

- logging이란 애플리케이션이 동작하는 동안 시스템의 상태나 동작 정보를 시간순으로 기록하는 것을 의미
- 로깅은 개발 영역 중 비기능 요구사항에 속함
 - 사용자나 고객에게 필요한 기능은 아니라는 의미
- 로깅은 디버깅이나 개발 이후 발생한 문제를 해결할 때 원인을 분석하는 데 꼭 필요한 요소임
- 자바 진영에서 가장 많이 사용되는 로깅 프레임워크는 Logback을 많이 사용함
- log4j 이후에 출시된 로깅 프레임워크로서 slf4j를 기반으로 구현되어 있으며, log4j에 비해 월등한 성능을 자랑함
- 스트링 부트의 spring-boot-starter-web 라이브러리 내부에 내장돼 있어 별도의 의존성을 추가하지 않아도 사용할 수 있음

특징

- 5개의 로그 레벨을 설정함
- ERROR
 - 로직 수행 중에 시스템에 심각한 문제가 발생해서 애플리케이션의 작동이 불가능한 경우를 의미
- WARN
 - 시스템 에러의 원인이 될 수 있는 경고 레벨을 의미
- INFO
 - 애플리케이션의 상태 변경과 같은 정보 전달을 위해 사용
- DEBUG
 - 애플리케이션의 디버깅을 위한 메시지를 표시하는 레벨
- TRACE
 - 레벨보다 더 상세한 메시지를 표현하기 위한 레벨을 의미

-
- 실제 운영 환경과 개발 환경에서 각각 다른 출력 레벨을 설정해서 로그를 확인 할 수 있음
 - Logback의 설정 파일을 일정 시간마다 스캔해서 애플리케이션을 재가동하지 않아도 설정을 변경할 수 있음
 - 별도의 프로그램 지원 없이도 자체적으로 로그 파일을 압축할 수 있음
 - 저장된 로그 파일에 대한 보관 기간 등을 설정해서 관리할 수 있음

패턴 의미

%Logger{length} 로거의 이름

%-5level 로그 레벨. -5는 출력 고정폭의 값

%msg(%message) 로그 메시지

%d 로그 기록 시간

%p 로깅 레벨

%F 로깅이 발생한 애플리케이션 파일명

%M 로깅이 발생한 메서드 이름

%I 로깅이 발생한 호출자의 정보

%thread 현재 스레드명

%t 로깅이 발생한 스레드명

%c 로깅이 발생한 카테고리

%C 로깅이 발생한 클래스명

%m 로그 메시지

%n 줄 바꿈

%r 애플리케이션 실행 후 로깅이 발생한 시점까지의 시간

%L 로깅이 발생한 호출 지점의 라인 수