* Module.expports, exports
  + application에 최종적으로 내보내지는 것은 module.exports다. exports는 초기 단계에서 속성을 추가할 수 있는 빈 객체인 module.exports의 전역 참조 역할을 하고 있다. exports에 다른 값이 할당되면 exports는 module.exports를 가리키지 않고 참조가 깨진다. 참조 관계를 유지하려면 module.exports = exports = myModule;로 다시 reference를 대입하면 된다.
* Node module search
  + Node는 module을 찾을 때, 기본 모듈인지 먼저 판단하고 기본 모듈이 아니면 가까운 위치부터 상위까지 node\_modules directory가 있는지를 판단한 다음 찾는 모듈이 node\_modules directory에 있으면 모듈을 반환하고 없으면 최상위 node\_modules directory까지 찾는다. 최상위 node\_modules directory에도 없으면 NODE\_PATH 환경 변수에 지정된 path에서 찾고 없으면 예외를 발생한다.
* node는 module을 객체 형태로 cache한다. application의 두 파일에서 같은 module을 이용하는 경우 첫 번째 호출 시 module을 application의 memory 영역에 저장한다. 따라서 두 번째 호출부터 모듈의 소스 파일에 접근하여 확인하는 과정이 필요 없다.
* 몽키 패칭
  + 실행 중 동적으로 속성을 변경한다.(cache된 데이터를 직접 변경) 하나의 모듈을 다르게 행동하도록 수정할 수 있게 하여 개발자가 새로운 버전의 모듈을 다시 만들 필요가 없다.
* Callback, eventListener
  + callback : 일반적으로 일회성 응답에 대한 로직을 정의하는데 사용
  + eventListener : 기본적으로 개념적인 요소(event)에 연관된 callback(event와 event가 발생할 때마다 실행되는 callback function과의 연관 관계)