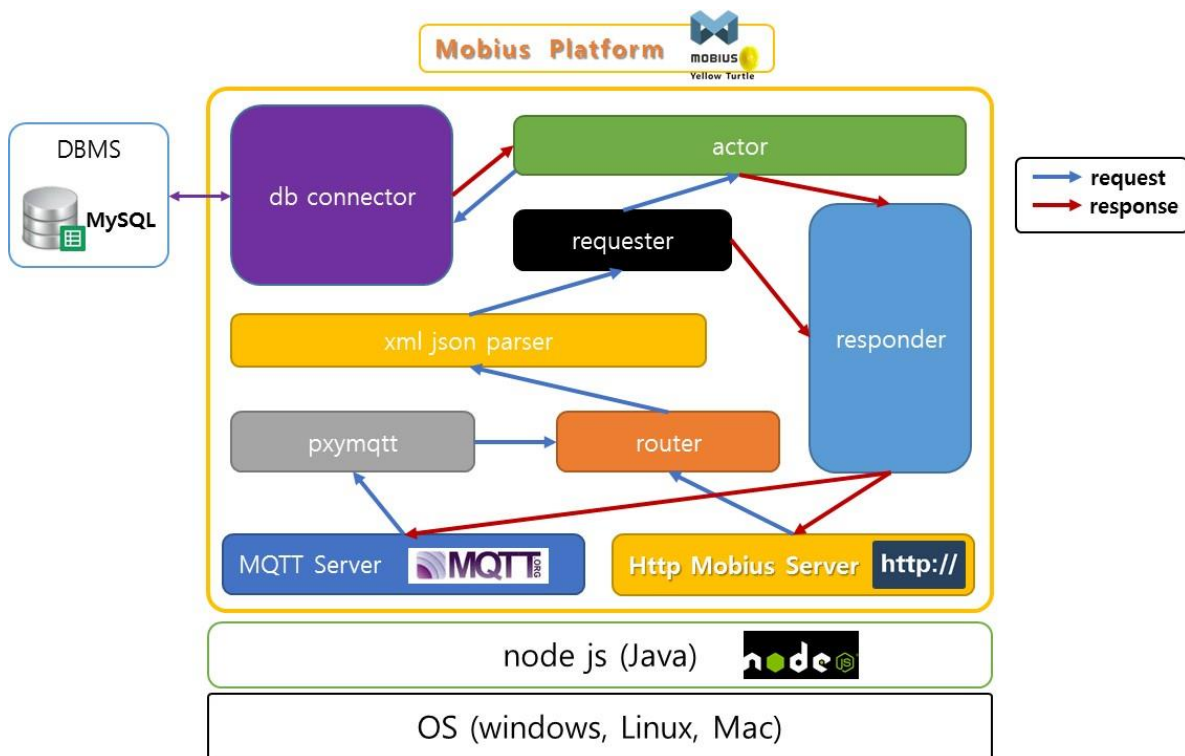
Mobius Platform은 디바이스와 어플리케이션을 연결하며, 디바이스는 HTTP, CoAP, MQTT 프로토콜을 지원하며, Application은 HTTP를 지원합니다. Mobius 서버의 DB는 MySQL을 사용하여 구성되어 있습니다.



[Mobius 서버 플랫폼 시스템 구성]

1.2.4. Mobius Yellow Turtle 서버 플랫폼 S/W 아키텍처

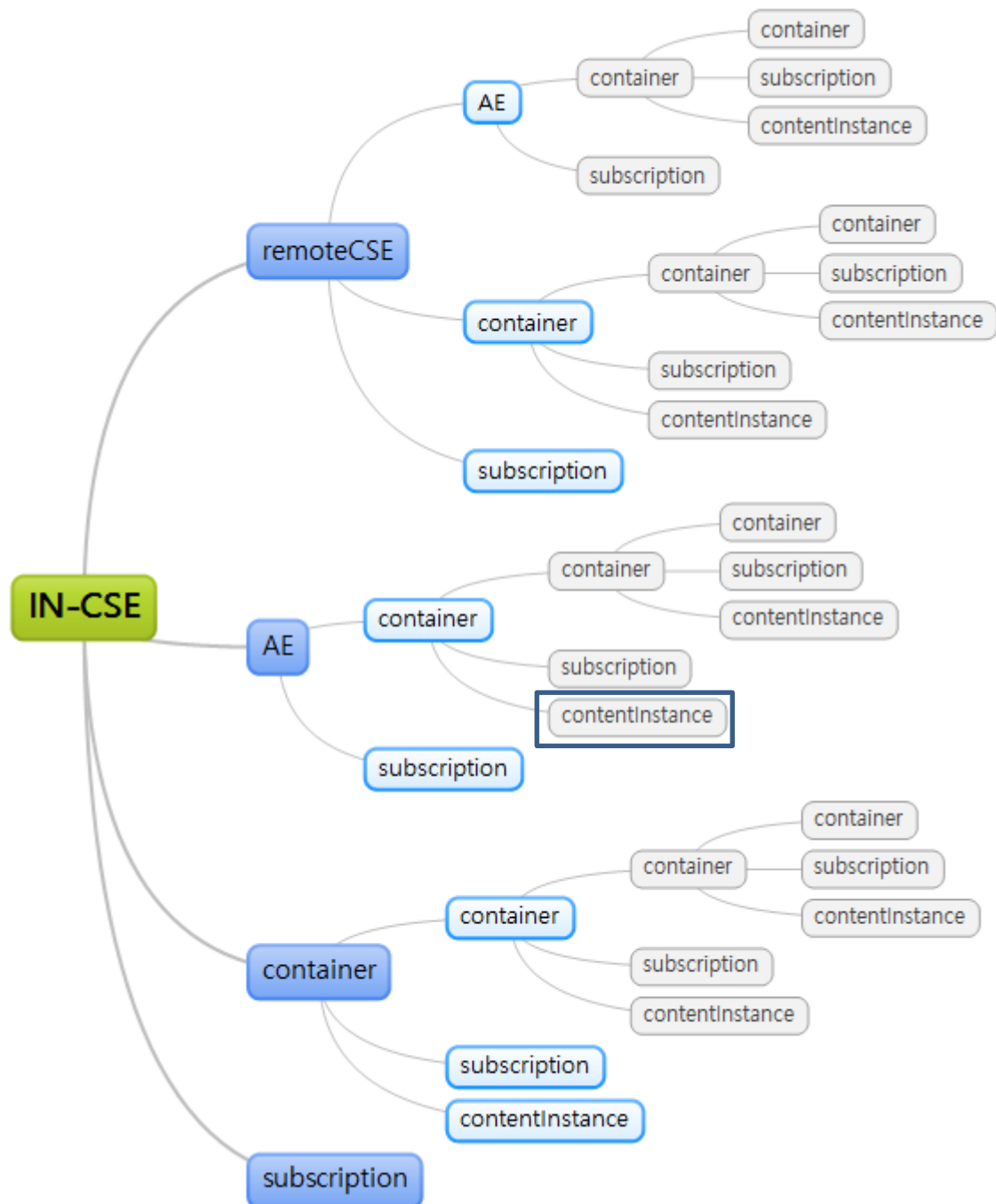
Mobius 의 Yellow Turtle 버전은 Node JS를 기반으로 Java Script로 개발되었다. DB는 MySQL을 지원하고 있으며 통신 프로토콜은 HTTP, MQTT를 지원한다. 아래는 Mobius Yellow Turtle의 소프트웨어 아키텍처를 나타낸다.



[Mobius Yellow Turtle S/W 아키텍처]

위의 아키텍처는 컴포넌트 개념의 기능 아키텍처를 나타낸다. Mobius Yellow Turtle 플랫폼은 기본적으로 http 프로토콜을 지원한다. 위의 그림처럼 MQTT의 경우 내부적으로 MQTT → Http로 변환하는 proxy를 구현하여 MQTT를 지원하고 있다. 따라서 모든 Http request는 router로 보이는데 이 router가 일반적인 main 역할을 수행하며 parser - actor를 거쳐 실제 데이터베이스에 데이터를 저장하도록 컨트롤한다. 응답은 responder에 의해 처리된다.

1.2.5. Mobius Yellow Turtle 서버 지원 Resource 구조



[지원 Resource 구조 체계]

Mobius Yellow Turtle은 위의 그림과 같은 리소스 체계를 가지고 이를 지원하고 있다. IN-CSE가 CSEBase Name이고 이는 서버가 실행될 때 자동으로 생성된다. 위의 리소스 체계

는 Resource 를 접근할 수 있는 URI와 그대로 매핑된다. 예를 들어 IN-CSE 밑에, AE 밑에, container 밑의 데이터를 접근하는 URL은 아래와 같다.

http://hostname:port/IN-CSE_name/AE_name/container_name/contentInstance_name

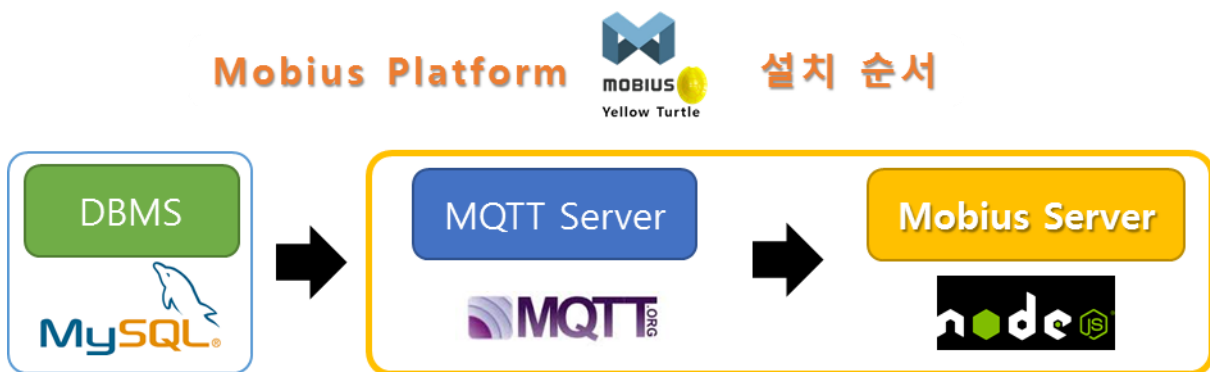
이처럼 모든 리소스는 위와 같은 방법으로 접근할 수 있으며 POST, GET, PUT, DELETE http method를 통해 생성, 조회, 업데이트, 삭제 동작을 수행할 수 있다.

현재 Mobius Yellow Turtle은 4단계까지의 URL만 지원하고 있다. Container 밑에 container를 지원할 수 있기 때문에 URL의 깊이는 제한하고 있지 않으나 서버 지원 정책상 4단계 깊이의 URL 정보로 모든 리소스를 다룰 수 있을 것으로 판단하고 있다.

2. Mobius Yellow Turtle 서버 플랫폼 설치

2.1. 설치 개요

Mobius Yellow Turtle Platform은 MySQL DBMS를 사용한다. 먼저 MySQL을 설치하고 MQTT 서버, Mobius 서버 순으로 설치를 하면 Mobius Yellow Turtle 서버 플랫폼 구축이 완성된다. MQTT 서버의 경우 MQTT를 지원하지 않는다면 필수적으로 설치하지 않아도 무방하다.



[Mobius 서버 플랫폼 설치 개요]

2.2. Mobius Yellow Turtle 서버 설치 (Windows)

2.2.1. MySQL 설치

Mobius Yellow Turtle은 데이터 저장 및 관리를 위해 MySQL DB를 기반으로 동작한다. 여기서는 오픈소스 RDBMS인 MySQL을 다운로드 받고 설치하는 방법에 대해 설명한다.

<http://www.mysql.com>



가. 다운로드 및 설치



Generally Available (GA) Releases

Development Releases

MySQL Community Server 5.6.22

Select Platform:

Microsoft Windows


Looking for previous GA versions?

Recommended Download:

MySQL Installer 5.6 for Windows

All MySQL Products. For All Windows Platforms. In One Package.

Starting with MySQL 5.6 the MySQL Installer package replaces the server-only MSI packages.




Windows (x86, 64-bit), MySQL Installer MSI

Download

Other Downloads:

<p>Windows (x86, 32-bit), ZIP Archive</p> <p>(mysql-5.6.22-win32.zip)</p>	<p>5.6.22</p> <p>MD5: 00abcb99a71708d372ff073f870deabd Signature</p>	<p>342.0M</p> <p>Download</p>
<p>Windows (x86, 64-bit), ZIP Archive</p> <p>(mysql-5.6.22-winx64.zip)</p>	<p>5.6.22</p> <p>MD5: 8810b875ff1651e3c91473faa7ed6509 Signature</p>	<p>347.5M</p> <p>Download</p>

 We suggest that you use the [MD5 checksums](#) and [GnuPG signatures](#) to verify the integrity of the packages you download.

<http://dev.mysql.com/downloads/mysql>

MySQL에 들어가 보면 MySQL의 다양한 버전이 있는데 위 링크를 클릭해서 들어가면 무료로 사용할 수 있는 GPL버전인 MySQL Community Server를 다운받을 수 있다. 여기서는 윈도우에 MySQL을 설치하기 위해 Windows (x86, 64-bit), MySQL Installer MSI라고 적힌 Download 버튼을 클릭한다. (MySQL에는 MySQL Enterprise Edition이나 MySQL Cluster CGE처럼 유료로 사용해야 하는(commercial) 것도 있으니 참고하길 바란다.)

웹 설치가 아닌, 설치파일 전부를 다운로드 받아 설치하기 위해서는 Zip Archive 다운로드 버튼을 클릭한다. 이 버전은 웹 설치가 어려운 곳에서 MySQL을 설치할 때 이 버전을 활용할 수 있다.

Begin Your Download - mysql-installer-community-5.6.2

Login Now or Sign Up for a free account.

An Oracle Web Account provides you with the following advantages:

- Fast access to MySQL software downloads
- Download technical White Papers and Presentations
- Post messages in the MySQL Discussion Forums
- Report and track bugs in the MySQL bug system
- Comment in the MySQL Documentation

Login »

using my Oracle Web account

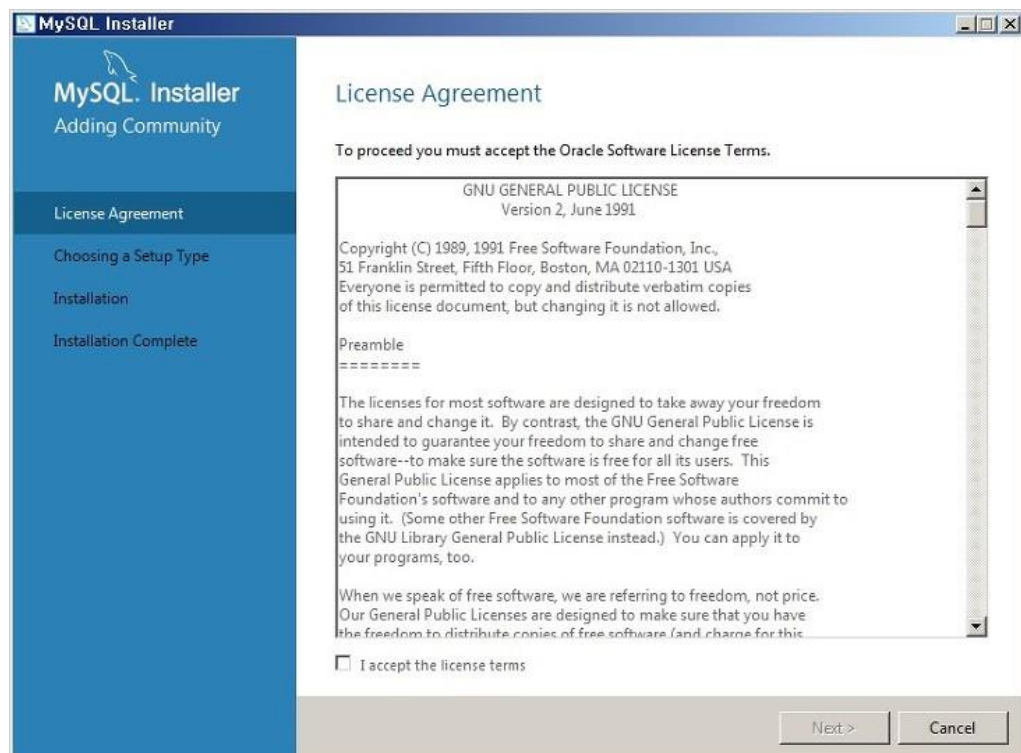
Sign Up »

for an Oracle Web account

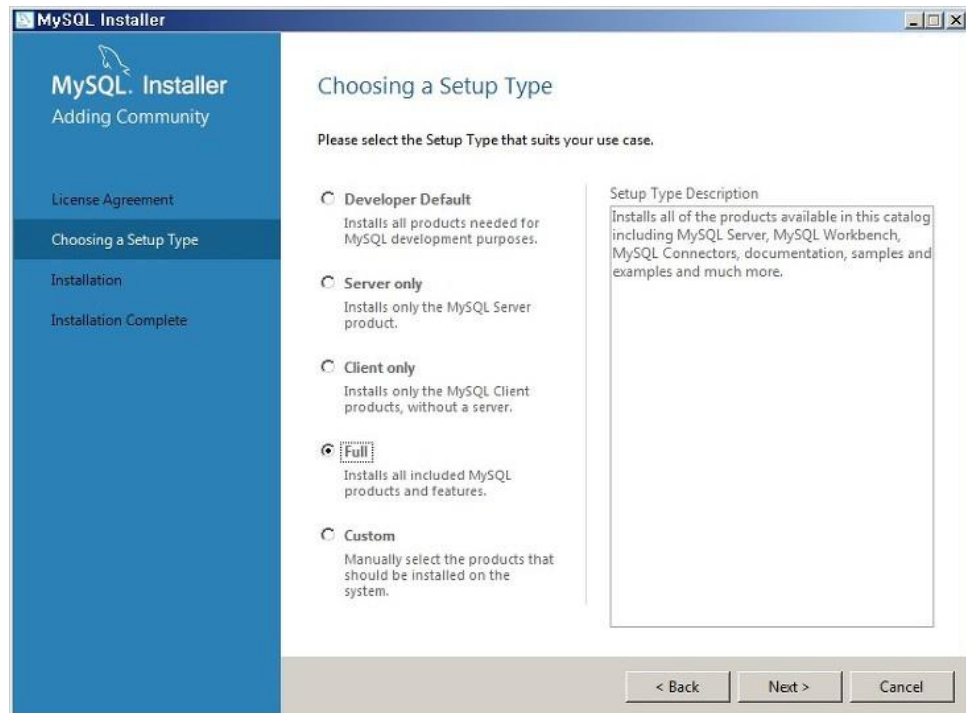
MySQL.com is using Oracle SSO for authentication. If you already have an Oracle Web account click the Login link. Otherwise, you can signup for a free account by clicking the Sign Up link and following the instructions.

No thanks, just start my download.

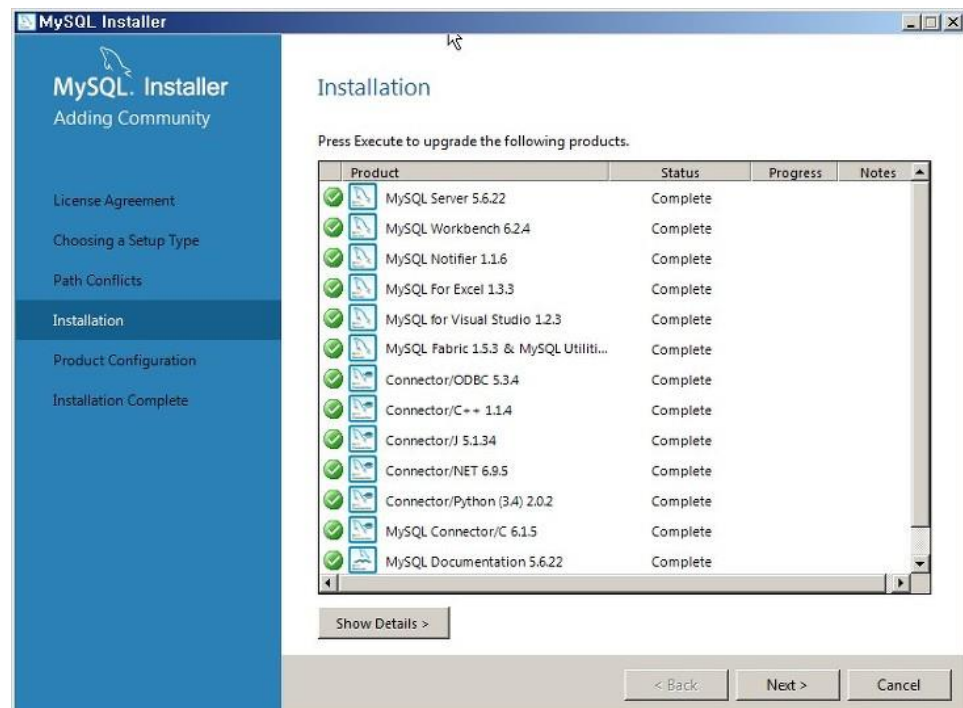
로그인 화면이 나오면 회원가입을 하거나 아래에 있는 **No thanks, just start my download** 링크를 클릭해서 다운로드 받는다.



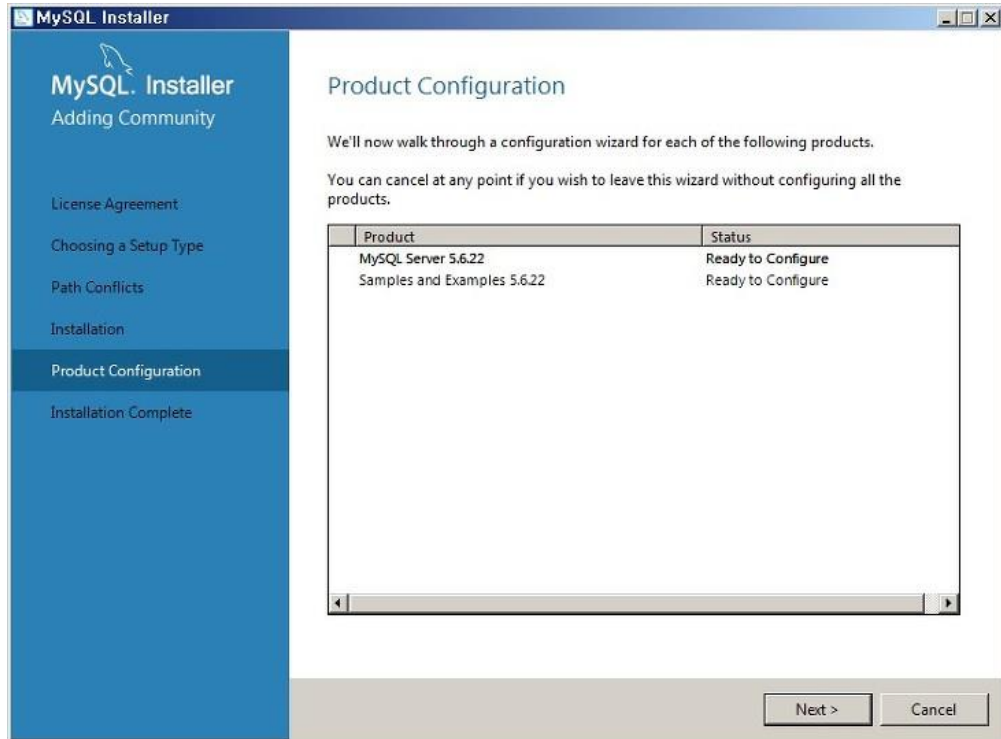
MySQL 설치파일을 다운받아서 실행시켜보면 위와 같은 화면이 나오면 라이선스에 동의 를 하고 Next 버튼을 누른다.



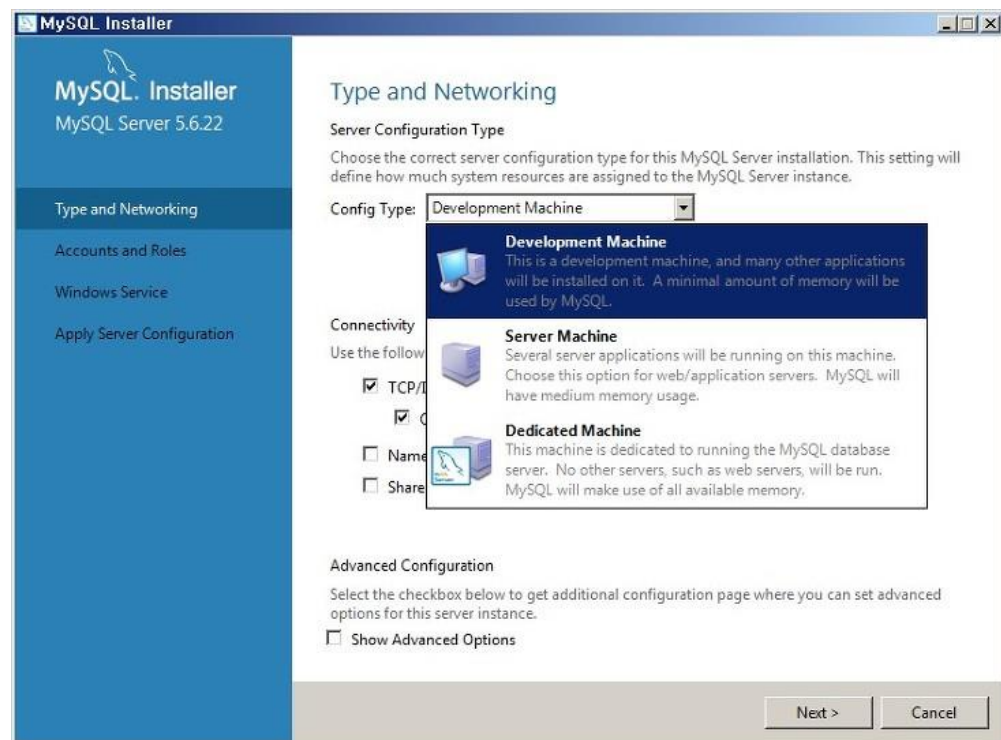
설치 스타일에는 Developer Default, Server only, Client only, Full, Custom 이렇게 5가지가 있는데, Mobius Yellow Turtle을 위해서는 MySQL Server와 MySQL Workbench가 필요하나 여기서는 무난하게 전체설치(Full)를 선택한다. 이후 계속 Next 버튼을 누르면서 설치를 진행한다.



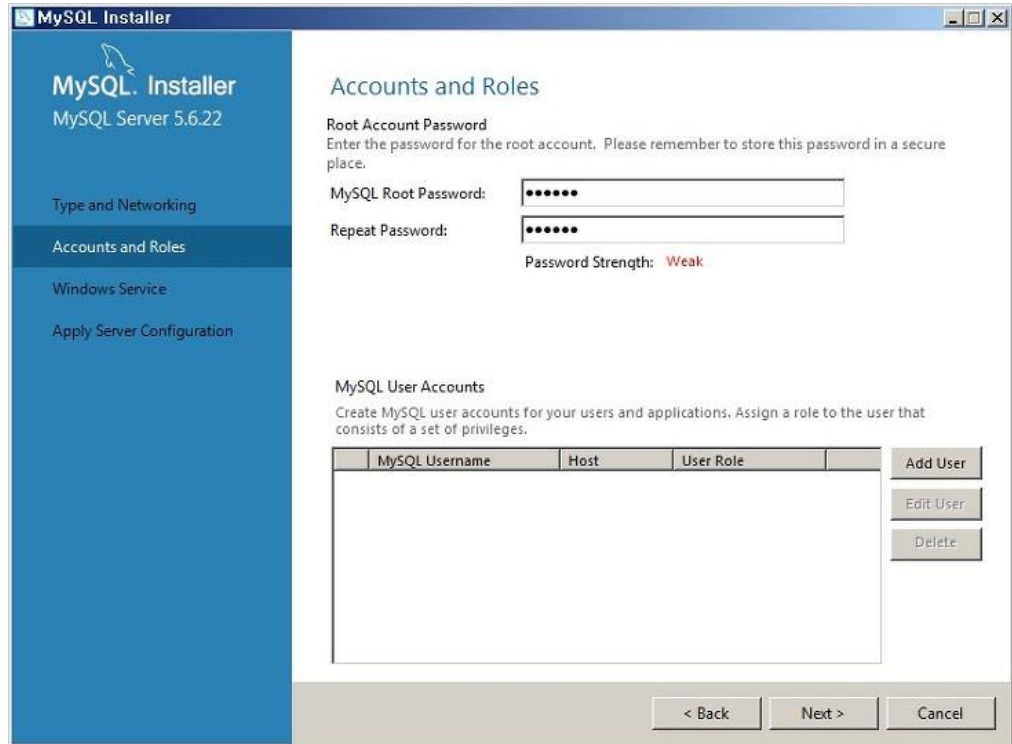
설치가 완료되었으면 Next



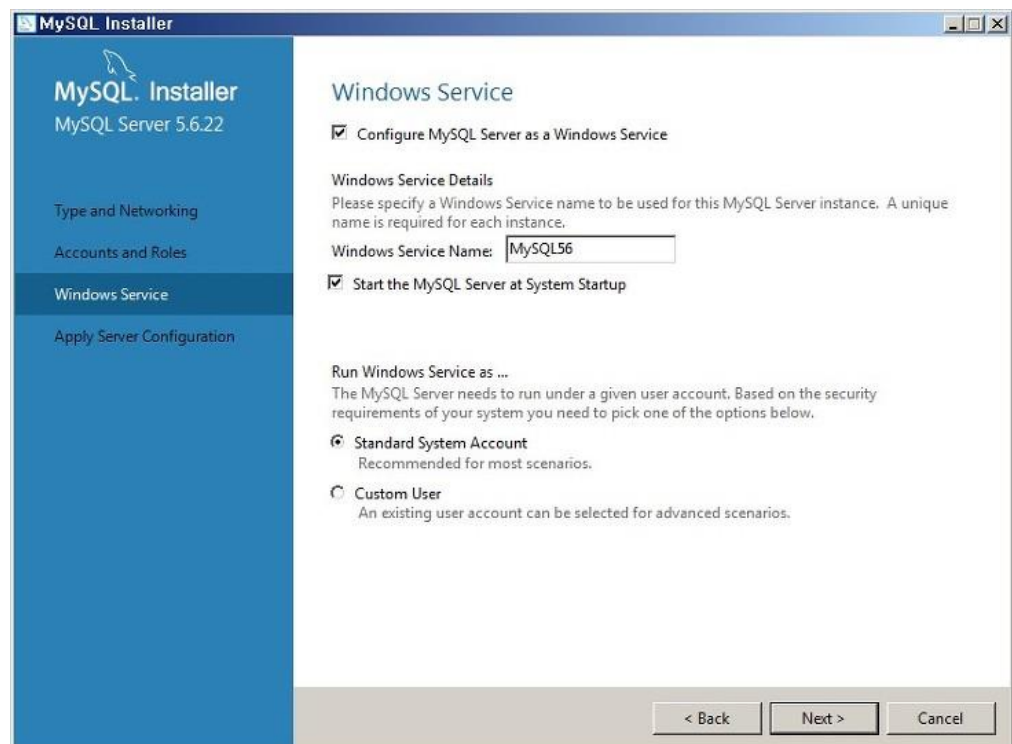
MySQL 환경설정 작업을 위해 Next



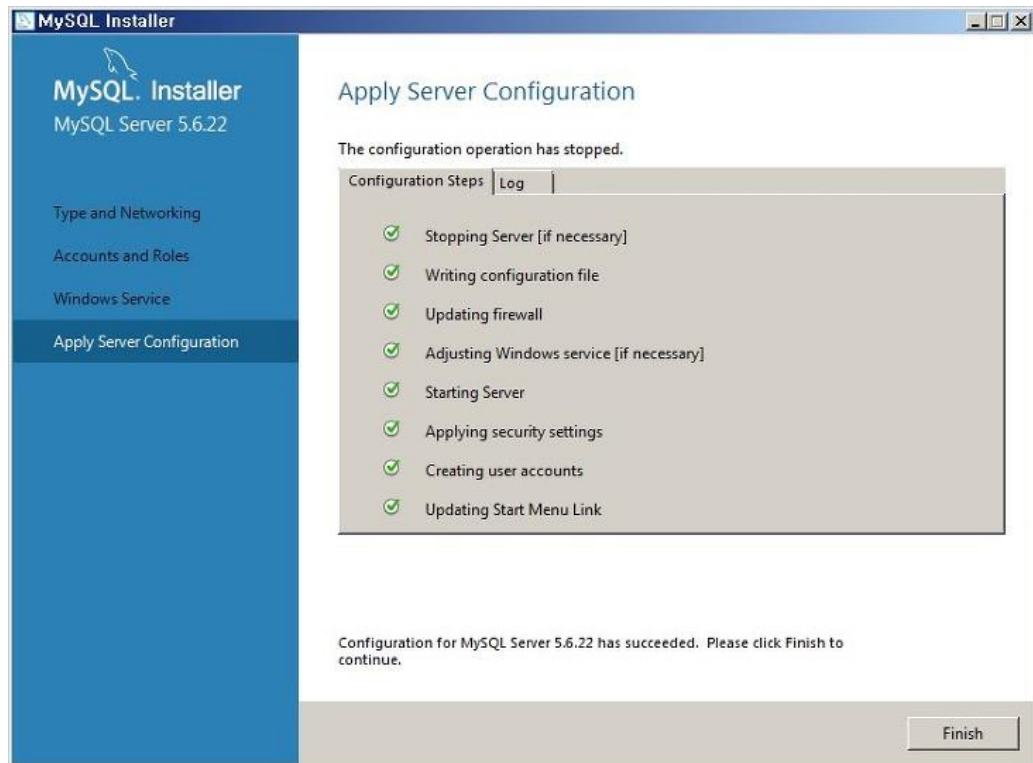
환경설정 타입에는 Development Machine, Server Machine, Dedicated Machine이 있는데 필요한 것을 선택하면 되고 여기서는 개발자라고 가정하고 Development Machine을 선택한다. Next



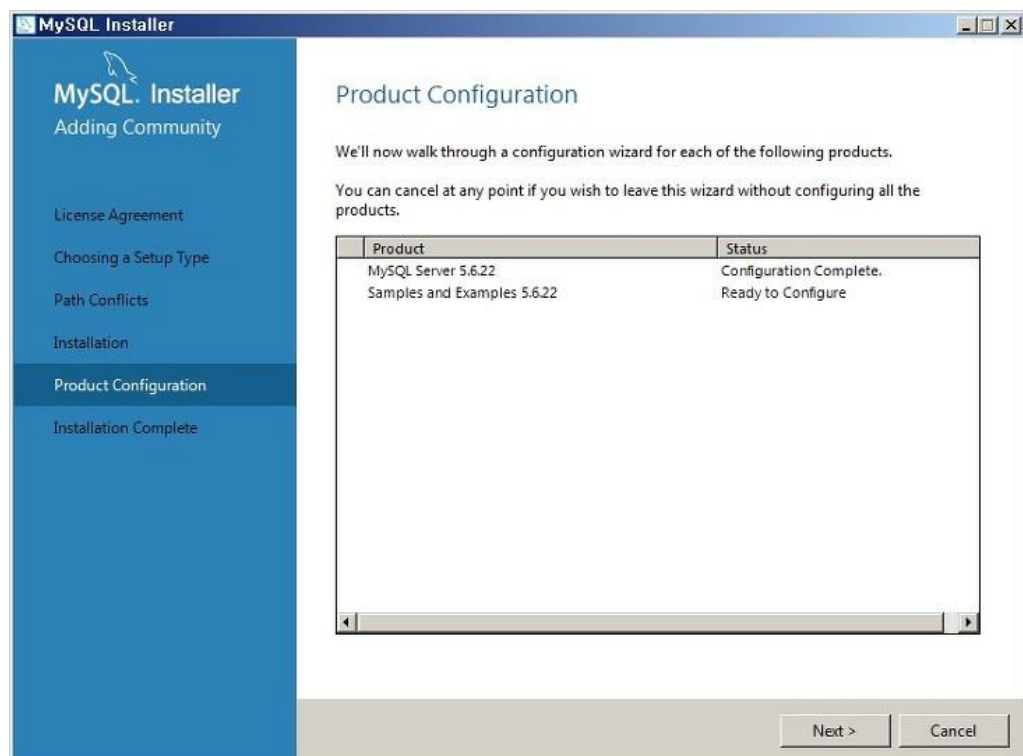
루트암호(Root Password)를 지정한다.



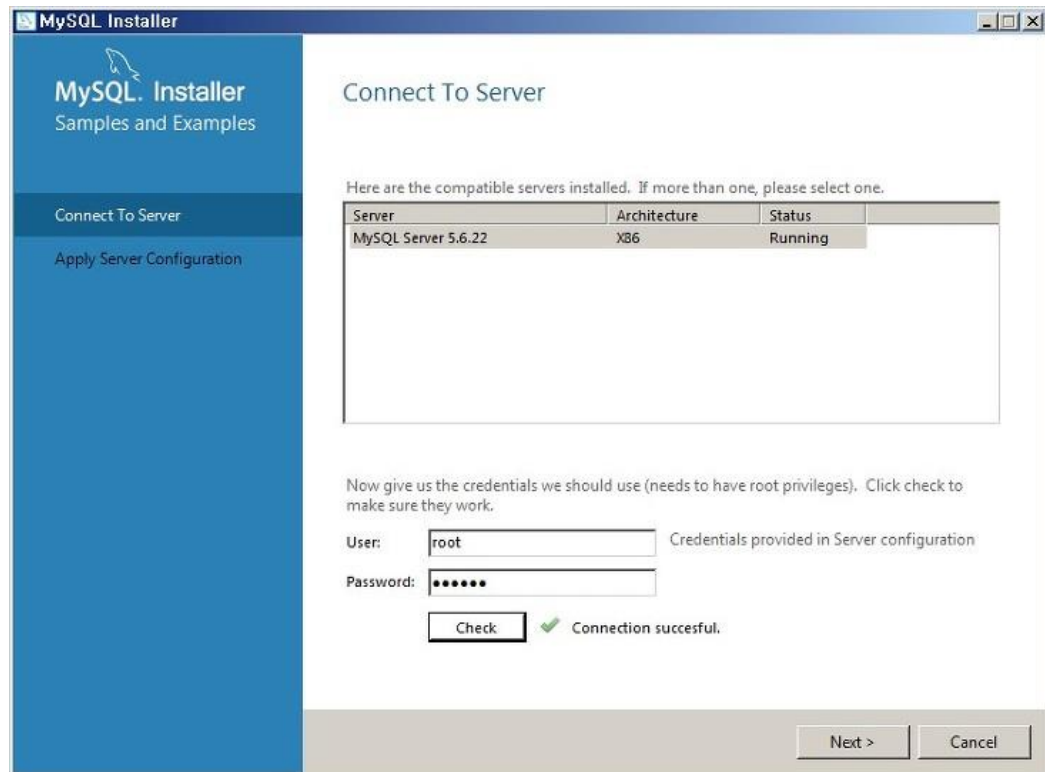
윈도우 부팅시 MySQL 자동 실행 옵션을 설정 할 수 있는 화면으로 윈도우 서비스 이름 지정한다. Next



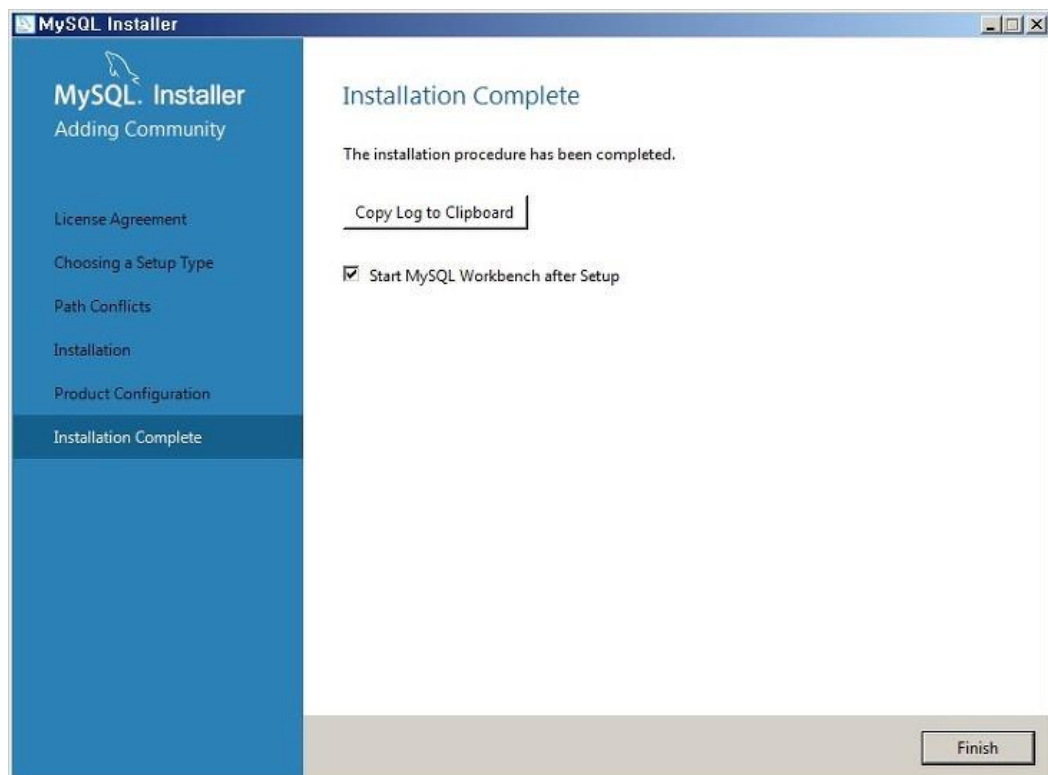
Execute를 누르고 MySQL Server 환경설정이 제대로 설정된 것을 확인하고 Finish를 누른다.



기타 환경설정을 위해 Next.



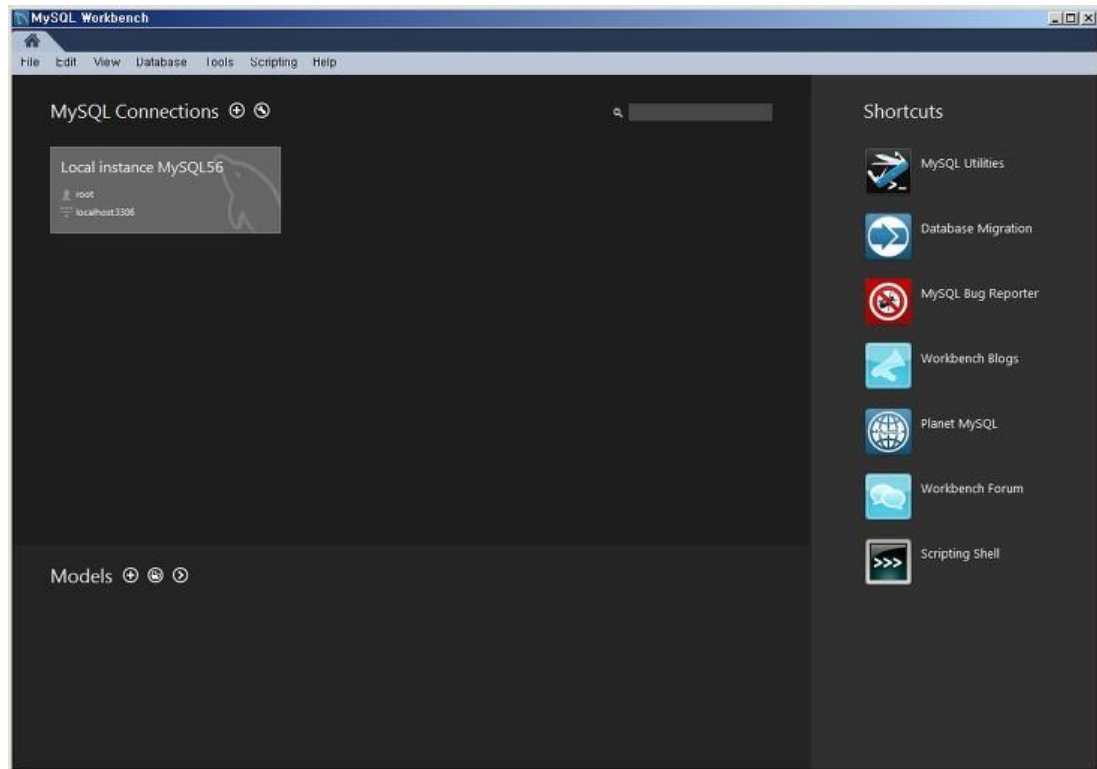
루트 ID와 암호를 입력하고 체크 후 계속 Next, Execute, Finish 버튼을 누른다.



MySQL 설치가 완료되었다. 마지막 Finish 버튼을 누른다.

나. MySQL Workbench 실행

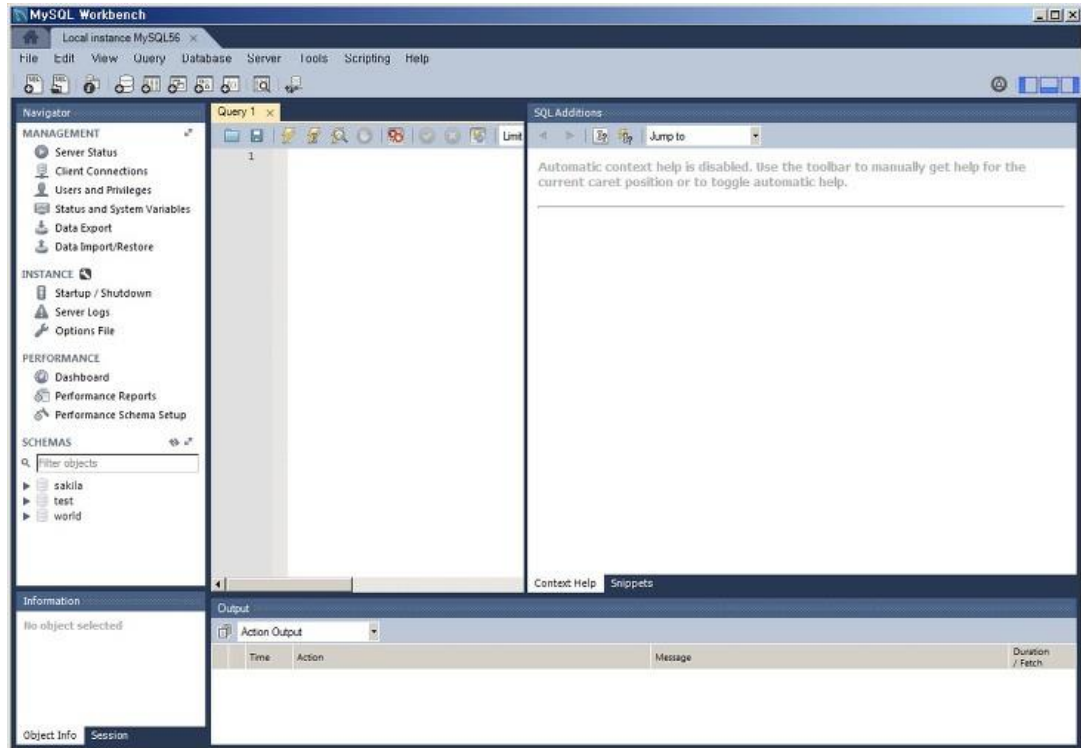
위에서 Full 옵션으로 MySQL을 설치했으므로 MySQL Workbench가 설치가 되어있을 것이다.



MySQL Workbench를 실행하고 Connections를 선택한다.



설치할 때 설정한 암호를 입력하면 사용할 수 있다.



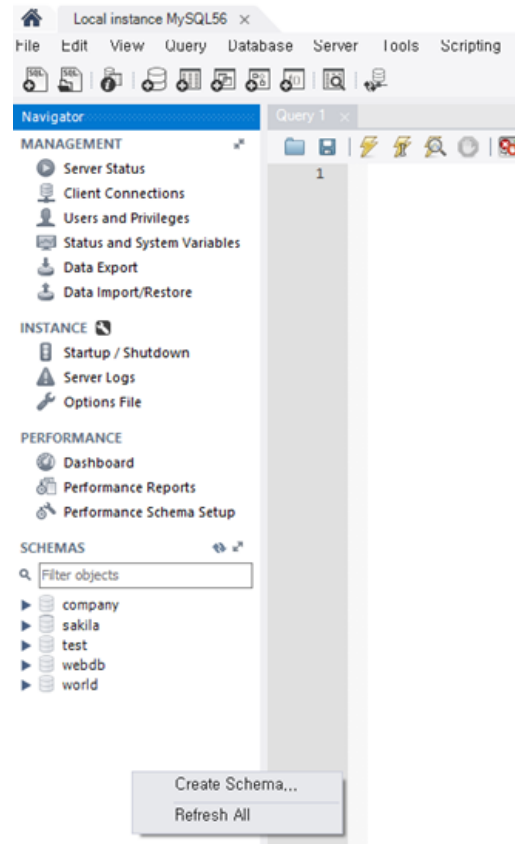
Workbench는 GUI환경에서 MySQL을 다룰 수 있는 프로그램으로 SQL 쿼리를 날릴 수 있는 것은 물론 데이터베이스, 테이블 생성, Import, Export 등 다양한 MySQL 관리 작업을 할 수 있다.

MySQL Command Line에서도 이와 같은 작업을 할 수 있지만 여기서는 Workbench를 기준으로 설명한다.

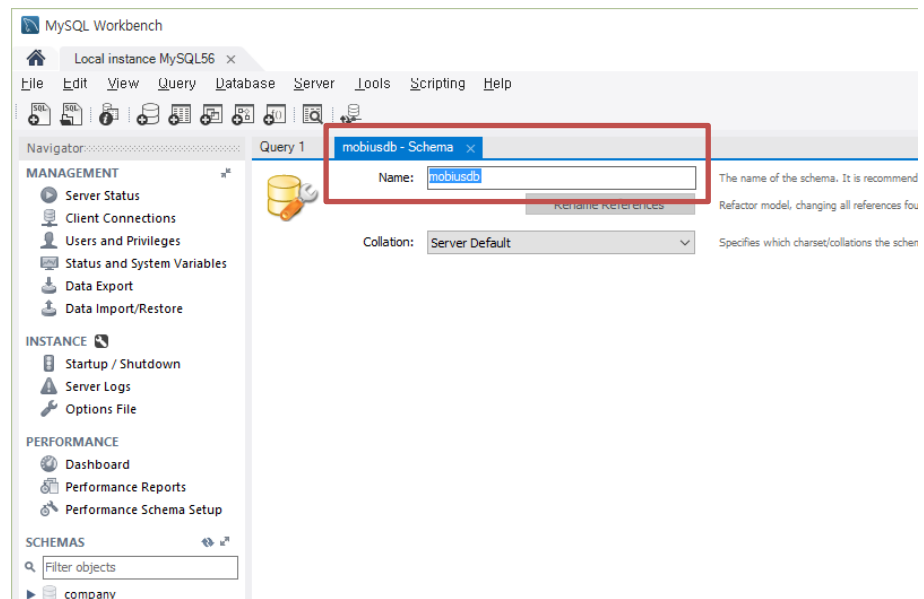
다. 환경 구축 및 데이터베이스 생성

이제 MySQL 설치가 완료되었으니 Mobius Yellow Turtle 을 위한 데이터베이스를 생성하고 테이블을 Import 하는 작업으로 MySQL 설치작업을 마무리 한다.

- 데이터베이스 생성

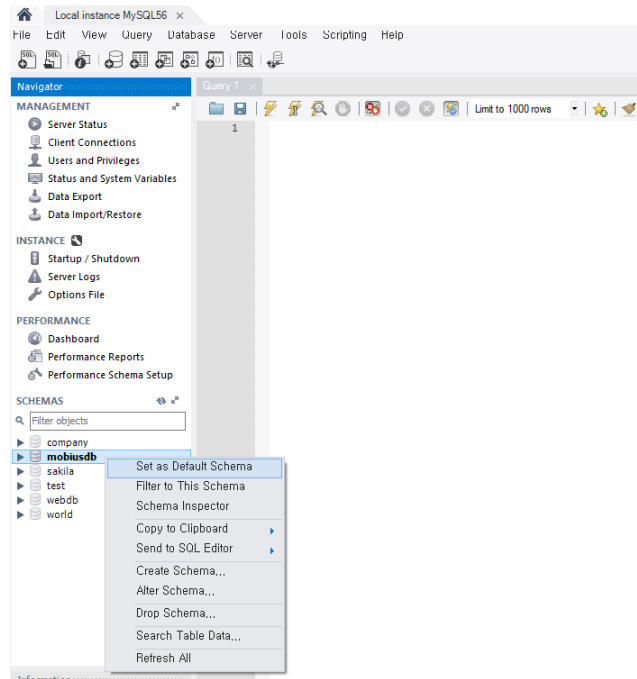


위의 그림처럼 SCHEMAS 빈 곳에서 마우스 오른쪽 버튼을 클릭하면 뜨는 메뉴에서 "Create Schema..." 선택한다.



스키마 생성 창이 뜨면 데이터베이스 이름을 "mobiusdb" 입력하고 Apply 버튼을 누르면

데이터베이스가 생성된다. mobiusdb 라는 데이터베이스(Schema) 이름은 Mobius Yellow Tutrle에서 사용하는 데이터베이스 명으로 지정되어 있기 때문에 같은 이름으로 만들어야 정상적으로 동작할 수 있다. 만약 이 이름을 바꾸고자 한다면 소스에서 데이터베이스 연결하는 부분에서 이름을 수정하면 된다.



“mobiusdb” 데이터가 생성이 되면 위의 그림과 같이 왼쪽 창에 데이터베이스가 생성됨을 확인할 수 있고 마우스 오른쪽 버튼을 눌러 Set as Default Schema로 설정을 해 준다.

- Mobiusdb로 테이블 가져오기

mobiusdb로 테이블을 가져오기 위해 Workbench의 Server 메뉴에서 Data Import 를 선택한다. Import할 sql 파일은 다음과 같이 OCEAN 사이트에서 다운로드 받을 수 있다.

OCEAN

About License Download Community Showcase Contact

Logout | MyPage

Download

Open alliance for IoT standard

Download

- Yellow Turtle

Yellow Turtle v1.0 2015-09-03 16:16

Mobius: Yellow Turtle

Yellow Turtle is an open source software of oneM2M-based IoT Server Platform based on Node.js JavaScript.

The source code and files of Yellow Turtle are under the OCEAN license terms and conditions, i.e., 3-clause BSD open source license.

Versions

Code Name	Framework	Version	Ref. Standards
Yellow Turtle	Node.js JavaScript	1.0	oneM2M Release 1

PREREQUISITES

-

SYSTEM REQUIREMENTS

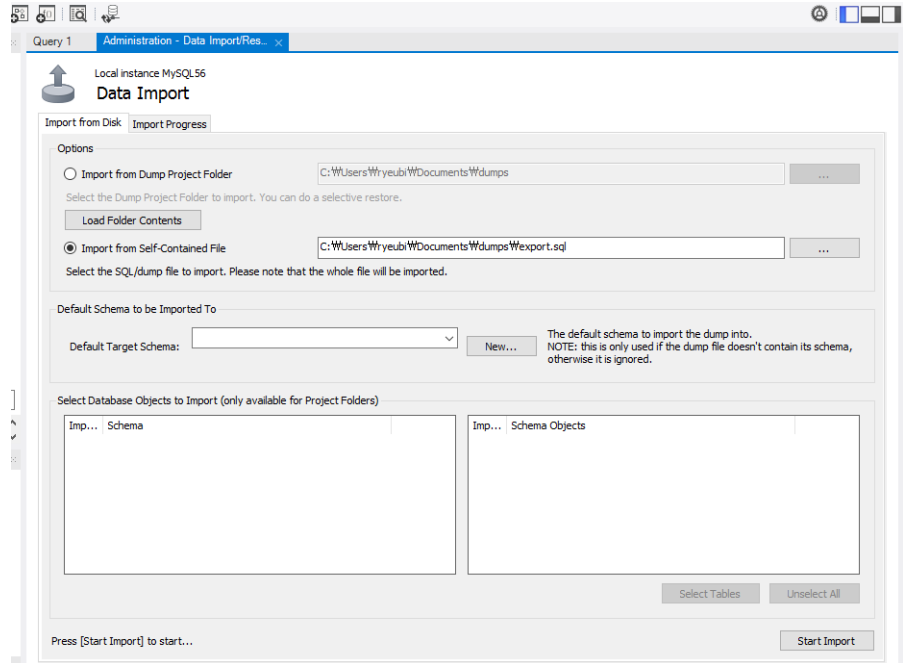
System Requirements	Remarks
Operating System	WindowsX, Linux Redhat and CentOS, Mac, Raspbian
Open Source Framework	Node.js
Web Application Server	Node.js
Database	MySQL
CoAP Framework	
MQTT Broker	Mosquitto 1.4.x

Files

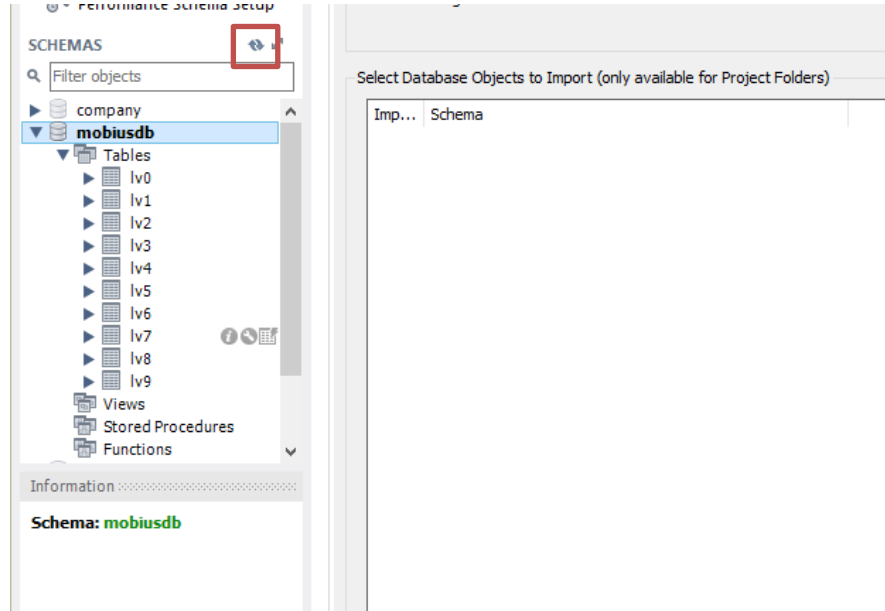
Name	Download Link
Mobius Installation Guide Korea	Installation Guide_Yellow_Turtle_v1.0_KR.pdf
Mobius Source 1.0	Mobius_Yellow_Turtle_v1.0.zip
Mysql Script	YellowTurtle_script.sql

File : Installation Guide_Mobius_Yellow_Turtle_v1.0_KR.pdf | Mobius Yellow Turtle.zip | YellowTurtle_script.sql

LAST PAGE



Data Import 창에서 Import from Self-Contained File을 선택하고 다운로드 받은 sql 파일을 선택하고 Start Import를 누르면 다음 그림과 같이 Mobius Yellow Turtle에서 사용되는 테이블이 생성된다.



Data Import를 수행한 후에는 테이블이 보이지 않을 텐데, 위에 있는 새로고침 아이콘을 클릭해야 Import 된 테이블들을 볼 수 있다.

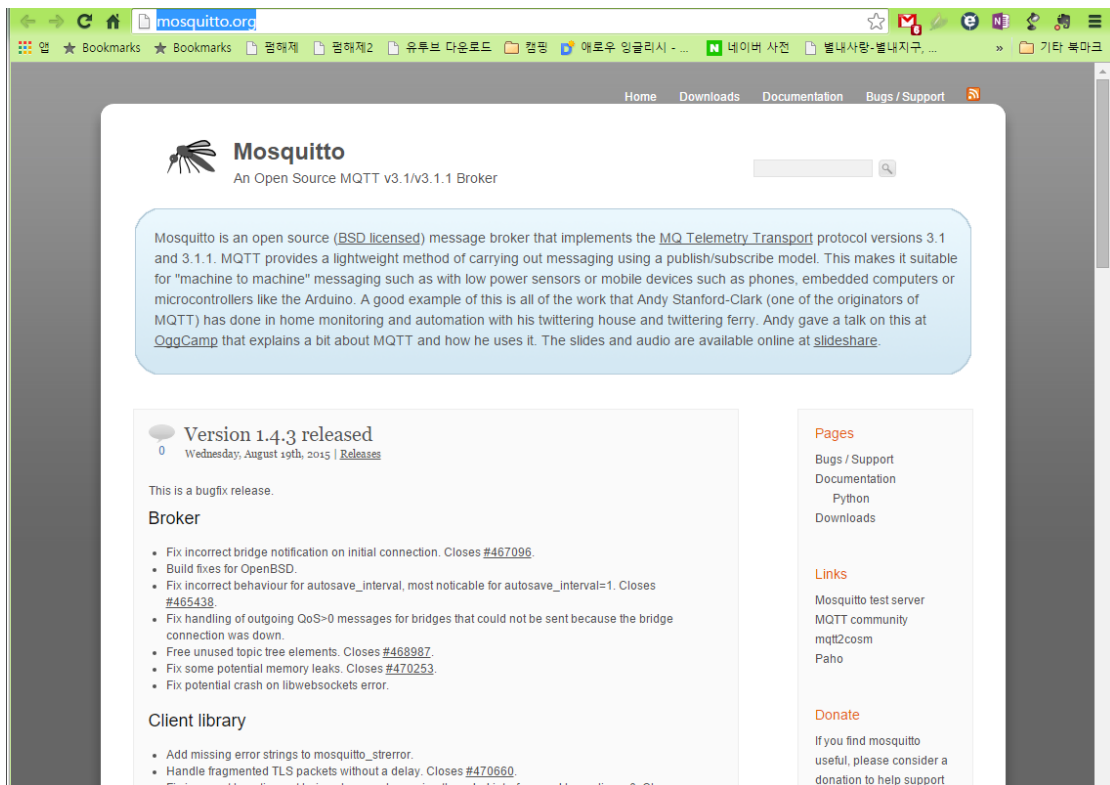
2.2.2. MQTT 서버 설치

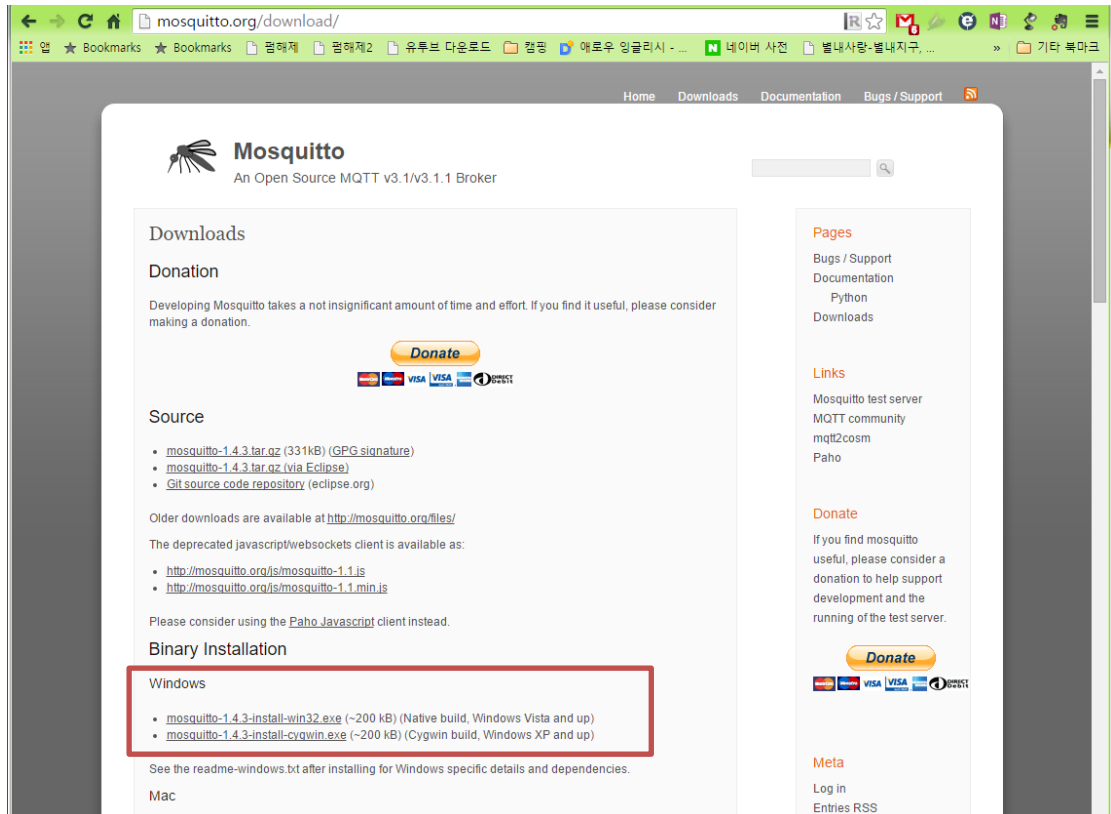
Mobius Yellow Turtle Platform은 디바이스와의 통신을 위해 MQTT 프로토콜을 지원하고 있으며 이를 위해서 오픈소스로 제공되는 MQTT broker Mosquitto 서버를 설치한다.

<http://mosquitto.org>

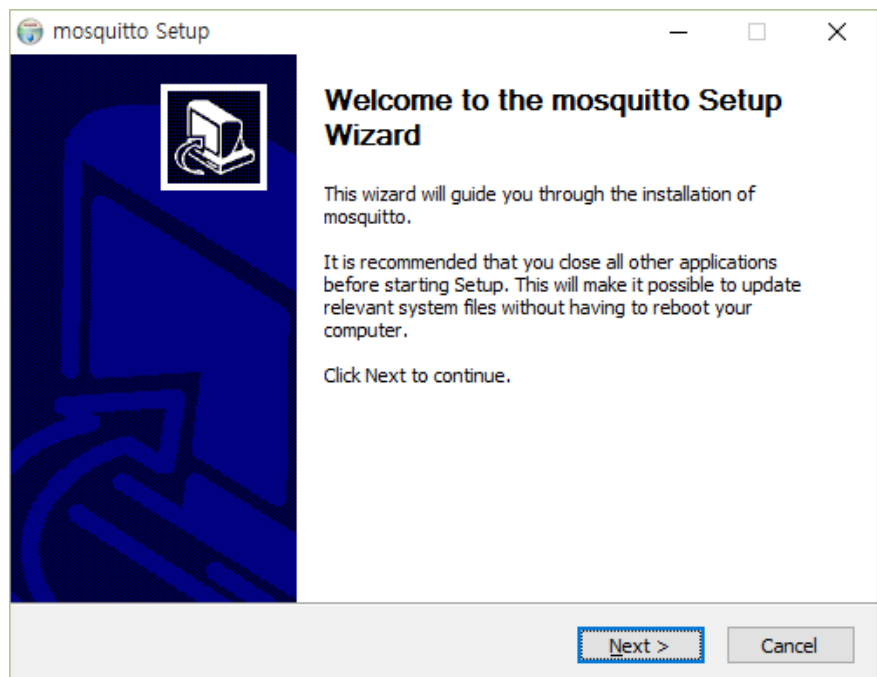
위의 사이트에서 다운로드 받을 수 있다. MQTT 서버는 만약 MQTT를 지원하지 않는다면 설치를 하지 않아도 무방하다.

사이트에서 다운로드 메뉴를 누르고 Binary Installation 의 Windows 설치파일 (mosquitto-1.4.3-install-win32.exe)을 선택하여 mosquito 설치파일을 다운로드 한다.

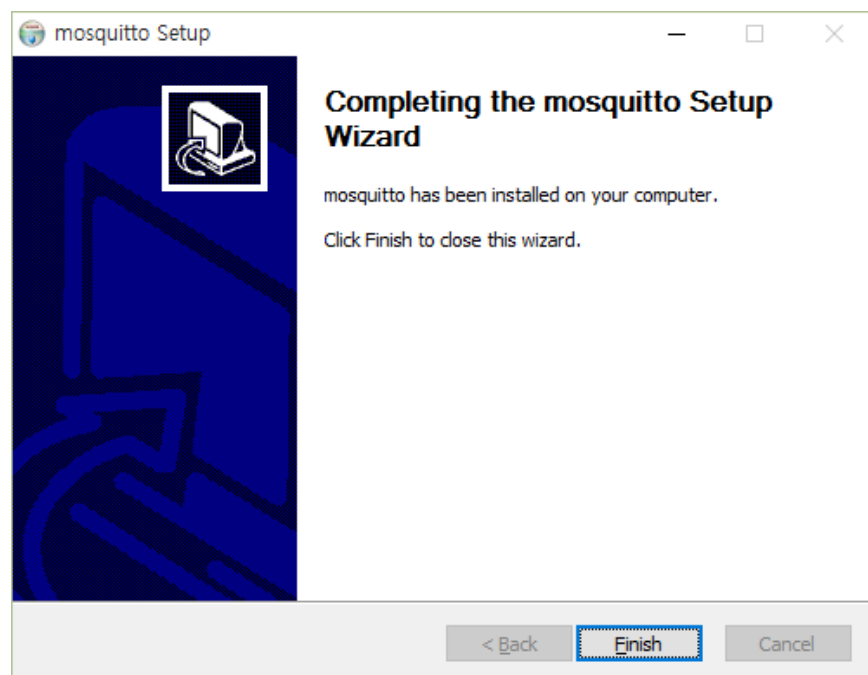
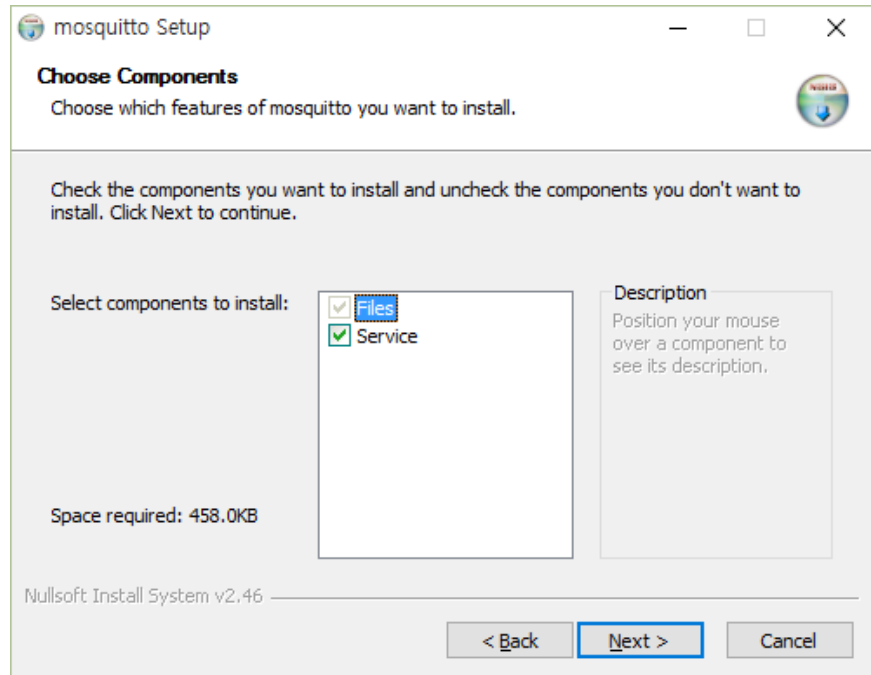




다운로드한 설치파일을 실행하면 다음과 같이 설치가 진행된다.



설치 중에 아래의 그림처럼 윈도우 서비스로 mosquitto 서버를 시작하는 옵션이 나오는데 아래와 같이 디폴트로 두고 Next를 누른다. 설치 후 별도의 작업 없이 윈도우 서비스로 등록되어 mosquitto 서버가 자동으로 시작되는 옵션이다.

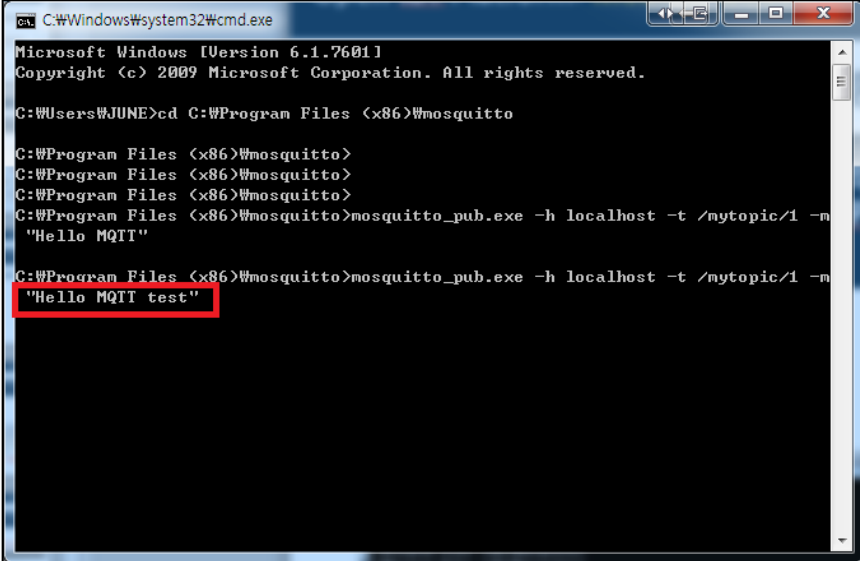


설치가 완료되면 정상적으로 설치가 되었겠지만 아래와 같이 MQTT broker mosquitto 서버의 정상 동작을 확인할 수 있다.

명령 프롬프트를 실행시키고 아래와 같이 mosquitto 서버가 설치된 폴더로 이동하여 다음의 명령어를 실행한다.

```
mosquitto_sub.exe -h localhost -t /mytopic/1
```

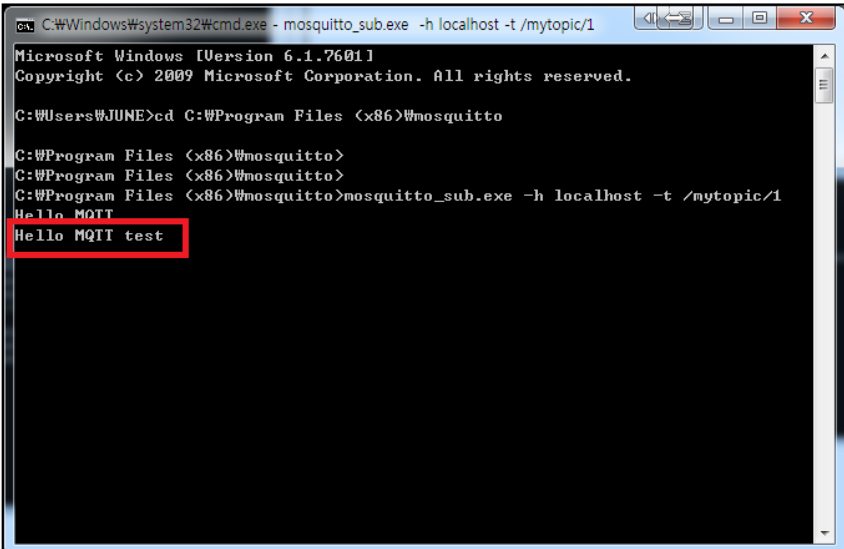
mosquitto_pub.exe -h localhost -t /mytopic/1 -m "Hello MQTT test"



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\JUNE>cd C:\Program Files (x86)\mosquitto

C:\Program Files (x86)\mosquitto>
C:\Program Files (x86)\mosquitto>
C:\Program Files (x86)\mosquitto>
C:\Program Files (x86)\mosquitto>mosquitto_pub.exe -h localhost -t /mytopic/1 -m
"Hello MQTT"
C:\Program Files (x86)\mosquitto>mosquitto_pub.exe -h localhost -t /mytopic/1 -m
"Hello MQTT test"
```



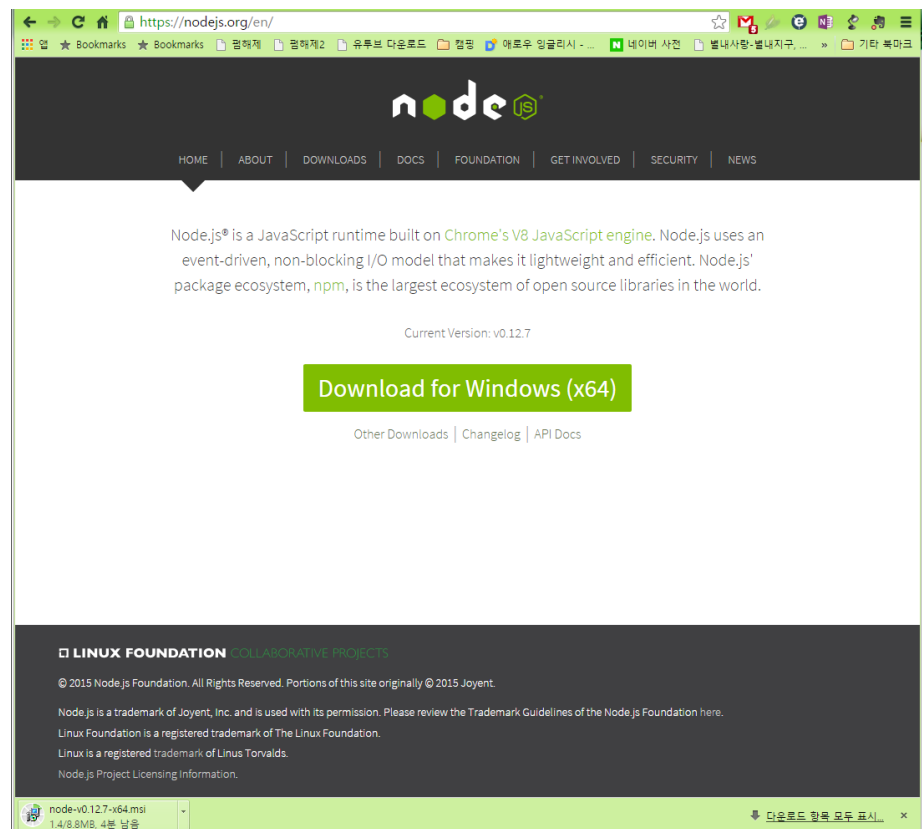
```
C:\Windows\system32\cmd.exe - mosquitto_sub.exe -h localhost -t /mytopic/1
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\JUNE>cd C:\Program Files (x86)\mosquitto

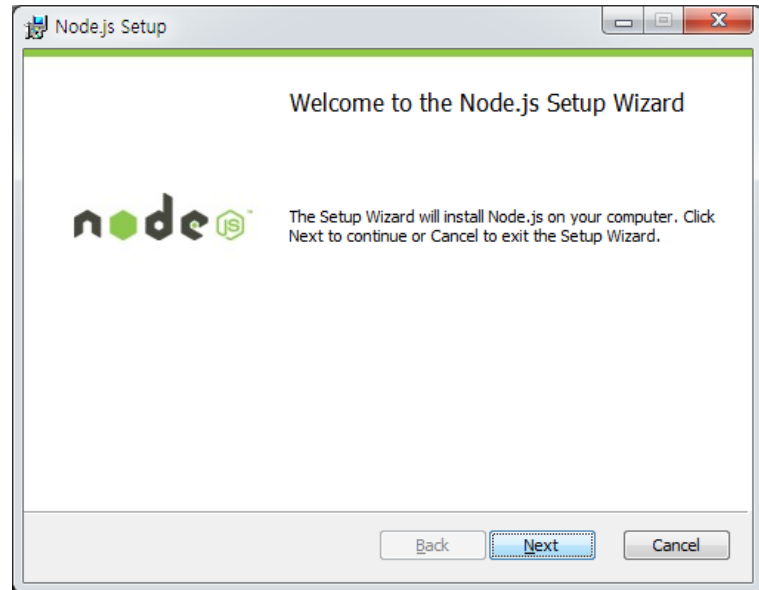
C:\Program Files (x86)\mosquitto>
C:\Program Files (x86)\mosquitto>
C:\Program Files (x86)\mosquitto>mosquitto_sub.exe -h localhost -t /mytopic/1
Hello MQTT
Hello MQTT test
```

2.2.3. Node JS 설치

node.js는 고성능의 비동기 IO (Async/Non-blocking IO)를 지원하는 single thread 기반 네트워크 서버이다. 2009년 Ryan Dahl에 의해 개발이 시작되었고 현재 수많은 지원 모듈을 가지고 있는 오픈 소스 프로젝트 중 하나이다. Node.js는 Google Chrome V8 엔진으로 개발되어 있고 Event 기반의 프로그래밍 모델로써 프로그래밍 언어로는 Java script를 사용한다. 근래에 들어서 많은 인터넷 기업들이 node.js를 도입하고 있다.

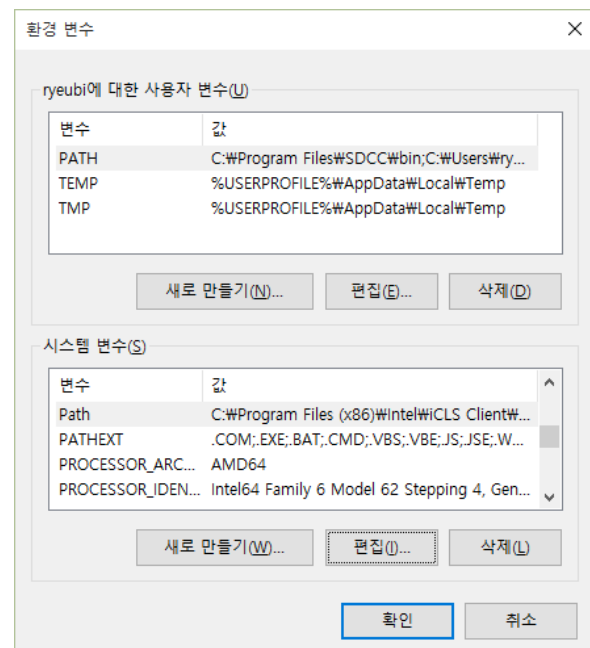


<http://www.nodejs.org> 페이지에서 Install 버튼을 누르면 OS에 맞는 인스톨러를 다운로드해 준다.

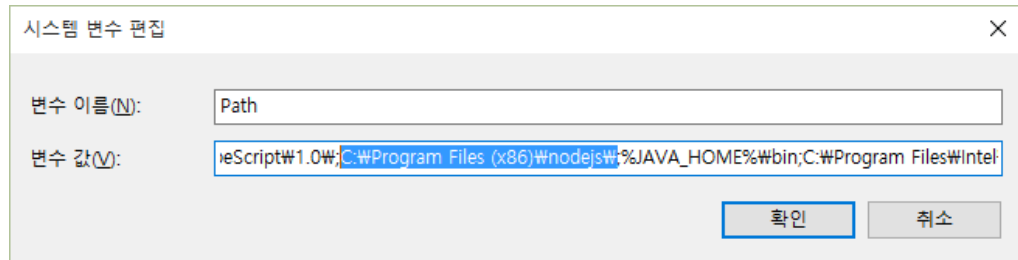


다음으로 위와 같이 다운로드한 인스톨러를 실행한다.

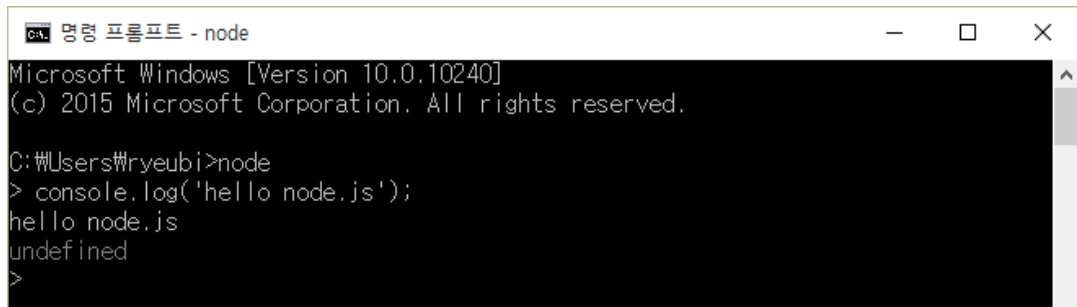
디폴트로 Next 버튼을 누르면 설치가 완료되고, 설치가 되었으면 설치된 디렉토리를 PATH에 추가한다.



Path를 선택하고 아래와 같이 C:\Program Files (x86)\nodejs\ 를 추가하고 확인 버튼을 누른다.



노드가 제대로 설치가 되었는지 확인한다. Node.js는 대화형 cli를 제공한다.



위와 같이 실행이 되면 node.js가 제대로 설치가 된 것이다.

2.2.4. Mobius Yellow Turtle 서버 설치

Mobius Yellow Turtle Platform은 OCEAN Alliance 가입사를 대상으로 Mobius 서버 플랫폼 바이너리 및 소스를 공개하고 있다. Mobius Yellow Turtle 서버 플랫폼 압축 파일을 다운로드 받은 후, 제공된 매뉴얼에 따라 Port 정보 등의 Configuration 정보를 설정한다.

OCEAN Alliance 사이트(<http://www.iotocean.org>)에 접속하여 아래와 Mobius Yellow Turtle 서버 파일 다운로드한다.

Download - Yellow Turtle

Yellow Turtle v1.0 2015-09-03 16:16

Mobius: Yellow Turtle

Yellow Turtle is an open source software of oneM2M-based IoT Server Platform based on Node.js JavaScript.

The source code and files of Yellow Turtle are under the OCEAN license terms and conditions, i.e., 3-clause BSD open source license.

Versions

Code Name	Framework	Version	Ref. Standards
Yellow Turtle	Node.js JavaScript	1.0	oneM2M Release 1

PREREQUISITES

SYSTEM REQUIREMENTS

System Requirements	Remarks
Operating System	WindowsX, Linux Redhat and CentOS, Mac, Rasbian
Open Source Framework	Node.js
Web Application Server	Node.js
Database	MySQL
CoAP Framework	
MQTT Broker	Mosquitto 1.4.x

Files

Name	Download Link
Mobius Installation Guide Korea	Installation Guide_Yellow_Turtle_v1.0_KR.pdf
Mobius Source 1.0	Mobius_Yellow_Turtle_v1.0.zip
MySQL Script	YellowTurtle_script.sql

File : [Installation Guide_Mobius_Yellow_Turtle_v1.0_KR.pdf](#) | [Mobius Yellow Turtle.zip](#) | [YellowTurtle_script.sql](#)

LAST PAGE

다운로드한 파일을 적당한 곳에서 압축을 풀면 다음과 같이 node.js 버전의 Mobius 파일들을 볼 수 있다. Node.js는 별도의 컴파일 과정이 필요하지 않으므로 바로 압축을 풀 후 실행할 수 있다. 이전의 MySQL과 Node.js가 제대로 설치가 되어 있다면 Mobius Yellow Turtle 에 사용된 node.js 모듈을 설치하면 별도의 작업 없이 실행이 가능하다.

이름	수정한 날짜	유형	크기
mobius	2015-10-09 오후...	파일 폴더	
app.js	2015-10-02 오후...	JetBrains WebStor...	36KB
conf.json	2015-09-25 오전...	JSON File	1KB
package.json	2015-09-03 오전...	JSON File	1KB
pxymqtt.js	2015-09-15 오전...	JetBrains WebStor...	14KB

app.js가 Mobius의 대문역할과 라우팅 역할을 수행하는 메인 코드이다. pxymqtt는 MQTT proxy를 구현한 코드이고 mobius 폴더에 있는 파일들이 실제 패킷을 처리하고 DB 액세스와 응답을 수행하는 코드들이다.

현재는 추가적인 node.js 모듈들이 설치가 되어있지 않아 애러가 날 것이다. 다음과 같이 명령 프롬프트를 실행시키고 npm 툴을 이용해 필요한 모듈을 설치해야 한다.

package.json 파일에 필요한 모듈 리스트를 가지고 있다.

```

C:\Users\ryeubi\Dropbox\Downloads\mobius yellow turtle>
C:\Users\ryeubi\Dropbox\Downloads\mobius yellow turtle>
C:\Users\ryeubi\Dropbox\Downloads\mobius yellow turtle>npm install

```

위와 같이 npm install 을 입력하고 엔터를 치면 아래와 같이 필요한 모듈들이 설치가 된다.

```

finalhandler@0.4.0 (unpipe@1.0.0)
proxy-addr@1.0.8 (forwarded@0.1.0, ipaddr.js@1.0.1)
send@0.13.0 (destroy@1.0.3, statuses@1.2.1, ms@0.7.1, http-errors@1.3.1,
mime@1.3.4)
type-is@1.6.7 (media-typer@0.3.0, mime-types@2.1.5)
accepts@1.2.12 (negotiator@0.5.3, mime-types@2.1.5)

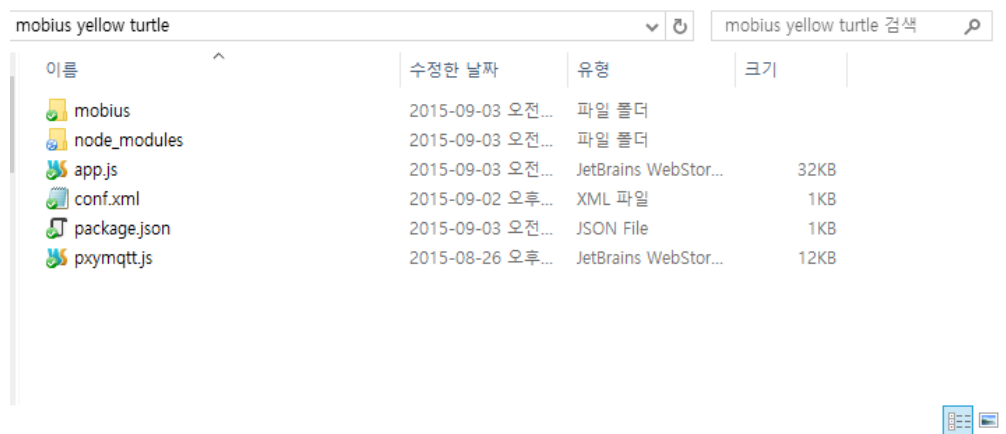
xmlbuilder@2.6.4 node_modules\xmlbuilder
  lodash@3.10.1

mqtt@1.4.0 node_modules\mqtt
  inherits@2.0.1
  xtend@4.0.0
  minimist@1.2.0
  readable-stream@1.0.33 (isarray@0.0.1, string_decoder@0.10.31, core-util-
is@1.0.1)
  commist@1.0.0 (leven@1.0.2)
  mqtt-packet@3.3.1 (bl@0.9.4)
  end-of-stream@1.1.0 (once@1.3.2)
  help-me@0.1.0 (pump@1.0.0)
  concat-stream@1.5.0 (typedarray@0.0.6, readable-stream@2.0.2)
  mqtt-connection@2.1.1 (through2@0.6.5, reduplexer@1.1.0)
  websocket-stream@2.0.2 (through2@0.6.5, duplexify@3.4.2, ws@0.8.0)

C:\Users\ryeubi\Dropbox\Downloads\mobius yellow turtle>

```

이렇게 설치가 되면 다음 그림과 같이 node_modules 폴더가 생긴 것을 볼 수 있다.



이제 모듈 설치가 끝났으니 다음과 같이 node app.js를 실행하면 Mobius Yellow Turtle 서버 플랫폼 설치가 완료된다.

```

명령 프롬프트
C:\Users\ryeubi\Dropbox\Downloads\mobius yellow turtle>node app.js
finalhandler@0.4.0 (unpipe@1.0.0)
proxy-addr@1.0.8 (forwarded@0.1.0, ipaddr.js@1.0.1)
send@0.13.0 (destroy@1.0.3, statuses@1.2.1, ms@0.7.1, http-errors@1.3.1,
mime@1.3.4)
type-is@1.6.7 (media-typer@0.3.0, mime-types@2.1.5)
accepts@1.2.12 (negotiator@0.5.3, mime-types@2.1.5)

xmlbuilder@2.6.4 node_modules\xmlbuilder
  lodash@3.10.1

matt@1.4.0 node_modules\matt
  inherits@2.0.1
  xtend@4.0.0
  minimist@1.2.0
  readable-stream@1.0.33 (isarray@0.0.1, string_decoder@0.10.31, core-util-
is@1.0.1)
  commist@1.0.0 (leven@1.0.2)
  matt-packet@3.3.1 (bl@0.9.4)
  end-of-stream@1.1.0 (once@1.3.2)
  help-me@0.1.0 (pump@1.0.0)
  concat-stream@1.5.0 (typedarray@0.0.6, readable-stream@2.0.2)
  matt-connection@2.1.1 (through2@0.6.5, reduplexer@1.1.0)
  websocket-stream@2.0.2 (through2@0.6.5, duplexify@3.4.2, ws@0.8.0)

C:\Users\ryeubi\Dropbox\Downloads\mobius yellow turtle>node app.js

```

아래와 같이 메시지가 나오면 정상적으로 Mobius Yellow Turtle 서버 플랫폼이 구동된 것이다.

```

C:\Users\ryeubi\Dropbox\Downloads\mobius yellow turtle>node app.js
server running at 7579 port
[IC create] - 2015-09-03T01:29:50+09:00

```

가. 환경설정

Mobius Yellow Turtle의 환경 설정은 conf.json 파일을 통해 설정 변경을 할 수 있다. 폴더 내에 conf.json 파일에 보면 다음과 같다.

```
{
  "m2m:conf": {
    "cbtype": "in",
    "nmtype": "short",
    "bodytype": "json",
    "in-cse": {
      "cseid": "/0.2.481.1.1.1.1",
      "csebase": "mobius-yt",
      "cseport": "7579",
      "dbhost": "localhost",
      "dbpass": "dksdlfduq2",
      "mqttproxy": "localhost",
      "mqttproxyport": "9726"
    },
    "mn-cse": {
      "cbhost": "203.253.128.150",
      "cbport": "9000",
      "cbname": "mobius",
      "cbcseid": "mobius"
    }
  }
}
```

반드시 맞춰줘야 하는 작업으로 <dbpass> 태그의 값은 MySQL 설치 시 설정한 루트 암호를 입력해야 한다.

<cseport> 태그의 값을 변경하면 서버의 포트 번호를 변경할 수 있다. 기본적으로 Mobius Yellow Turtle의 포트번호는 7579 로 설정되어 있다.

<csebase> 태그는 Mobius 서버 Base url 이름을 나타내고 서버의 이름을 나타낸다. 서버의 이름을 변경하고자 할 때 이 <csebase> 태그를 변경하면 서버의 이름을 변경할 수 있다.

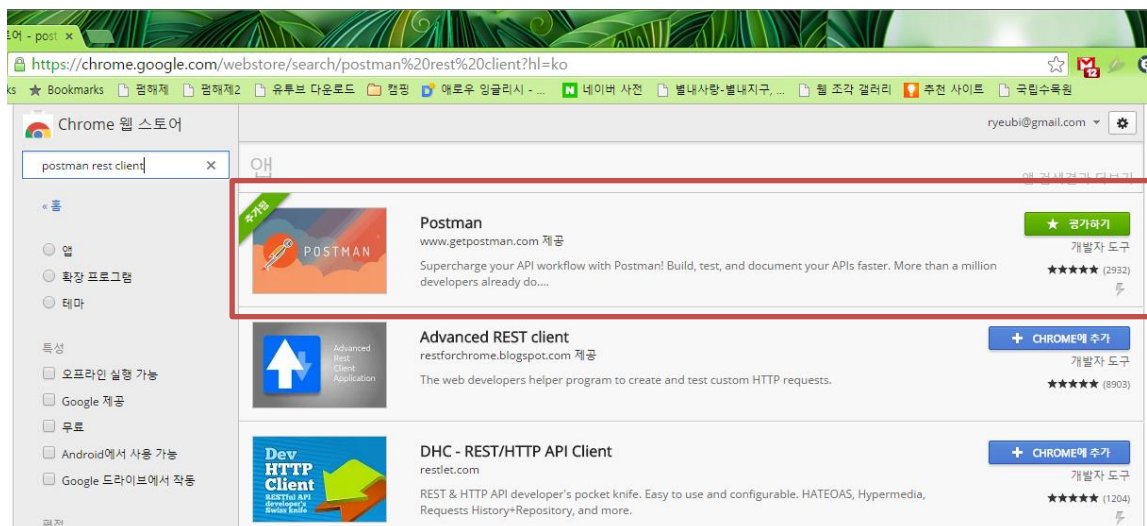
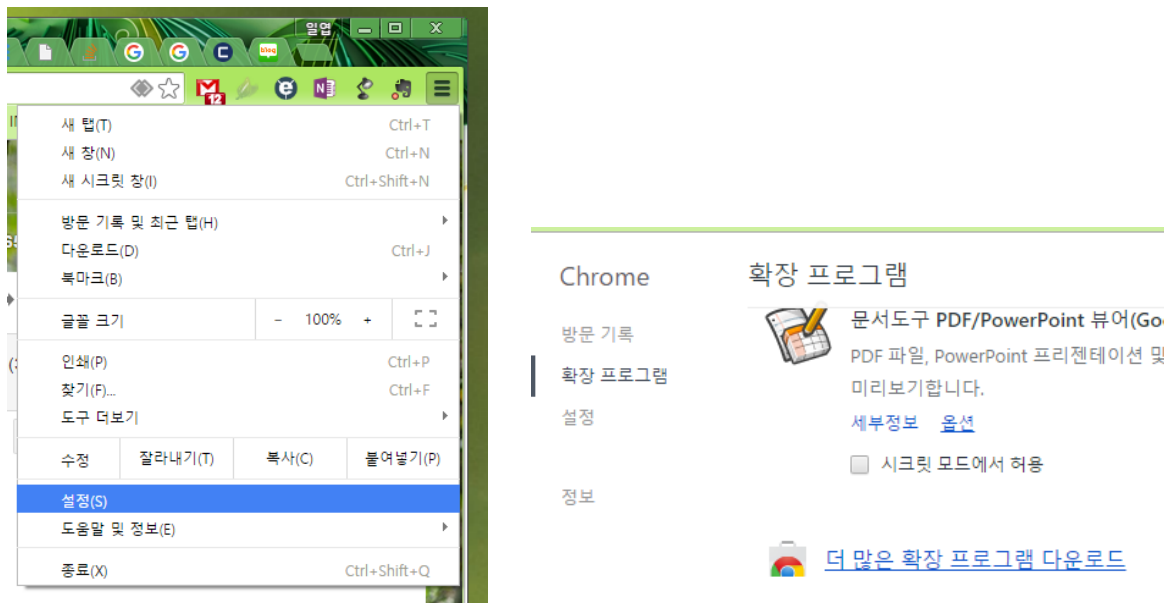
<cseid> 태그는 unique 한 id로 설정해야 하고 UUID를 활용하거나 KISA에서 발급되는 ID를 사용할 수 있다. 현재는 테스트로 위의 포맷을 그대로 두고 마지막의 숫자를 변경하여 사용할 수 있다.

<mqttproxy> 태그는 MQTT broker 주소를 적는데 현재는 로컬에 설치하였으므로 localhost라고 설정하였으며 외부의 broker를 사용한다면 해당 ip를 적으면 된다.

나. 테스트

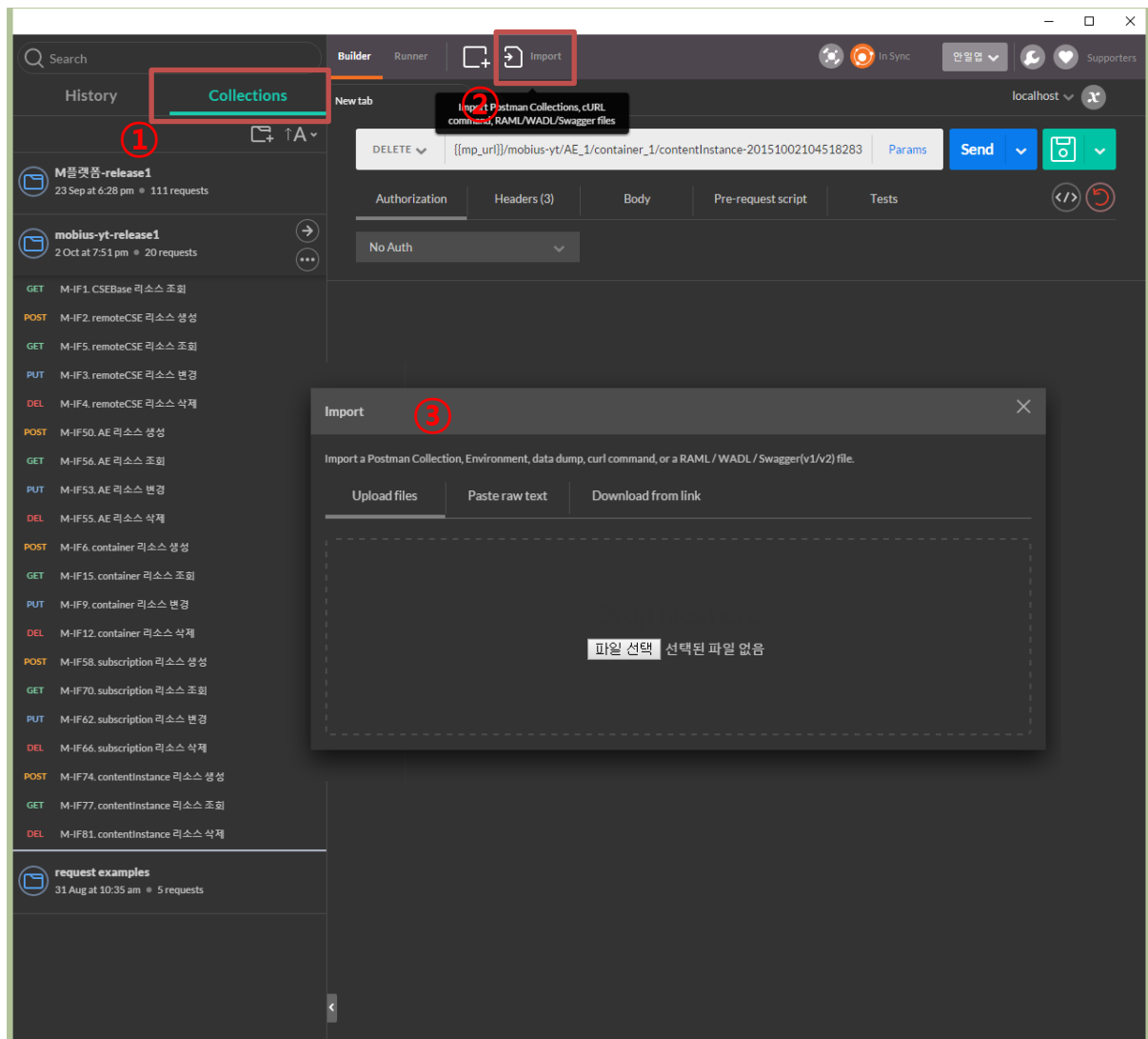
크롬 확장 프로그램인 Postman을 활용하여 직접 구축한 Mobius Yellow Turtle을 테스트 할 수 있다.

아래와 같이 크롬 브라우저에서 설정 메뉴를 선택하고 설정화면의 확장 프로그램 탭에서 “더 많은 확장 프로그램 다운로드”를 선택한다. 검색란에 “postman rest client”를 입력하고 Postman을 찾아 설치할 수 있다.



Postman을 설치하면 아래와 같이 실행이 된다. Mobius Yellow Turtle 테스트를 위해 OCEAN에서 미리 만들어둔 스크립트를 다운받아 Import 한다. Import 하는 방법은 그림과 같이 Collections 탭을 선택하고 바로 오른쪽 위에 Import 메뉴를 선택한다. Import를

위한 창이 열리면 Upload files 탭의 파일 선택을 눌러 OCEAN에서 다운로드 받은 스크립트 파일을 선택하고 import 한다.



Import가 성공하면 아래 그림처럼 왼쪽에 Mobius-yt-release1 이라는 컬렉션 탭이 생성이 될 것이다. 이 테스트 스크립트는 현재 지원되는 리소스의 CRUD를 할 수 있도록 정리를 해 둔 것이고 기존적으로 fullname type의 리소스 이름으로 설정되어 있고 xml 데이터 포맷으로 테스트 할 수 있도록 설정되어 있다.



테스트를 위해 직접 구축한 서버 정보를 입력해야 하는데 아래 그림과 같이 주소창에서 {{mp_url}} 대신에 http://(직접구축한서버주소):포트번호 를 입력한다.

