IT운영실 정보 1실

NICE신용정보㈜

2022-01-13

**개정 이력**

**Docker – talend ETL, bookstack 설치**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 개정번호 | 제∙개정 장 | 제∙개정 내역 | 작성자 | 작성일 |
| 0.1 |  | 최초 작성 | 박선우 | 2022.1.13 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**목 차**

[[1. 개요 3](#_개요)](#_Toc87532100)

[[2. 알림 생성하기 3](#_개요)](#_Toc87532101)

[[2.1. 커넥터 생성하기 3](#_개요)](#_Toc87532102)

[[2.2. 알림 생성하기 4](#_개요)](#_Toc87532103)

[[2.3. METRICS – Metric threshold 5](#_개요)](#_Toc87532104)

[[2.4. Uptime – Uptime monitor status 6](#_개요)](#_Toc87532105)

[[2.5. STACK Alerts – Elasticsearch query 7](#_개요)](#_Toc87532106)

[[3. Elastic Query DSL 7](#_개요)](#_Toc87532107)

[[4. Output 형식 지정 12](#_개요)](#_Toc87532108)

# 개요

Docker 에 대해 알아보고 centOS 가상환경에 이를 설치한 후 , 설치한 docker환경에 talend ETL,

Bookstack을 설치하는 과정을 진행할 것입니다.

## Docker란 무엇인가

도커란 리눅스 컨테이너를 기반으로 하여 특정한 서비스를 패키징하고 배포하는데 유용한 오픈소스 프로그램이다. 도커는 소프트웨어를 컨테이너라는 표준화된 유닛으로 패키징하며, 이 컨테이너에는 라이브러리, 시스템 도구, 코드, 런타임 등 소프트웨어를 실행하는데 필요한 모든 것이 포함되어 있다. 실행환경을 컨테이너로 추상화하고 동일한 인터페이스를 제공하여 환경에 구애받지 않고 앱을 신속하게 배포, 확장, 관리할 수 있게 해주는 특징이 있다.

* 도커의 컨테이너란 ?

컨테이너는 격리된 공간에서 프로세스가 동작하는 기술이다. 가상화 기술의 하나이지만 주로 운영체제를 가상화한 기존방식과는 차이가 있다. 이는 호스트 운영체제 위에 게스트 운영체제전체를 가상화하여 사용하는 방식이다. Docker의 컨테이너 기술을 사용하면 운영체제 레벨은 공유하고 앱레벨에서는 각각의 앱을 격리해서 실행할 수 있어서 하나의 서버에 여러 개의 컨테이너를 실행하면 서로 영향을 미치지 않고 독립적으로 실행할 수 있다.

* 도커의 이미지란 ?

도커에서 이미지는 컨테이너를 정의하는 읽기 전용 탬플릿이다. 컨테이너 실행에 필요한 파일과 설정값등을 포함하고 있는 것으로 상태값을 가지지 않고 변하지 않는다. 컨테이너는 이미지를 실행한 상태라고 볼 수 있고 추가되거나 변하는 값은 컨테이너에 저장된다. 같은 이미지에서 여러 개의 컨테이너를 생성할 수 있고 컨테이너의 상태가 바뀌거나 컨테이너가 삭제되더라도 이미지는 변하지 않고 그대로 남아있다.

이런 이미지들은 docker hub 에 등록하거나 docker registry저장소를 직접 만들어서 관리할 수 있다.

## Docker 장점( 왜 사용하는가)

1. 빠른 속도(시작과 종료)

* 가상환경이 커널에서 공유되기 때문에 새로운 커널을 시작할 필요가 없고, 하드웨어 초기화 등의 작업이 필요없다.

1. 높은 집적도

* 컨테이너는 커널이 직접 프로세스를 조작하여 분리된 공간을 구성하기 때문에 PC상에서 동작하는 운영체제는 하나이다. 또한 여러 개의 컨테이너를 만들어서 실행중이라고 해도 운영체제는 하나이기 때문에 가상머신에 비해 고밀도가 가능하며 컨테이너에서는 실행되는 프로세스를 위한 메모리만 필요하기 때문에 낮은 사양의 환경에서도 동작이 가능하다.

1. 앱 컨테이너 지원

* 컨테이너는 목적에 맞는 프로세스만 존재하는 환경을 만들 수 있다. 예를 들어 웹서버용 컨테이너를 만든다고 하면 Apache Httpd 프로세스만 존재하는 컨테이너를 만들 수 있다

1. 많은 소프트웨어를 빠르게 제공

* Docker를 사용하면 더 많은 소프트웨어를 제공하며 필요할때마다 격리된 서비스를 제공할 수 있다

1. 운영 표준화

* 작은 컨테이너식 앱을 사용하면 보다쉽게 배포, 문제파악, 수정을 위해 롤백할 수 있다

1. 원활한 이전

* Docker기반 앱을 로컬 개발 시스템에서 AWS의 프로덕션 배포로 원활하게 이전할 수 있다

1. 비용 절감

* Docker 컨테이너를 사용하면 각 서버에서 좀 더 쉽게 더 많은 코드를 실행하여 사용률을 높이고 비용을 절감할 수 있다.

## 빅데이터 분석 분야 주요 공개 소프트웨어

빅데이터 분석 분야 주요 공개 소프트웨어 중 Pentaho, Jaspersoft, Talend가 있는데 그중 talend 를 사용하여 테스트를 진행하였다.

## Uptime – Uptime monitor status

위 로그 타입의 내용은 하트비트의 스위치 태그를 가진 데이터 중에서 monitor.status값이 한번이라도 down이 되면 알림을 보내는 역할을 한다. 따라서 특정 태그를 가진 메트릭 비트의 알림 중에서 down일 경우 알림을 주고자 한다면 Uptime monitor status가 적절하다고 본다.

## STACK Alerts – Elasticsearch query

Elasticsearch query 타입은 해당 쿼리문을 이용해서 정해진 시간마다 검색을 실행한 뒤 쿼리에 맞는 결과값의 개수에 따라 알림을 보내는 방식으로 동작한다.

# 2.Docker 설치하기

sudo yum update

sudo yum install yum-utils device-mapper-persistent-data lvm2

sudo yum-config-manager –add -repo <https://download.docker.com/linux/centos/docker-ce.repo>

sudo yum install docker-ce

* 도커 설치 완료

sudo systemctl start docker

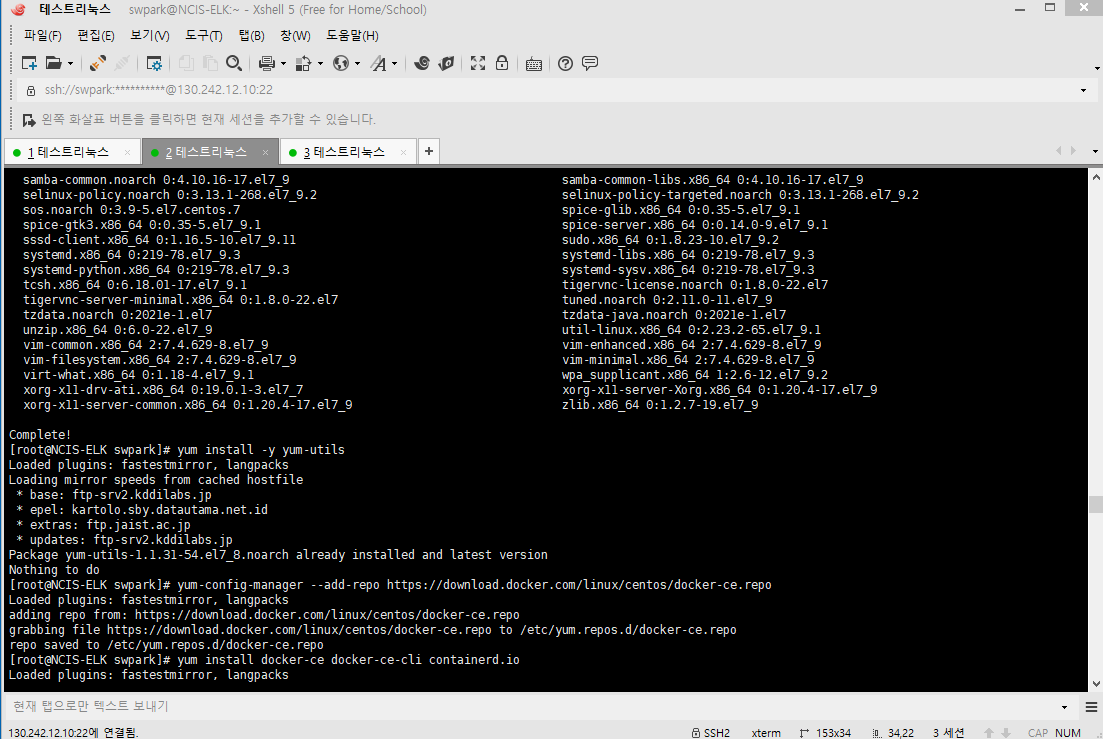
sudo systemctl enable docker

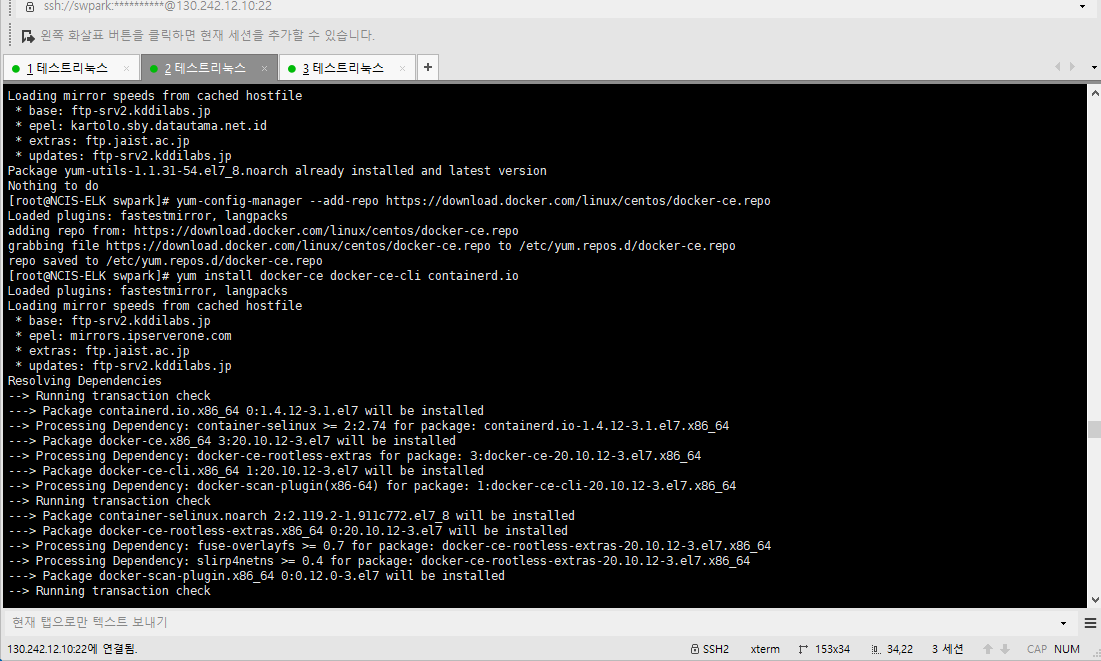
sudo systemctl status docker

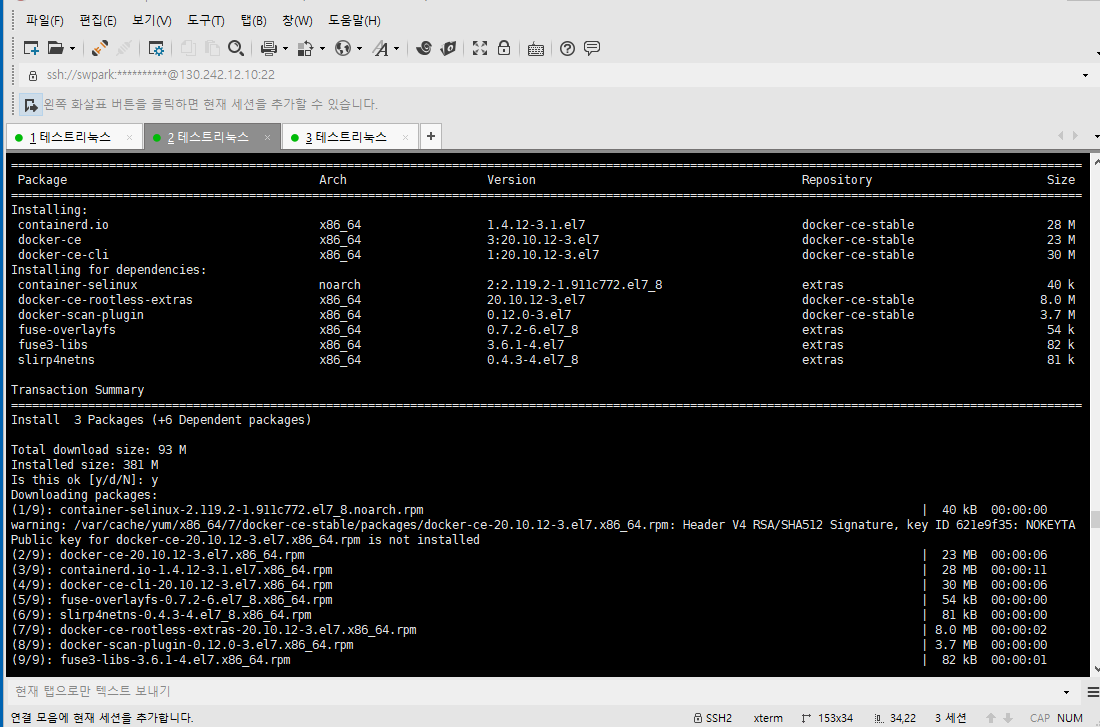
* 도커 실행 명령

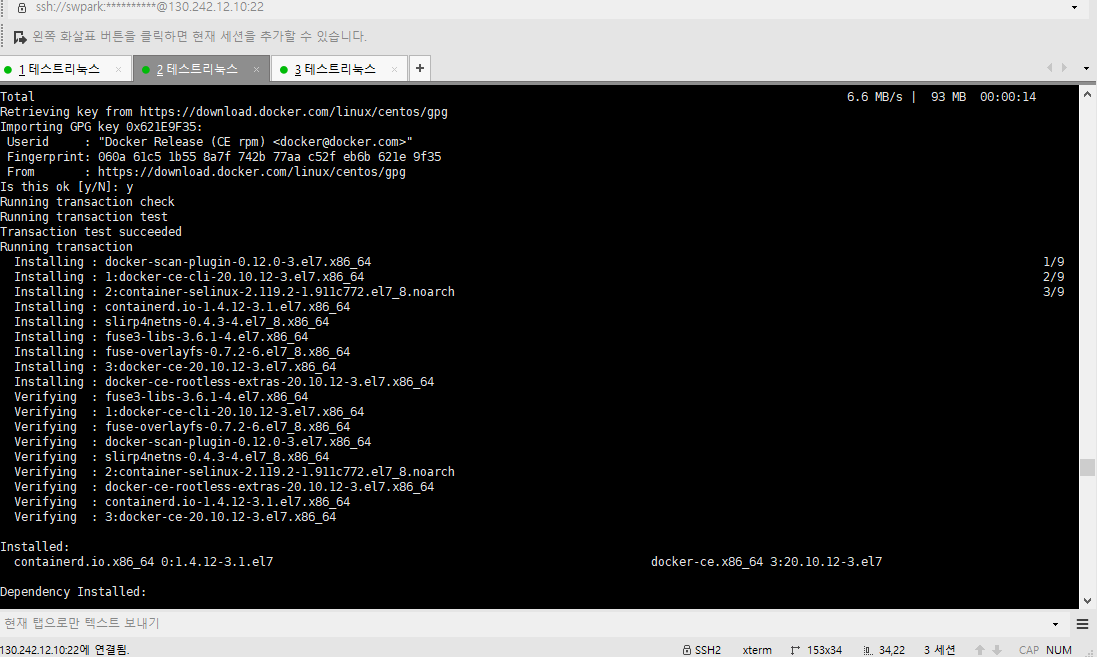
docker -v

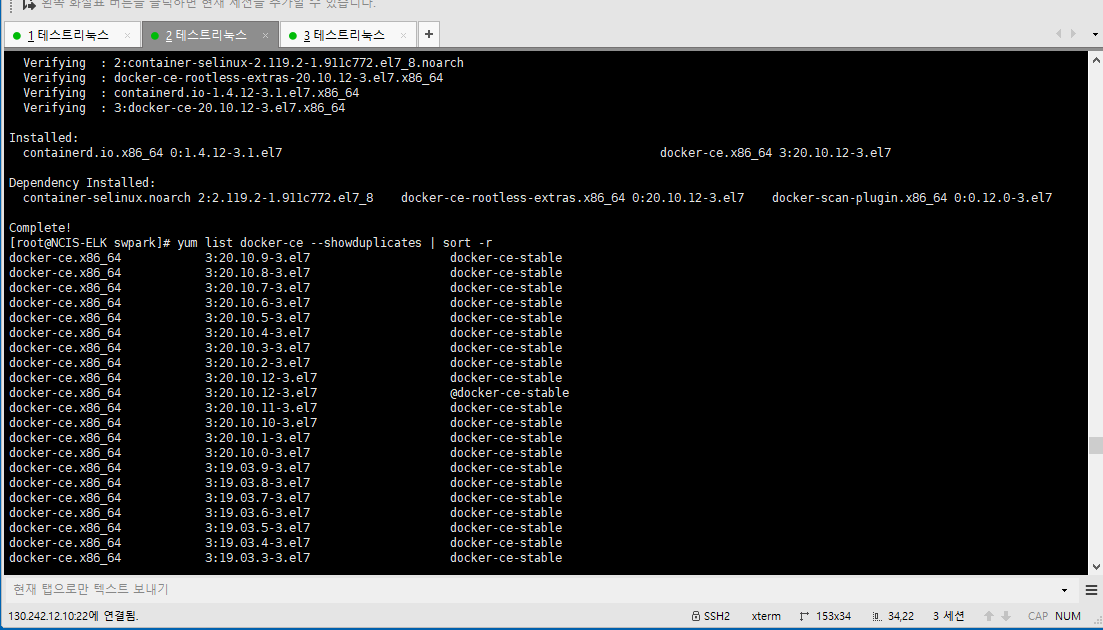
* 도커 버전을 확인해준다

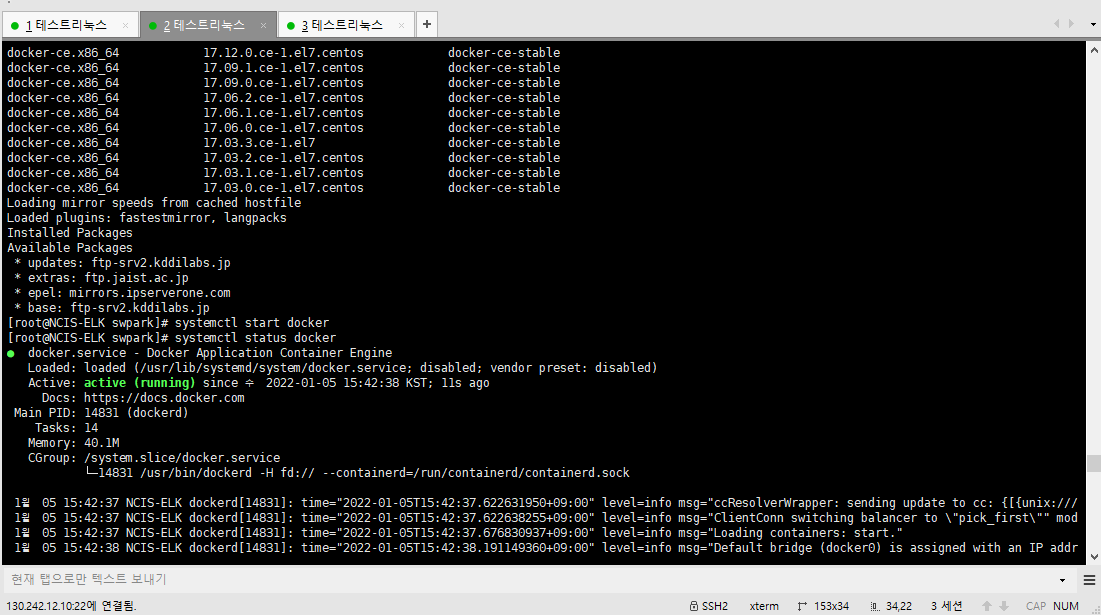


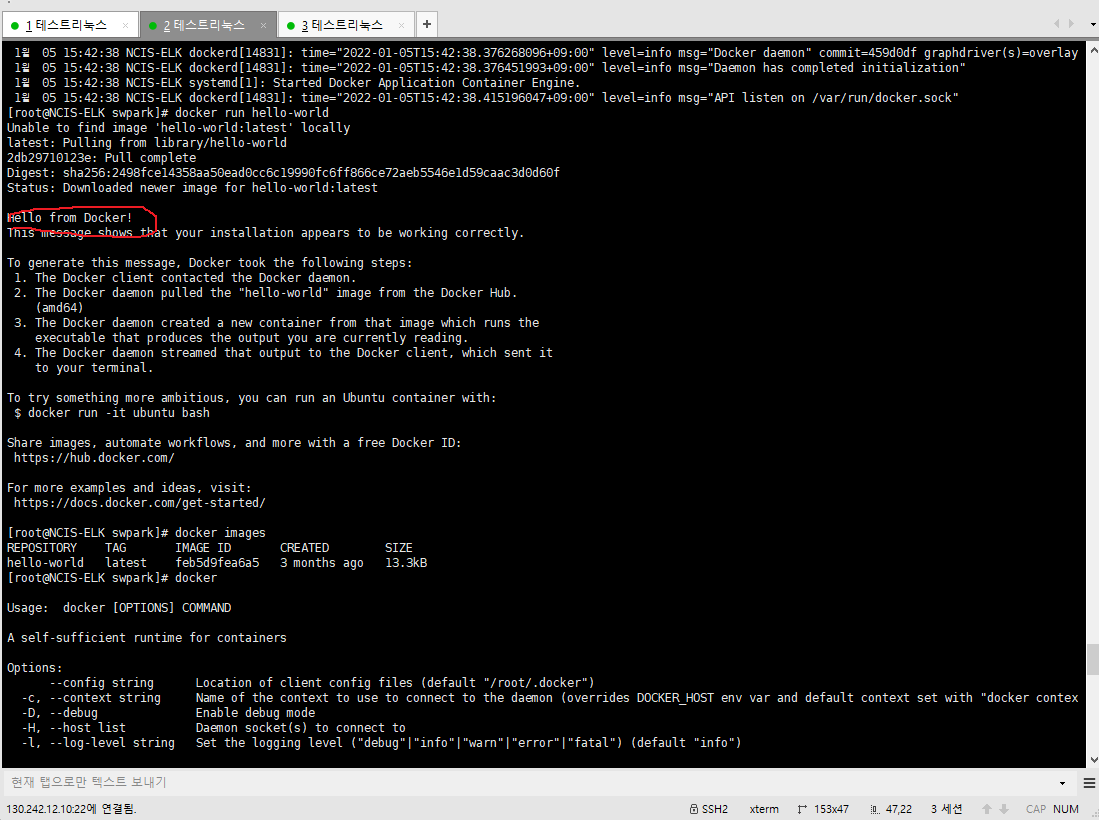




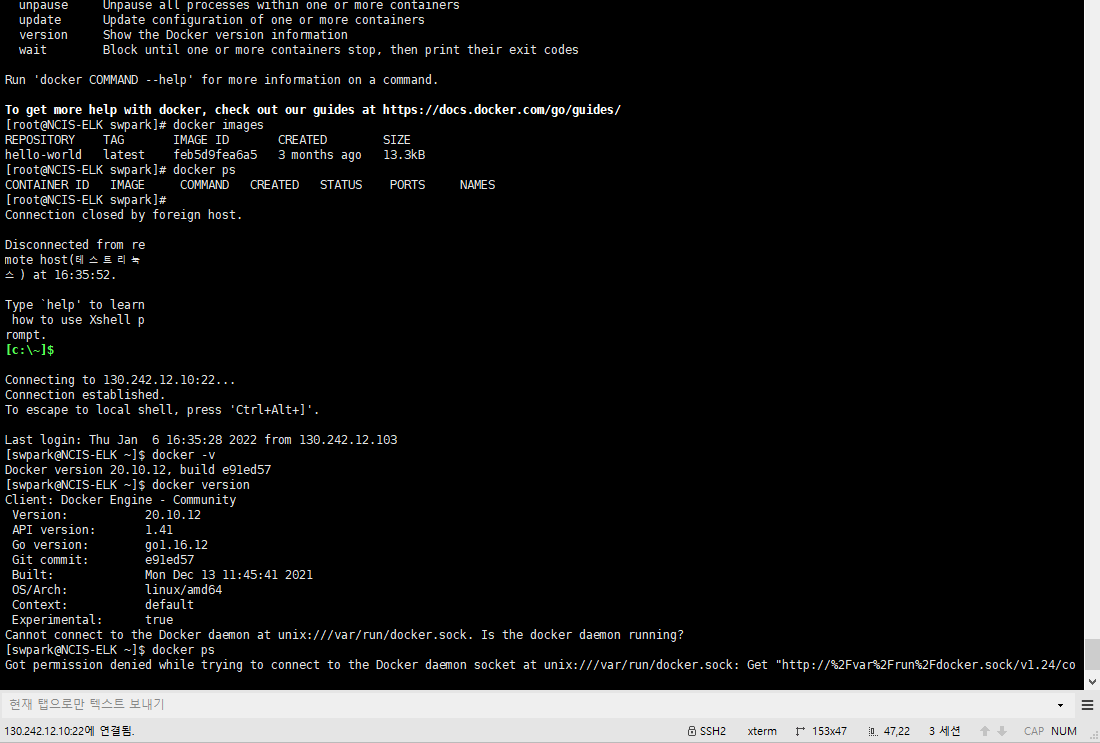






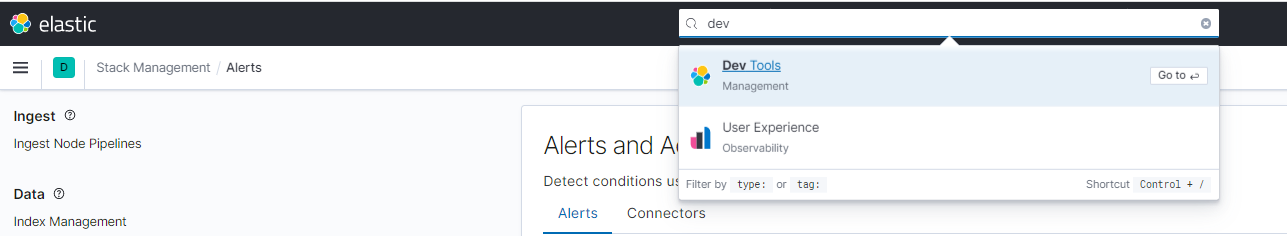


* Docker 설치가 완료되었다

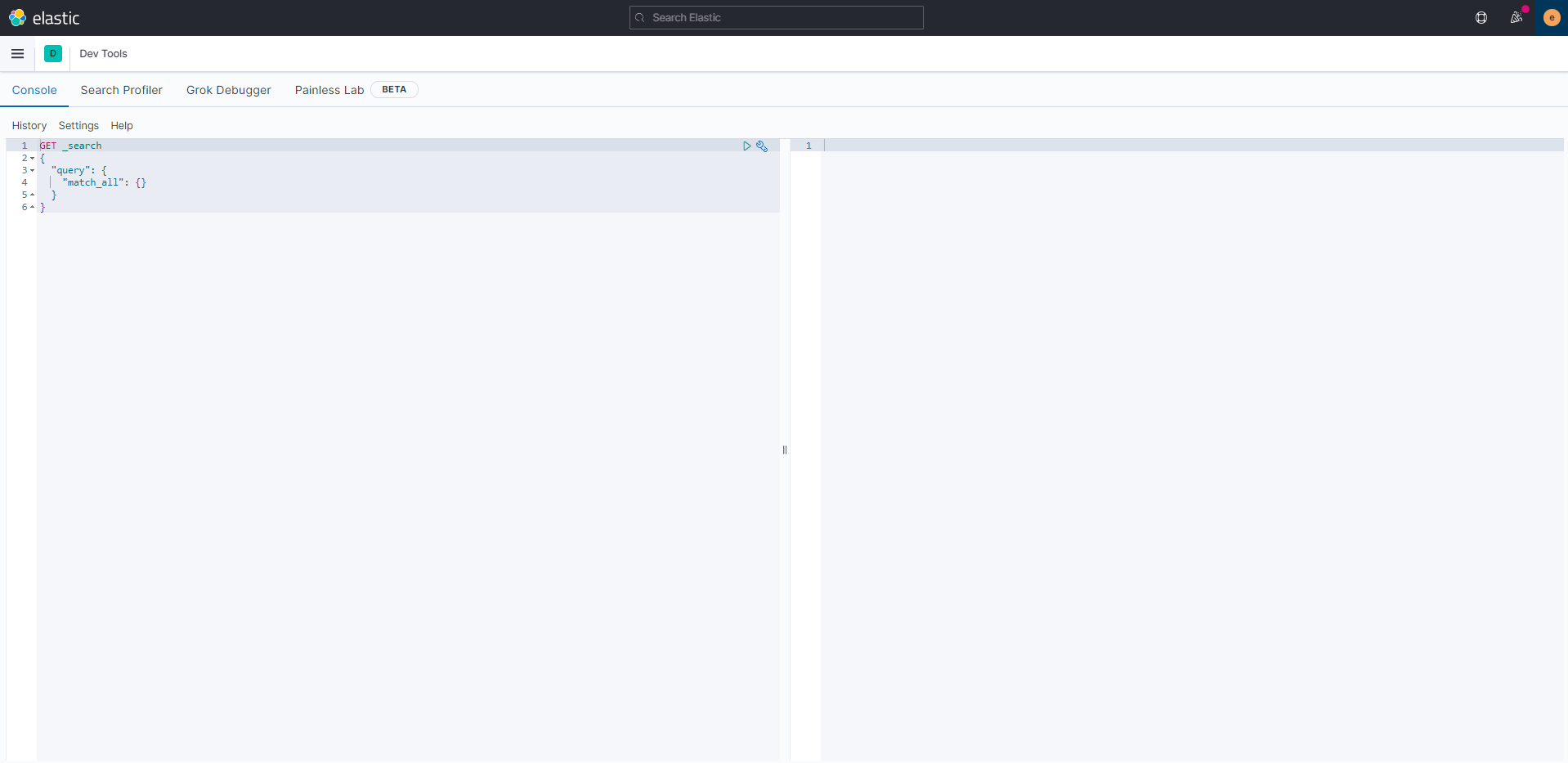


* 도커 버전 확인 : 20.10.12

이번에는 Elastic Query DSL에 관해서 알아봅니다. Kibana에는 해당 쿼리를 테스트 하기위한 기능인 Dev tools 를 제공합니다. Kibana의 상단에서 Dev tool을 검색해서 이동한다.



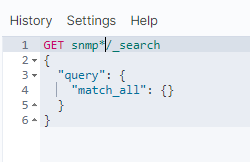
Dev tools



Dev tools는 엘라스틱이나 키바나에서 사용하는 여러 조건식, 쿼리를 작성하는데 도움을 준다.

이중에서 우리는 Elastic query DSL을 실험해 보는 console을 이용한다.

먼저 상단의 GET 옆에 검색하고자 하는 인덱스를 지정해 준다.



이러면 snmp로 시작하는 인덱스중에서 검색을 한다고 볼 수 있다.

다음으로 간단하게 검색을 하는 쿼리에 대해 알아본다.

- 쿼리 예시 1

|  |
| --- |
| GET snmp\*/\_search  {  "query": {  "bool": {  "must": [  {  "match": {  "ha\_status.keyword": "active"  }  },  {  "match": {  "tags": "fw"  }  }  ]  }  }  } |

위의 쿼리는 snmp\* 인덱스 중에서 tags에 fw가 있고 ha\_status.keyword가 active인 데이터를 출력해 달라는 쿼리이다.

엘라스틱 쿼리는 JSON의 형태를 지닌다.

처음 속성인 query는 본 JSON의 내용이 query라고 엘라스틱에게 알려준다.

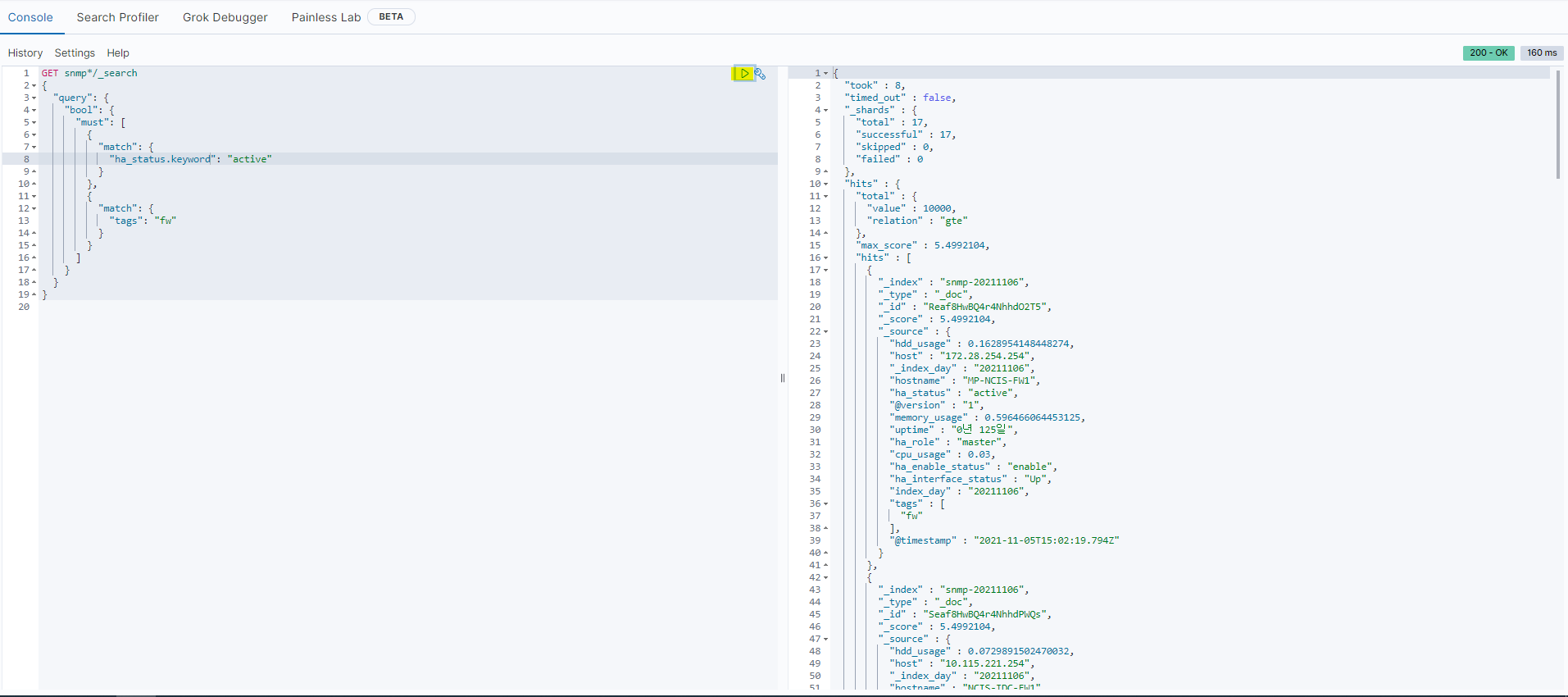
내부 속성인 bool은 쿼리를 참과 거짓으로만 구별한다인데 본래 엘라스틱의 경우 쿼리와 내용이 일치하지 않아도 해당 값에 대해 점수(score)를 부여해서 결과로 반환해 준다.

“bool” 속성의 경우 이런 점수화를 하지 않고 아래의 내용이 전부 일치해야만 값을 출력해야 할때 사용한다.

“must” 속성은 일반적인 쿼리문에서의 and와 같은 역할을 합니다. 배열형태의 값( [] )을 가지며 배열 안의 내용이 전부 참을 반환해야 한다.

“match”는 속성의 값이 같은 지 확인한다.

위의 쿼리를 실행한 결과 값은 다음과 같다.



상단의 재생 버튼을 눌러 쿼리를 실행한다.

total에서 몇개의 인덱스가 나왔는지 확인 할 수 있고 hits에서 해당 인덱스가 어떤 것인지 볼 수 있다.

- 쿼리 예시2

\* 실제 알림에 들어가는 query에는 맨 윗줄의 GET snmp\*/\_search 를 적지 않는다.

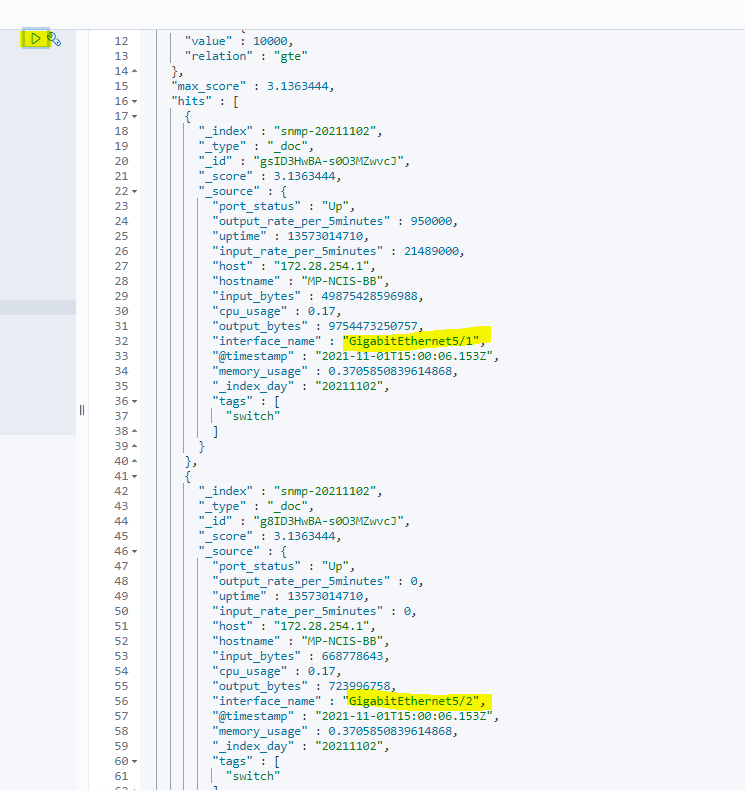
\* Dev tools 에서만 이용한다.

|  |
| --- |
| GET snmp\*/\_search  {  "query": {  "bool": {  "must": [  {  "match": {  "port\_status.keyword": "Up"  }  },  {  "match": {  "hostname.keyword": "MP-NCIS-BB"  }  },  {  "terms": {  "interface\_name.keyword": [  "GigabitEthernet5/1",  "GigabitEthernet5/2"  ]  }  }  ]  }  }  } |

위의 쿼리 예시와 다른 점으로는 term이라는 속성이 추가 되었다. term은 일반적인 쿼리에서의 or 을 의미한다.

따라서 해당 쿼리의 terms의 내용은 interface\_name.keyword 값이 "GigabitEthernet5/1" 또는 "GigabitEthernet5/2" 라면 결과를 반환하라는 의미가 된다.

이것으로 쿼리에 대한 간단한 설명을 마친다.



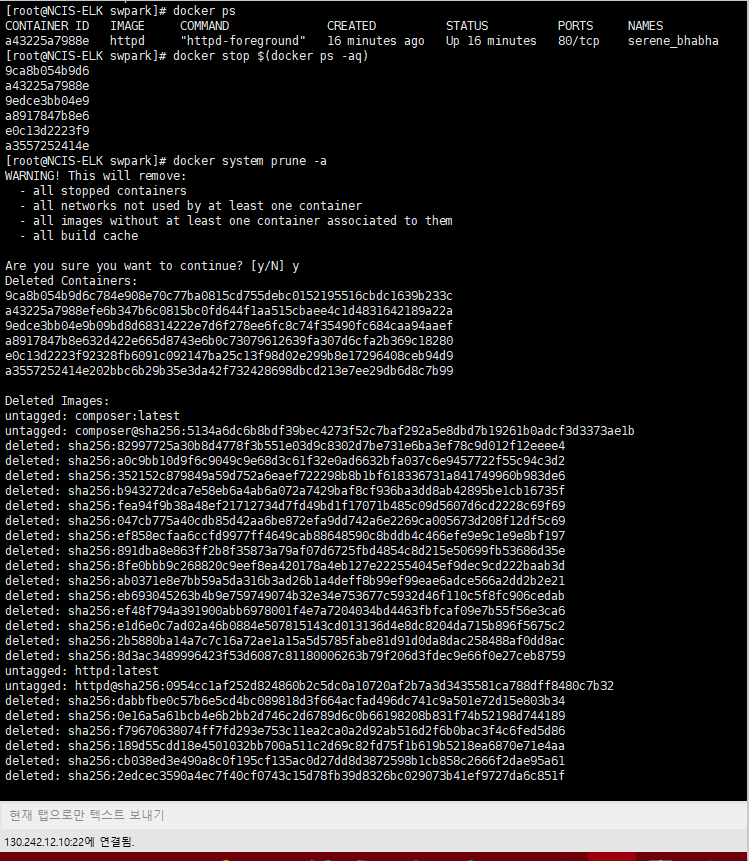
## Docker pull image 관련 명령

docker images 명령을 치면 현재 docker에 가지고 온 이미지들을 확인 할 수 있다.

docker ps명령으로 현재 동작 상태를 확인 할 수 있다.

실습을 위해 기존의 이미지들을 모두 삭제하기 위해 docker stop $(docker ps -aq

Docker system prune -a명령으로 한꺼번에 없애고 실습을 시작하였다.



위의 이미지는 쿼리를 실행했을때 docker에서 반환하는 결과 값이다.

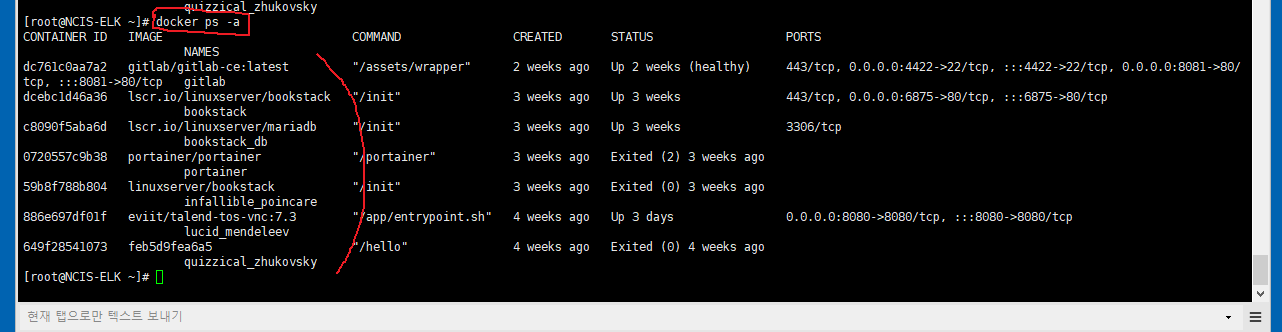
문서의 끝.

## Docker 사용하기(주요 명령어)

pull해온 이미지의 상태를 보기 위해 이 명령어를 수시로 사용하여 확인한다

docker ps -a

이 명령으로 현재 pull해온 이미지들을 확인할 수 있다



* docker ps -a명령을 사용하여 pull한 이미지 목록을 볼 수 있다
* 가져온 이미지 컨테이너의 ID 와 상태를 확인할 수 있다
* 우리가 실습한 bookstack, gitlab, talend이미지를 확인해 볼 수 있었다

다음으로는 docker에 사용할 이미지를 올리는 명령어를 사용해보겠다

우리는 docker에 talend를 올리기위해 그 이미지를 열어주어야 한다.

그러기위해서는 위의 container\_ID로 사용할 아이드를 복사하여

docker start 컨테이너ID 명령을 사용하여 활성화해준다

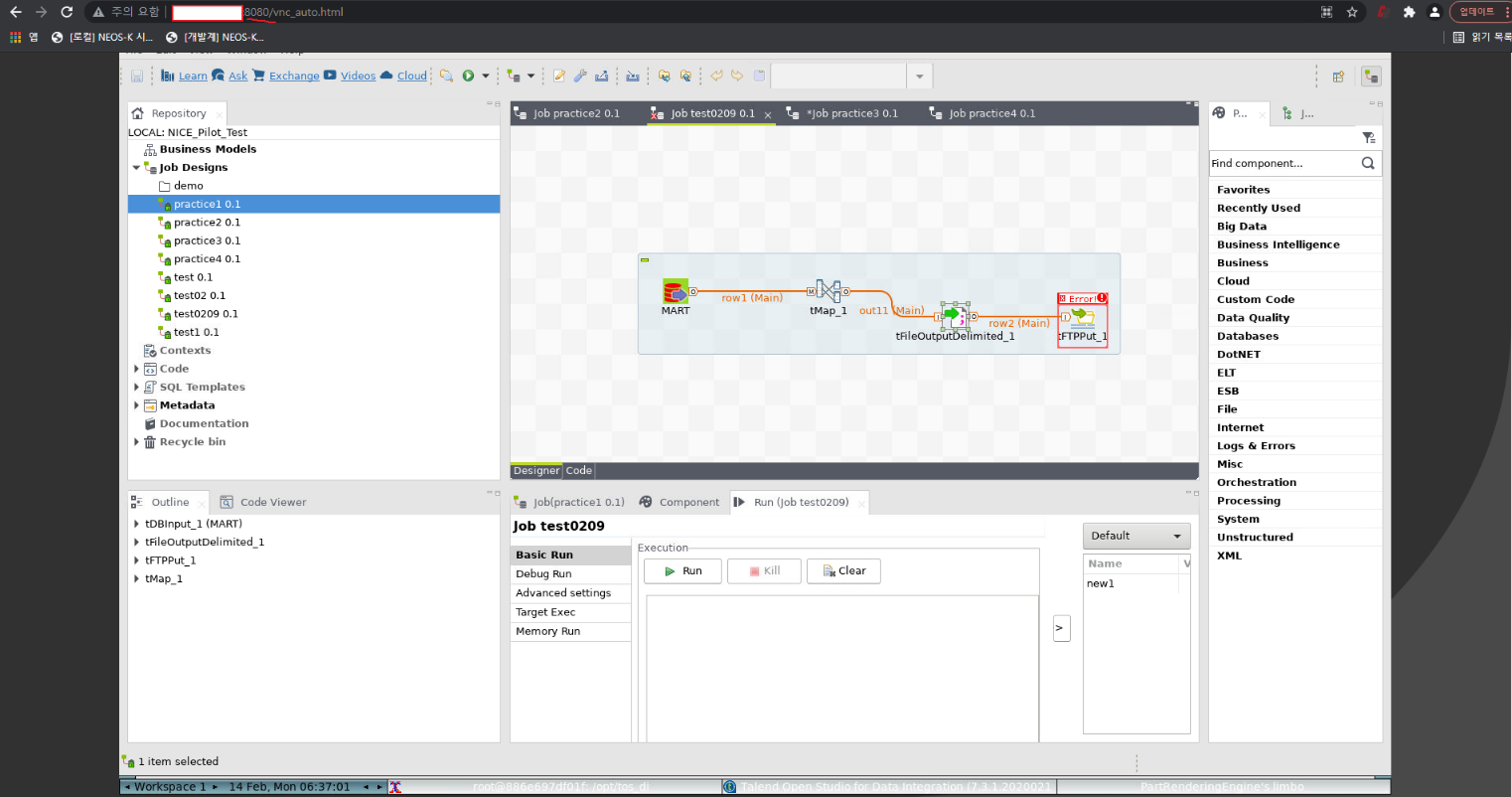
위명령을 사용하고 docker ps -a명령으로 확인해보면 위처럼 Status(상태)에 이미지가 잘 올라왔다면 UP이라고 떠있는 것을 알 수 있다.

올라온 이미지를 내리고 싶으면 docker stop 컨테이너ID명령을 사용하면 된다

내린 이미지를 다시 올리고 싶으면 docker restart 컨테이너ID명령을 사용하면 된다

* Docker 명령 정리
* docker ps -a
* docker stop 컨테이너ID 🡪 컨테이너 아이디는 docker ps -a명령으로 확인
* docker start 컨테이너ID
* docker restart 컨테이너ID

위의 올린 talend를 사용하기 위해 올라온 talend 포트번호를 IP와 함께 서서 실행하면 talend가 실행되는 것을 확인 할 수 있다



* 8080 포트번호로 talend가 docker위에 잘 올라온 것을 확인해 볼 수 있다

# Talend (TOS:Talend Open Studio)

## Talend란

Talend Open Studio는 개발, 테스트, 배포, 데이터관리 및 앱 통합프로젝트를 관리하기위한 다기능 공개 소프트웨어 세트이다.

Talend는 전사적 영역에 걸쳐 전체 생명주기를 관리하기 위한 통합 환경을 제공하여 데이터 관리 및 앱 통합을 용이하게 하는 통합 플랫폼을 제공한다

개발자는 사용하기 쉬운 데이터 통합, 데이터 품질, MDM, 앱 통합 및 빅테이터를 결합하여 이클립스 기반의 그래픽 환경을 통해 엄청난 생산성 향상을 얻어낼 수 있다

Talend Open Studio는 하둡, Hbase, 하이브와 같은 빅테이터 기술이 통합된 오픈소스환경과 결합하여 사용자가 쉽게 로드, 추출, 변환하고 다른 시스템에서 크고 다양한 데이터 세트를 처리할 수 있도록 지원한다.

추가적인 자세한 정보는 아래의 링크를 참고하였다(도움이 된 링크)

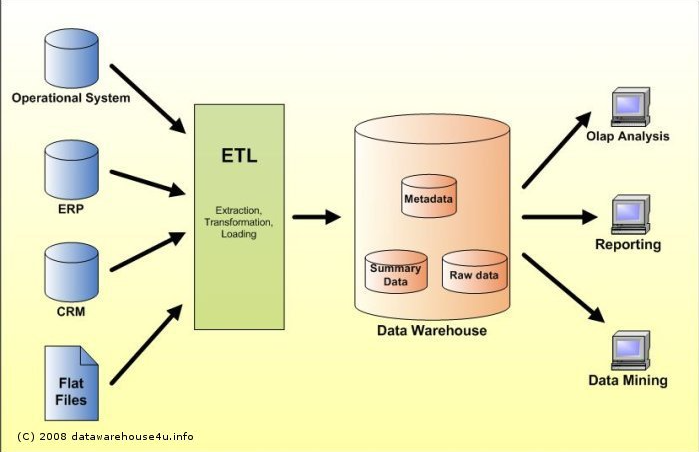
<https://www.talend.com/>

<https://help.talend.com>

Talend Open studio는 작업의 결과물로 Java나 Perl 코드를 직접 생성하는 것이 특징이다.

ETL의 도구이다

* + - etl이란



E(Extract, 추출), T(Transform, 변환), L(Load, 적재)를 나타내는 용어로,조직 내외부의 복수의 source들로부터 데이터 분석을 위한 데이터웨어하우스, 데이터 마트 내로 이동시키는 프로세스로서 추출, 재구성, 정제, 통합, 변형 등을 포함한다.

ETL은 컴퓨팅에서 데이터베이스 이용의 한 과정으로 동일 기종, 타 기종의 데이터 소스로부터 데이터를 추출하고, 조회 또는 분석을 목적으로 적절한 포맷이나 구조로 데이터를 저장하기 위해 데이터를 변환하고, 최종 대상으로 변환 데이터를 적재한다.

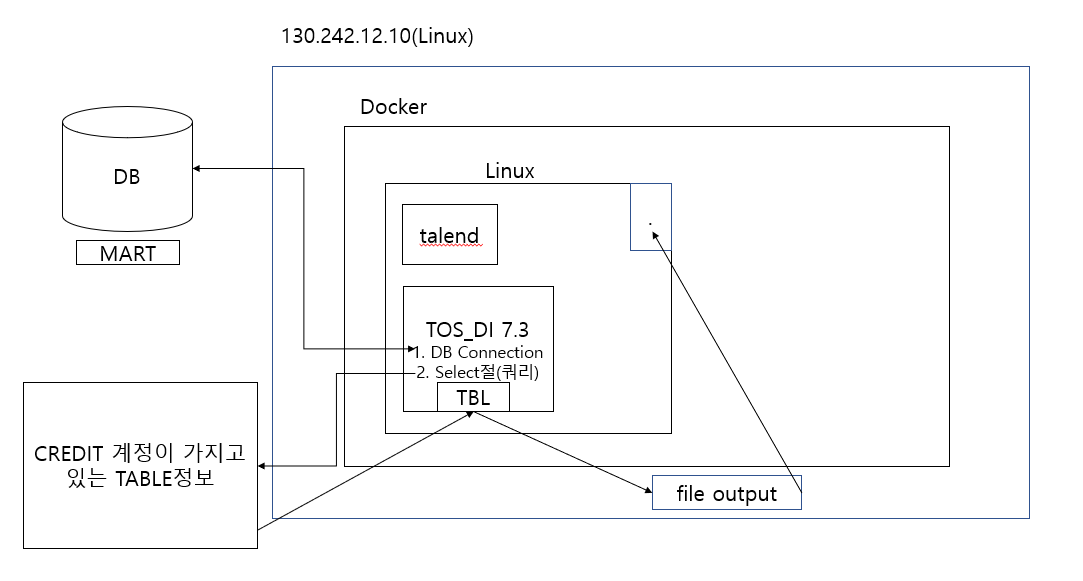
## Docker에 Talend 올리기

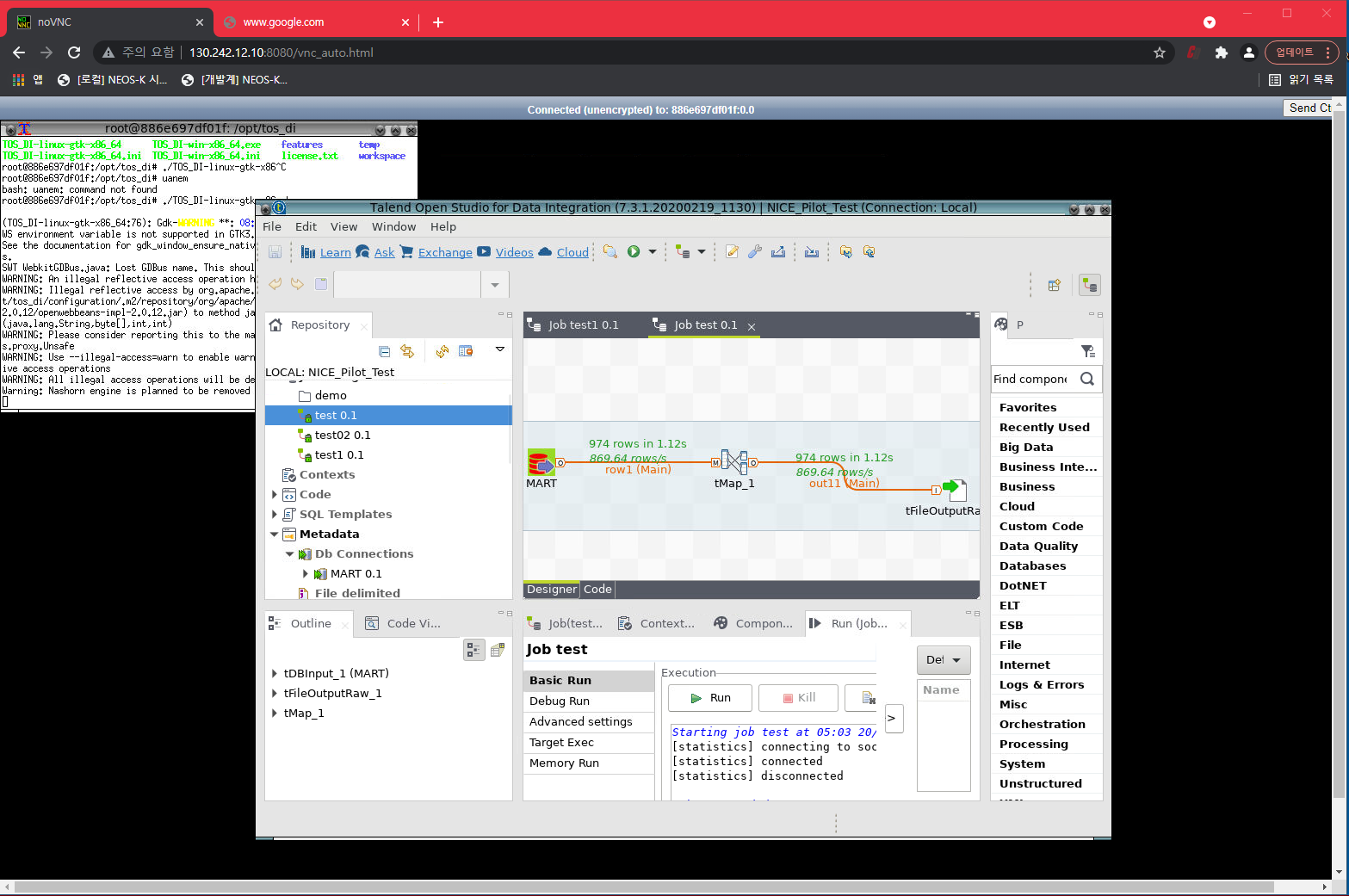
가상환경 Centos 버전 7환경에 Docker를 올리고 그 위에 Talend Open Studio 테스트 설치를 수행하였다.

# Talend 사용하기

## 도면

어떤 것을 하는지 도면을 작성해보았다.



****

**다음과 같이 수행된 것을 확인할 수 있다**

## 5.2 talend 사용하기

Workspace1과 workspace2를가상환경위에 놓고 workspace1 가상환경의 bash 쉘창을 열어서 pull명령으로 올린 talend를 실행하여 사용해보겠다.

실행을 위해서 workspace2 쉘창에 $ ./TOS\_DI-linux-gtx-x86.sh 명령을 쳐서 talend를 띄운다.

Talend 7.3버전을 설치하였다.

개발망 크롬창에 130.242.12.10:8080 ip를 쳐서 VNC에 접속하였다

## Talend 구성요소 목록

**tMysqlConnection** − 컴포넌트에 정의 된 MySQL 데이터베이스에 연결합니다.

**tMysqlInput** − 데이터베이스 쿼리를 실행하여 데이터베이스를 읽고 쿼리에 따라 필드 (테이블, 뷰 등)를 추출합니다.

**tMysqlOutput** − MySQL 데이터베이스에서 데이터를 작성, 업데이트, 수정하는 데 사용됩니다.

**tFileInputDelimited** − 구분 된 파일을 행 단위로 읽어서 별도의 필드로 나누어 다음 구성 요소로 전달합니다.

**tFileInputExcel** − 엑셀 파일을 한 행씩 읽어서 별도의 필드로 나누고 다음 구성 요소로 전달합니다.

**tFileList** − 주어진 파일 마스크 패턴에서 모든 파일과 디렉토리를 가져옵니다.

**tFileArchive** − 파일 또는 폴더 세트를 zip, gzip 또는 tar.gz 아카이브 파일로 압축합니다.

**tRowGenerator** − 샘플 데이터를 생성하기 위해 함수를 작성하거나 표현식을 선택할 수있는 편집기를 제공합니다.

**tMsgBox** − 지정된 메시지와 확인 버튼이있는 대화 상자를 반환합니다.

**tLogRow**− 처리되는 데이터를 모니터링합니다. 실행 콘솔에 데이터 / 출력을 표시합니다.

**tPreJob** − 실제 작업이 시작되기 전에 실행될 하위 작업을 정의합니다.

**tMap**− Talend 스튜디오에서 플러그인으로 작동합니다. 하나 이상의 소스에서 데이터를 가져 와서 변환 한 다음 변환 된 데이터를 하나 이상의 대상으로 보냅니다.

**tJoin** − 메인 플로우와 조회 플로우 사이에서 내부 및 외부 조인을 수행하여 2 개의 테이블을 조인합니다.

**tJava** − Talend 프로그램에서 개인화 된 자바 코드를 사용할 수 있습니다.

**tRunJob** − Talend 작업을 하나씩 실행하여 복잡한 작업 시스템을 관리합니다.

# Docker에 Bookstack 올리기

## bookstack이란 무엇인가

- bookstack은 간단하고 자체 호스팅되며 사용하기 쉬운 플랫폼을 목표로 하는 무료 오픈 소스 위키 소프트웨어이다.

- 단순하며 소규모 비즈니스에 적합하다.

- bookstack을 설치하려면 PHP 7.3 이상, MYSQL 5.6이상, GIT 및 Composer이 필요하다.

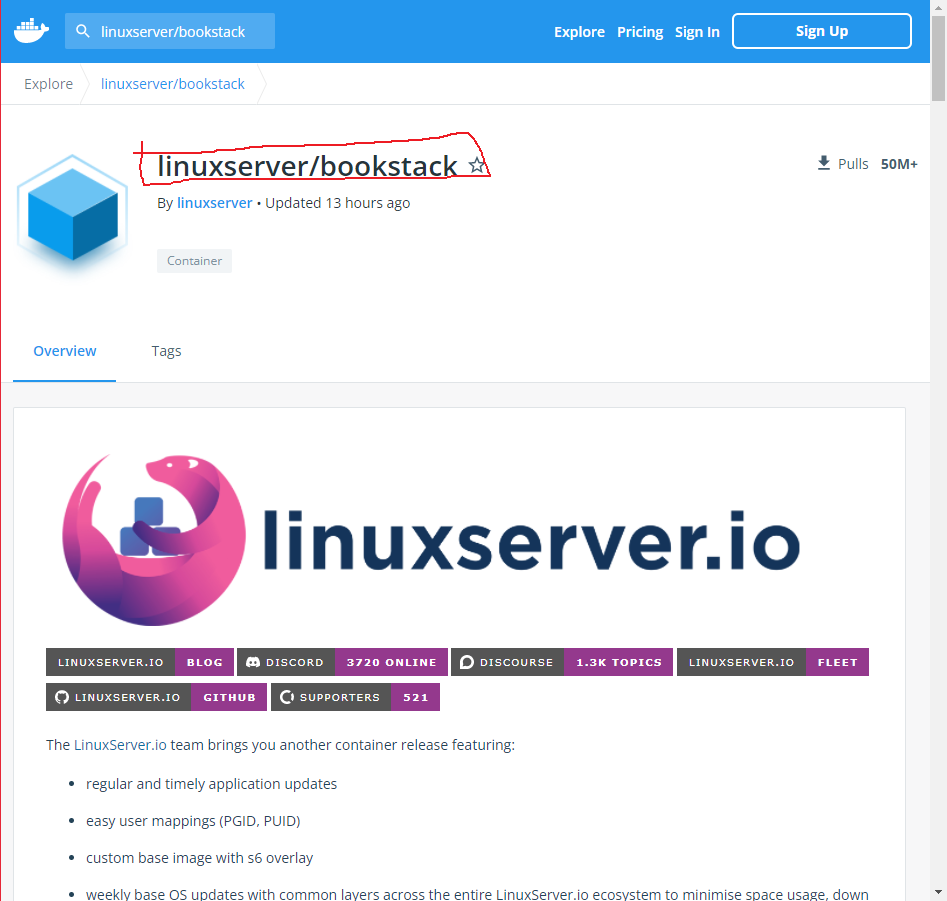
Docker 컨테이너를 통해 설치할 수도 있다.(이방식으로 테스트를 진행하였다)

- bookstack은 책 같은 목화처럼 단계적으로 작성할 수 있는 웹어플리케이션이다.

- 남들에게 체계적으로 정보를 제공할 때 유익하고 가볍고 심플한 UI를 가지고 있으며 Python으로

제작되었다

## 6.2 docker에 bookstack 올리기



* 크롬창에 docker hub를 검색하여 들어가서 검색창에 linuxserver/bookstack을 친다.
* 그러면 다음과 같이 docker에 올릴 수 있는 bookstack이미지가 나온다.
* 실행한 docker위에 pull명령으로 bookstack이미지를 가져온다
* 명령 : docker pull linuxserver/bookstack

가져온 bookstack이미지를 run을 한다.

* 명령 : docker run -it linuxserver/bookstack
* 명령 : docker run -d –p 8080:8080 linuxserver/bookstack

가상서버 8080아이피 번호로 pull해온 bookstack을 실행할 수 있다

관련 명령 수행중 삭제해야될 컨테이너명령 : docker rm 컨테이너ID

관련 명령 수행중 상황 확인 명령 : docker ps -a

* 두가지 명령을 적절하게 필요에의해 활용해가면서 수행하였다

# Docker에 Gitlab 올리기

## 7.1 Gitlab이란

* 버전관리 시스템

: 소프트웨어를 제작하면서 발생하는 변화들을 기록해서 소프트웨어의 변화를 열람하고, 수정할 수 있도록 도와주는 시스템

* Git

: Git의 원격 저장소의 역할을 제공하는 것이다.

* 원격 저장소

: Gitlab은 소스코드를 저장할 수 있는 서버를 운영할 수 있으며, 서버에 소스코드를 저장함으로써 다양한 협업을 할 수 있다.

* 이슈 트래커

: 업무를 추적하고 관리할 수 있게 해주는 로그, 즉 변화를 파악할 수 있도록 해준다

이것이 중요한 이유는 업무 변화를 잘 파악하고 관리할 수 있게 해준다.

## Gitlab의 특징

* 이슈 트래커 제공
* Git 원격 저장소 제공
* API 제공
* Team Group 기능 제공
* 이 특징들로 인해서 업무를 편리하게 진행하게 도와주며 Github과 사용방법이 유사하며 가격 경쟁력도 있으며 효율적이다.

## Gitlab 기능

* 강력한 클라우드 네이티브를 지원한다
* 그룹 및 프로젝트 관리

: 그룹 및 프로젝트를 통해 업무를 계획하고 세부화한 업무를 구성원에게 배정하여 팀의 수행 프로젝트를 관리할 수 있다. 그룹은 동일한 네임스페이스에서 관련 프로젝트를 구성하고 최상위 그룹에 구성원을 추가하여 여러 프로젝트와 여러 팀 구성원에게 더 적은 단계로 액세스 권한을 부여할 수 있다. 프로젝트는 코드베이스 단위로 만들며 이를 이슈 트래커로 사용하고, 코드에 대해 공동 작업하고, 내장된 Gitlab CI/CD를 사용하여 앱을 지속적으로 빌드하고 테스트하고 배포할 수 있다

* 리포지토리

: 프로젝트를 생성하면 자동으로 git원격 저장소가 생성된다. 코드베이스를 저장하고 버전 제어를 통해 코드를 변경하는데 사용되며, 파일을 탐색하고 상호 작용하는 다양한 방법을 Web UI로 제공한다

* 이슈 트래커

: 공동으로 아이디어를 개발하고 문제를 해결하며 작업을 계획하기 위한 고급도구로, 이는 새로운 아이디어의 논의, 작업 상태 추적, 기능 제안, 지원 요청등 새로운 코드 구현에 대해 자세히 설명하는데 활용할 수 있다.

* 이슈 보드

: 기능 또는 제품 릴리스에 대한 워크플로우를 계획, 구성 및 시각화하는데 사용되는 소프트웨어 관리 도구이다. 레이블을 추가하여 기존 이슈에 해당하는 목록을 만들고 카드를 한 단게에서 다른 단계로 끌어다 놓을 수 있다.

* 레이블

: 이슈, 병합 요청 및 에픽의 수가 증가함에 다라 이러한 항목을 추적하는 것이 점점 더 어려워지는데 이를 레이블은 작업을 구성하고 태그를 지정하여 관심있는 항목을 추적하고 찾을 수 있도록 도와준다. 버그, 기능요청 같은 설명 제목과 색상을 사용하여 에픽, 이슈 및 병합 요청을 분류하여 동적으로 필터링하고 관리하며 이런 항목들을 검색할 수 있다.

* 마일스톤

: 특정 기간 동안 더 광범위한 목표를 달성하기 위해 생성된 이슈 및 병합 요청을 추적하는 방법이다. 이를 이용하면 선택적 시작일과 마감일을 사용하여 이슈와 병합 요청을 응집력 있는 그룹으로 구성할 수 있다.

* 병합 요청

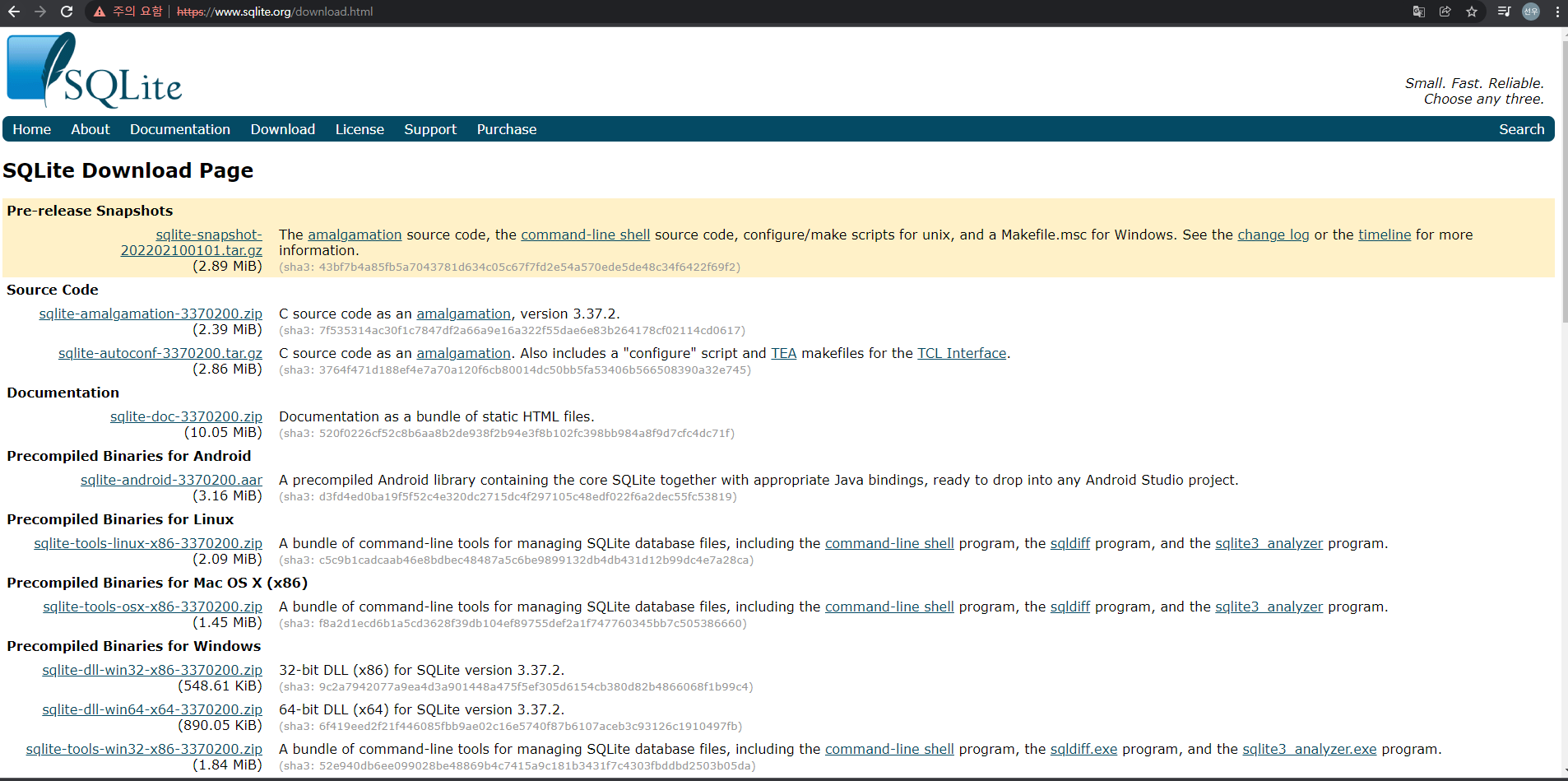
: 한 브랜치를 다른 브랜치로 병합하기 위한 요청이다. 병합 요청을 사용하여 소스 코드에 대해 제안된 변경사항을 시각화하고 협업할 수 있다.

* WIKI

: 문서화를 위한 별도 시스템이 각 Gitlab프로젝트에 내장되어 있다. WIKI는 문서를 저장소에 보관하고 싶지 않지만 코드가 있는 동일한 프로젝트에 보관하고 싶은 경우 배우 편리하다. 모든 WIKI는 별도의 Git저장소이므로 웹 인터페이스에서 또는 Git Clone하여 로컬에서 작업할 수 있다.

# SQLite 브라우저 설치

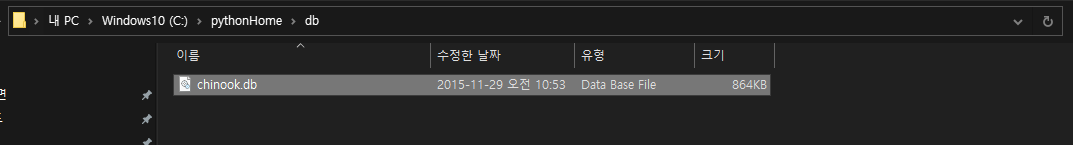
## 6.1. SQLite DB



<https://www.sqlite.org/download.html>

SQLite 수행을 해보기 위해 위의 사이트에 들어가 자신의 PC환경에 맞는 설치파일을 다운로드 받는다

그다음으로는 예제db파일을 다운받는다 (우리는 상무님께서 보내주신 블로그의 chinook.db 예제 파일을 받았다)

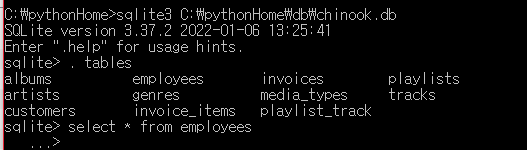


이를 C드라이브아래에 pythonHome이라는 명칭의 폴더를 만들고 그안에 db라는 명칭의 폴더를 만들고 그안에 chinook.db의 경로를 지정해주었다

그다음에는 cmd창으로 들어가서 다음 명령들을 사용하여 해당 경로로 간뒤 sqlite를 실행하였다

Cd C:\pythonHome

Sqlite3 C:\pythonHome\db\chinook.db



Sqlite에 접속하여 예제db안의 테이블이 어떤 것들이 있는지 보기 위해 . tables명령을 수행해보았다. 여러 테이블이 있는 것을 확인해 볼 수 있었다.

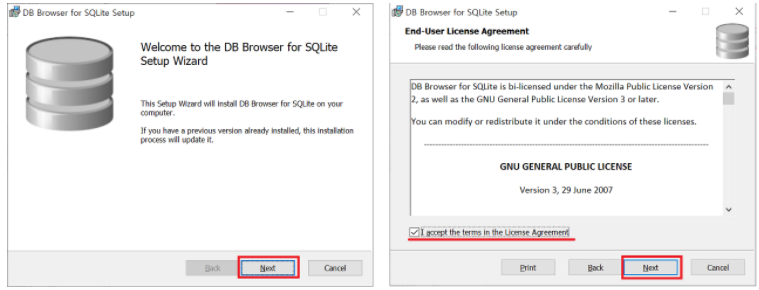
## 6.2. 프로그램 다운로드 하기

윈도우 10 64bit 사용자라면 다음 링크를 통해서 다운 받으면된다

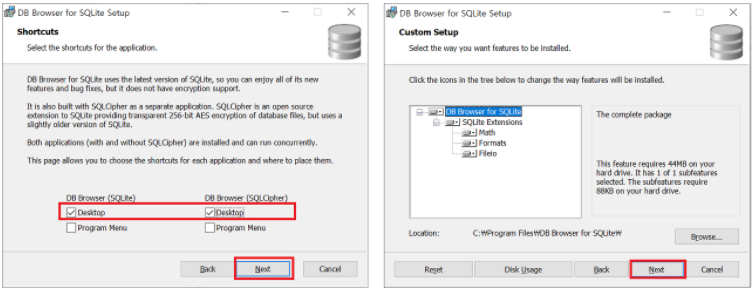
DB Browser for SQLite – Standard installer for 64-bit Windows

(리눅스의 경우 sqlitebrowser.org/dl/ 에서 다운)

다운로드 후 받은 파일을 실행한다

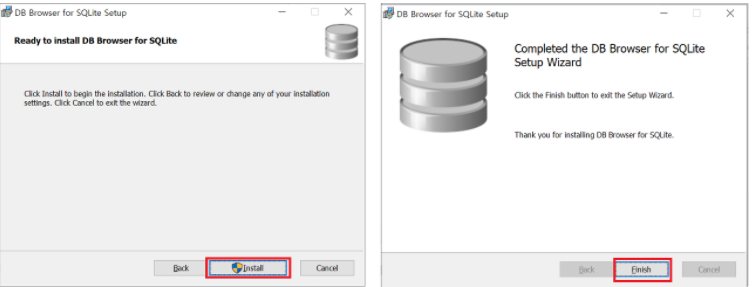


동의 후 next 버튼 클릭한다

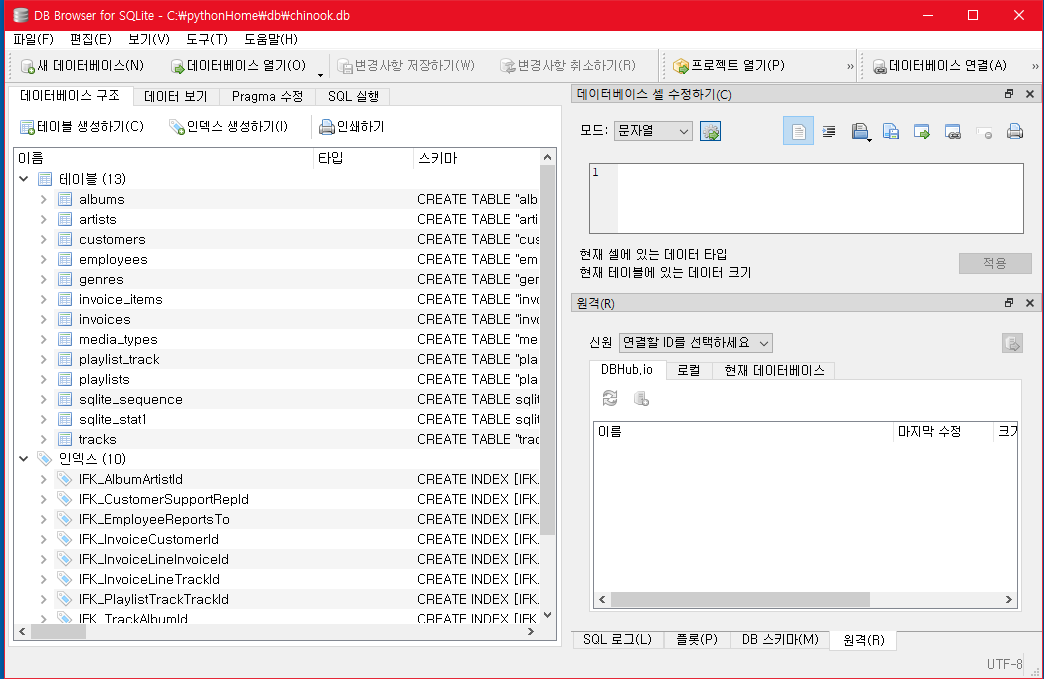


바탕화면 바로가기 아이콘 생성여부를 체크한다. 체크 후 Next버튼 클릭한다

프로그램을 설치할 디렉토리를 선택한다.



Install버튼을 클릭한다. 그다음 Finish버튼을 누르면 설치가 완료된다



설치된 바로가기 아키콘을 클릭하여 실행해서 상단의 데이터베이스 열기-> 만들어놓은 DB파일이

있으면 읽어온다(우리는 chinook.db파일을 가져왔다)

가져온 유의 테이블과 인텍스, 뷰, 트리거 등의 정보를 확인할 수 있다

설치한 이를 통해서 테이블의 정보도 볼 수 있고, 새롭게 생성해 볼 수도 있다.