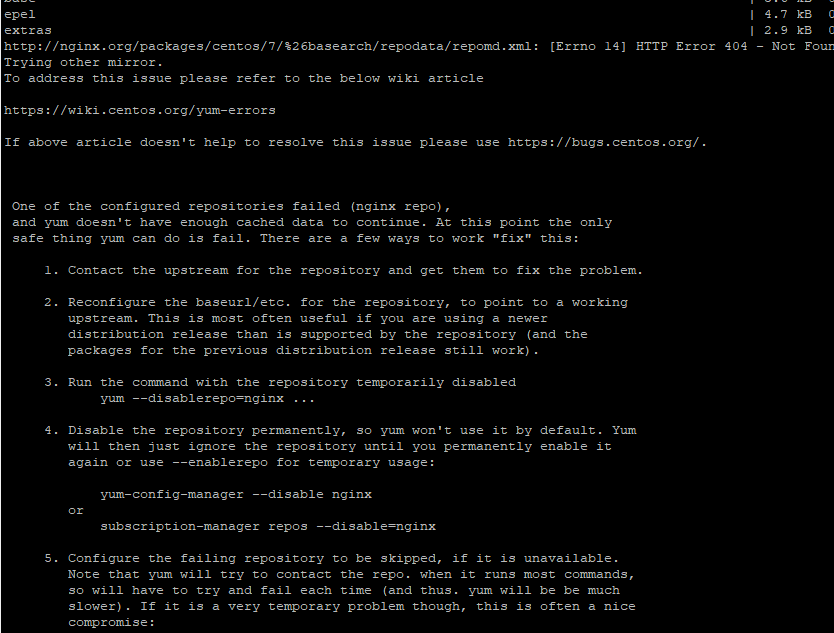
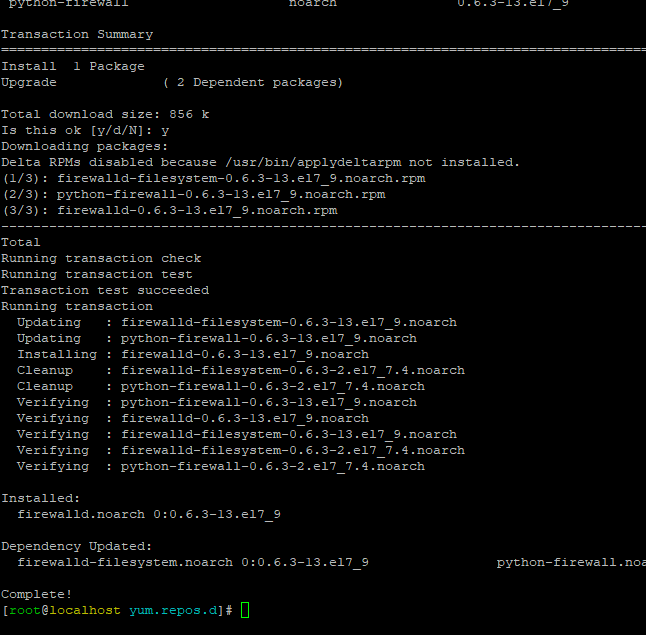
2021.07.09 금요일 \_ 업무내용(정준호)

**1.** NGINX에 대한 이해 시작함.   
(NGINX는 차세대 웹 서버로 불림. 더 작은 자원으로 더 빠르게 데이터를 서비스 할 수 있다.)

[[CentOS 7] CentOS 7 - NGINX 설치 방법 :: 정리하며 배우다. (tistory.com)](https://holjjack.tistory.com/114)   
이곳을 통해서 CentOS에 nginx 설치함.

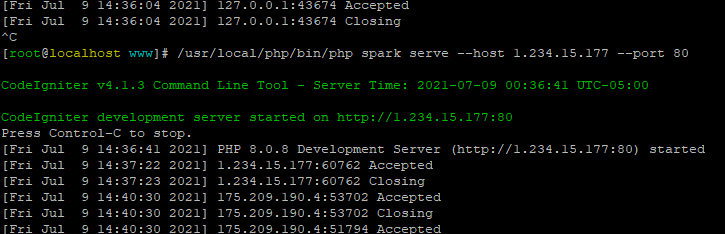


Nginx를 다운로드 이후, 방화벽 포트 개방시에 문제가 발생함.



문제가 발생하였으나, 구글링을 통해서 configure를 통해서 이를 해결함.

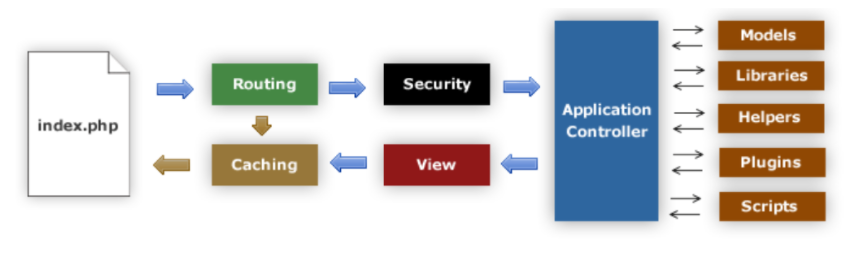
결국 nginx를 다운 받기 전에, spark module을 모두 다운 받았고, 아래와 같이 잘 실행 되는 것을 알 수 있다. 따라서 nginx로 서버를 실행하여 개발하는 것 보다, spark로 서버를 열어 개발하는 쪽으로 방향 바꿈. 현재 spark의 디렉토리와, 우리가 사용하고 있는 서버를 넣었고, port는 80임.



**2. CodeIgniter -> CI**   
[CentOS 7에 CodeIgniter를 설치하는 방법 (linux-console.net)](https://ko.linux-console.net/?p=118)  
이곳에서 URL 변경하는 방법 참조,

MVC 패턴 숙달 진행

Application Flow 확인.



**- 프레임 워크 구조**  
1.컨트롤러 -> (모델에서 필요한 데이터 처리) -> 뷰로 넘겨주기  
URL 구조: localhost/index.php/컨트롤러/컨터롤러\_함수/인자들..

2. 모델  
모델은 대체적으로 DB와 상호작용하는 직접적인 역할을 한다.

3. 뷰는 말그대로 사용자에게 보여지는 파일이다.

참조 문서 : [정적 페이지(Static pages) — CodeIgniter 4.1.3 documentation (cikorea.net)](http://ci4doc.cikorea.net/tutorial/static_pages.html)  
가장 먼저 할 일은 정적 페이지를 처리할 controller를 설정하는 것임. 컨트롤러는 웹 어플리케이션의 접착제 역할을 하며, 작업 위임을 돕는 클래스임. -> 첫번째 컨트롤러 만들었음(클래스 안에서 함수 사용하면 그걸 메서드라고 함.)->컨트롤러 로직 추가 -> 앱실행 -> 라우팅(app/Config/Routes.php)

**CI Framework 및 PHP 공부**