빅데이터(big data)

1. 개요

- 현재, 빅데이터는 **수십 테라** 이상의 데이터를 의미한다.
- 빅데이터는 기존의 데이터베이스로 관리하기 어려운 대용량의 데이터를 지칭한다.
- 빅데이터는 기존의 데이터베이스시스템으로는 데이터 분석이 불가능하다.
- 빅데이터는 **정형, 반정형, 비정형** 등 다양한 형태의 데이터를 포함한다.
- 소셜미디어, 스마트폰 폭증, 멀티미디어 콘텐츠는 빅데이터와 직접적인 연관을 가진다.
- 자연어 처리는 빅데이터 분석기술 중의 하나이다.

// 빅데이터 특징

- 빅데이터는 **디지털 환경**에서 생성되는 데이터이다.
- 빅데이터는 기존 데이터에 비해 **생성주기**가 짧다.
- 빅데이터는 형태가 **다양**하다. (수치, 문자, 영상 등)
- 빅데이터는 규모가 **방대**하다. 대량의 정형 또는 비정형 데이터를 포함한다.
- 빅데이터를 이용하여 사람들의 행동, 생각과 의견까지 분석하고 **예측**할 수 있다.
- 빅데이터는 기존 데이터베이스 관리 도구의 분석 역량을 넘어서는 대량의 데이터이다.
- 빅데이터는 데이터로부터 가치를 추출하고 그 결과를 분석하는 기술을 의미하기도 한다.
- 빅데이터는 단순히 '많은 양의 데이터'를 의미하는 것이 아니다.
- 빅데이터는 **크기(voulme), 속도(velocity), 다양성(variety)** 등의 차원에서 기존 데이터 개념과는 구별된다.

// 빅데이터 모델 - 3V / 4V

3V 모델 - 2001년, 더그 레이니(Doug laney)의 연구 보고서에서 정의

데이터 규모(volume)
데이터 입출력 속도(velocity)
데이터 종류의 다양성(variety)

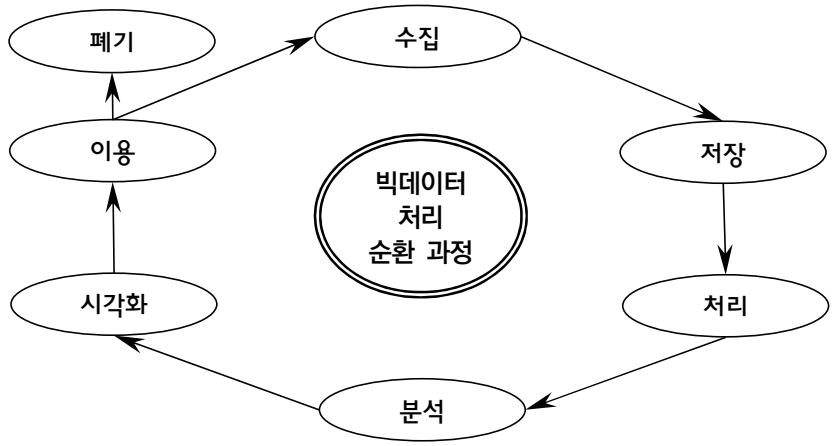
- 브라이언 홉킨스(Brian hopkins)은 **가변성(variability)**을 추가하여 **4V**를 정의하였고
- IBM은 **진실성(veracity, 정확성)**이라는 요소를 더해 **4V**를 정의하였다.

2 http://cafe.daum.net/pass365(홍재연)

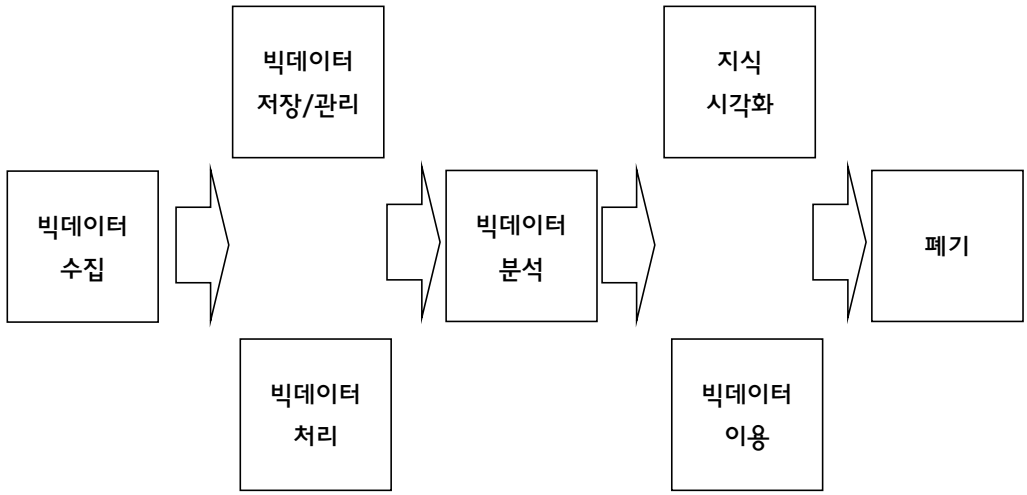
// 빅데이터 처리 과정

수집 → 저장 → 처리 → 분석 → 시각화(표현) → 이용 → 폐기

① 빅데이터 처리 과정(출처, 스마트 교육 환경에서 빅데이터 동향, 재구성)



② 빅데이터 처리 과정(출처, 황승구 외)



2. NoSQL이란?

- NoSQL : "Not Only SQL" 약어
- NoSQL은 SQL만을 사용하지 않는 DBMS을 지칭하는 단어이다.
- 관계형 데이터베이스를 사용하지 않는다는 의미는 아니다.
- NoSQL은 여러 유형의 데이터베이스를 사용하는 것이다.
- 여러 유형의 데이터베이스 : 리스트, 해시, 트리, 그래프 등
- NoSQL 등장은 빅데이터 시대에서 많은 양의 데이터를 효율적으로 처리하기 위함이다.
- NoSQL은 RDBMS의 독점적인 지위에 대한 현재 상황을 반발하는 정신을 담고 있다.

// NoSQL 특징

스키마	• 스키마를 강제하지 않는다. • 동적으로 스키마를 구현한다. ← 비정형 데이터 취급 가능
데이터 조직	• 트리, 그래프, 리스트, 해시 등의 다양한 방법 이용 • 다양한 데이터 모델을 사용한다.
분산처리	• 대부분 NoSQL DB는 분산처리 기능을 목적으로 개발되었다. • 빅데이터 취급 가능 - NoSQL이 등장한 이유라고도 함
조인	• 조인(join)은 관계 정의가 없으므로 불가능하다. • Join은 reference 같은 기능으로 비슷하게 구현은 가능
확장성	• 우수함 - 수평적 확장 • 여러 대의 컴퓨터에 데이터를 분산 저장하는 것!
가용성	• 우수함 - 궁극적 일관성을 이용 • 일관성이 데이터베이스의 절대적인 요소가 아니라는 주장!

SQL	• SQL은 **고정된 형식**의 **스키마**를 가진다. • 스키마는 엄격한 형식의 데이터 구조**(정형 데이터)**이다. • 엄격한 형식의 데이터 구조는 정형 데이터이다.

〈MongoDB〉

- MongoDB는 **NoSQL**로 **문서**(document) 지향 데이터베이스이다.
- 여기서, 문서 타입은 **XML, JSON, YAML**과 같은 데이터 타입이다.
- MongoDB는 고정된 스키마가 아니고, **JSON** 형태의 **동적 스키마**형 문서를 사용한다.
- MongoDB에서 가장 기본적인 데이터는 **문서**이다. 이는 RDBMS에서는 **행(row)**에 해당된다.
- MongoDB에서 **문서**(document)의 집합을 **컬렉션**(collection)이라 한다.
- **컬렉션**(collection)은 RDBMS에서는 **테이블**(table)에 해당된다.
- MongoDB에서 데이터베이스는 **컬렉션**(collection) 집합이다.
- MongoDB는 C++로 작성되었다.

3. 아파치 하둡(apache hadoop)

- hadoop = high-availability distributed object-oriented platform
- 하둡은 **빅데이터**를 처리할 수 있는 자바 소프트웨어 프레임워크이다.(프리웨어)
- 하둡은 빅데이터 처리에 필요한 비용 및 시간을 획기적으로 줄일 수 있게 도와준다.
- 하둡을 이용하면 **x86 서버**로 실시간 빅데이터 분석이 가능하다.

// 하둡은 공통 패키지로 구성되어 있다.

〈하둡 1.0〉

Data processing frameworks
맵리듀스 (MapReduce)
하둡분산파일시스템 (HDFS)

- 하둡의 필수 핵심 구성요소는 **분산파일시스템**과 **맵리듀스**이다.

// 하둡 분산 파일시스템(HDFS, Hadoop distributed file system)

- HDFS은 하둡 프레임워크를 위해 자바 언어로 작성된 분산 확장 파일시스템이다.
- HDFS은 여러 서버에 대용량 파일들을 나누어서 저장한다.(분산 저장)
- HDFS은 데이터들을 여러 서버에 중복해서 저장을 함으로써 데이터 안정성을 얻는다.
- 해서, 호스트에 RAID 저장장치를 사용하지 않아도 된다.

// 맵리듀스(MapReduce) : 맵 단계 + 리듀스 단계

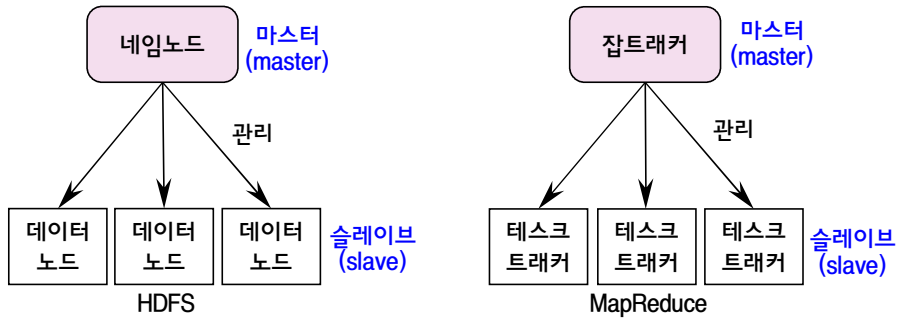
- Map : 흩어져 있는 데이터를 (key, value)의 형태로 관련 있는 데이터끼리 **묶는 작업**
- Reduce : Map 작업 결과에서 중복 데이터를 제거하고 원하는 데이터를 **추출**하는 작업

// 하둡의 HDFS는 마스터-슬레이브 구조

마스터	• 마스터로 지정된 장치(또는 프로세스)가 다른 장치(슬레이브)를 제어하는 모델이다. • 제어 방향은 항상 마스터에서 슬레이브로 흐른다.
슬레이브	• 슬레이브는 다른 장치(마스터)에 의해 제어되는 장치를 의미한다.

- 마스터(master)는 주인, 지배자, 슬레이브(slave)는 노예, 종속 장치를 의미한다.
- 인종차별 의미가 있어서 중립적인 단어로 전환하라는 문제가 제기됐다.
- 해서, 마스터는 메인(main), 프라이머리(primary) 등으로
- 슬레이브는 세컨더리(secondary)나 레플리카(replica) 등으로 대체하는 방안이 빠르게 확산하고 있다.

〈하둡 1.0〉



- HDFS는 **하나의 네임노드**와 **다수의 데이터노드**로 구성된다.
- **네임노드** : 메타데이터를 가지고 있다. **메타데이터**는 **파일 및 블록** 정보 등으로 구성되어 있다.
- **데이터노드** : 블록 단위로 나누어진 데이터가 **분산 저장**된다.
- 사용자는 네임노드를 이용하여 데이터를 쓰고, 읽을 수 있다.

데이터노드	• 데이타노드는 **파일**을 **저장**하는 역할을 한다. 파일은 **블록**단위로 저장된다. • **데이타노드**는 주기적으로 **네임노드**에 **하트비트**와 **블록리포트**를 전달한다. • **하트비트**는 데이타노드의 **동작여부**를 판단하는데 이용된다. • **네임노드**는 **하트비트** 전달이 없는 **데이터노드**를 **동작하지 않는 것**으로 판단하고 • 더 이상 데이터를 저장하지 **않도록 설정**한다. • **블록리포트**로 블록의 변경사항을 점검하고, **네임노드**의 **메타데이터**를 **갱신**한다.
메타데이터 관리	• **네임노드**의 주요 역할은 **메타데이터**와 **데이터노드**를 **관리**하는 것이다. • 메타데이터는 파일명, 파일크기, 파일생성시간, 파일접근권한, 파일소유자 및 그룹 소유자, 파일이 위치한 블록 정보 등으로 구성된다. • 각 데이터노드에서 전달하는 메타데이터를 받아서 전체 노드의 메타데이터 정보와 파일 정보를 묶어서 관리한다.
데이터노드 관리	• 네임노드는 데이터노드가 주기적으로 전달하는 **하트비트**(3초)와 **블록리포트**(6시간)를 이용하여 데이터노드의 동작상태, 블록상태를 관리한다. • 네임노드는 하트비트를 이용하여 데이터노드가 동작 중이라는 것을 알 수 있다. • **하트비트**가 도착하지 않는 데이터노드가 동작하지 않는 것으로 간주한다. • 더 이상 IO가 발생하지 않도록 조치한다. • **블록리포트**는 데이터노드에 저장된 블록 목록과 각 볼록이 로컬 디스크의 어디에 저장되어 있는지에 대한 정보를 가지고 있다. • **블록리포트**를 이용하여 HDFS에 저장된 파일에 대한 최신 정보를 유지한다.

// 맵리듀스(MapReduce)

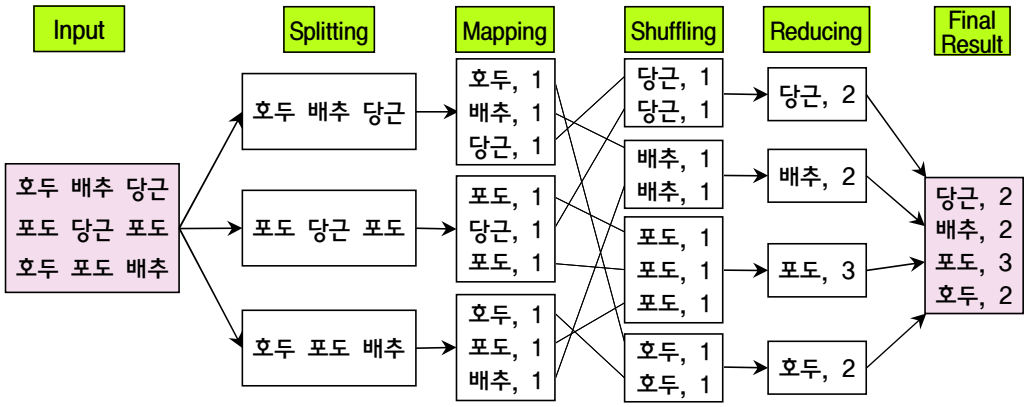
맵리듀스 **(MapReduce)**	• **구글**에서 2004년 발표한 소프트웨어 프레임워크 • 대용량 데이터를 처리를 위한 **분산** 프로그래밍 모델 • 타고난 **병행성**(병렬처리 지원)을 내포 • 누구든지 임의로 활용할 수 있는 **대규모** 데이터 분석 가능 • **흩어져** 있는 데이터를 (key, value)의 형태로 **묶고**(Map), • Sorting을 거쳐 • 데이터를 **뽑아내는**(Reduce) **분산처리 기술**과 관련 프레임워크를 의미

잡트래커 Job tracker	• 사용자로부터 Job을 요청 받고, Task Tracker에 **작업 할당**한다. • **잡트래커**는 전체 **진행 상황**을 관리한다.
테스크트래커 Task tracker	• Job Tracker로부터 할당 받은 작업을 Map-Reduce하여 결과 반환한다. • **테스크트래커**는 **실제 작업**을 처리한다.

• Tracker : **추적자, 사냥꾼**

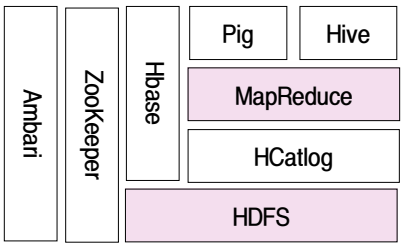
• **병렬처리**는 **잡트래커**(JobTracker)와 **태스크트래커**(TaskTracker)가 담당한다.

예제 어떤 교재에 포함된 단어 빈도수를 출력해주는 과정



Splitting	문자열 데이터를 라인별로 나눈다.
Mapping	라인별로 문자열을 입력한다. (key, value) 쌍의 형태로 묶는다.
Shuffling	같은 key를 가지는 데이터끼리 분류한다.
Reducing	각 key 별로 빈도수를 합산해서 출력한다.
Final Result	리듀스 메소드의 출력 데이터를 합쳐서 하둡 파일시스템에 저장한다.

〈하둡 2.0〉



하둡2.0 프레임워크

• 프레임워크 그림에서 **주키퍼**(사육사), **피그**(돼지), **하이브**(벌집) 같은 **동물이름**이 눈에 띈다

주키퍼 (Zookeeper)	**주키퍼**는 하둡의 분산상호조정서비스를 이용하여 분산환경에서 **노드**들 사이의 정보 공유, 락, 이벤트 등의 보조 기능을 제공한다. - **주키퍼**는 **조정자** 역할

• **하둡1.0**은 **네임노드**와 **세컨더리** 네임노드가 동시에 **다운**되면 시스템 전체 **장애**가 발생한다.
• 즉, **하둡1.0**은 SPOF(Single Point Of Failure) 문제를 가지고 있었다.**(병목현상)**
• **주키퍼**는 이 문제를 해결하기 위한 방법으로 **네임노드 고가용성**(HA)이 가능하도록 구성하였다.
• **네임노드**는 분산처리시스템에서 **master**(주인, 주된 역할)를 담당한다.
• **주키퍼 용도** : 설정 관리, 클러스터 관리, 리더 채택, 락 • 동기화 서비스 등

하이브 (Hive)	• 하이브는 페이스북이 개발하였다. • 하이브는 하둡 기반의 **데이터웨어하우징**용 솔루션이다.**(데이터 분석)** • 하이브는 SQL과 매우 유사한 HiveQL이라는 쿼리 언어를 제공한다. • HiveQL은 내부적으로 맵리듀스 잡으로 변환되어 실행된다.
피그(pig)	• 하이브와 함께 **데이터 분석**
Hbase	• 분산 데이터베이스
MapReduce	• 분산 데이터 병행처리
HCatlog	• 메타 데이터 관리
Ambari	• 작업흐름 관리

HDFS 고가용성 (high availability)	• **네임노드**에 문제가 발생하면 모든 작업이 **중지**된다. • 파일을 읽거나 쓸 수 없게 된다. • 하둡 v2에서 이 문제를 해결하기 위해서 **HDFS 고가용성**을 제공한다. • HDFS 고가용성은 이중화된 2대의 서버인 **액티브**(active) 네임노드와 **스탠바이**(standby) 네임노드를 이용하여 지원한다. • **액티브** 네임노드와 **스탠바이** 네임노드는 데이터 노드로부터 블록 리포트와 하트비트를 모두 받아서 **동일한 메타데이터**를 **유지**한다. • **액티브**에 문제가 발생하면 **스탠바이**가 **액티브 네임노드**로 동작한다.

// 하둡 버전별 요약

하둡1.0 (2011년)	• **자바** 프레임워크로 **빅데이터 분산저장**과 **병렬처리**를 목적으로 하둡이 탄생하였다. • 핵심 구성요소는 **하둡파일시스템**(HDFS)과 **맵리듀스**(MapReduce)이다. • **분산저장**은 **네임노드**(name node)와 **데이터노드**(data node)로 나누어 처리한다. • **네임노드**는 블록정보를 가지고 있는 메타데이터와 데이터 노드를 관리한다. • **데이터노드**는 데이터를 블록단위로 저장한다. • **병렬처리**는 **잡트래커**(JobTracker)와 **태스크트래커**(TaskTracker)가 담당한다. • **잡트래커**는 전체 **진행 상황**을 관리한다. • **테스크트래커**는 **실제 작업**을 처리한다.
하둡2.0 (2012년)	• **잡트래커**의 **병목현상**을 **제거**하기 위해 **얀**(yarn) 아키텍처를 도입하였다. • **얀**은 잡트래커가 혼자서 처리하던 일을 나누어서 **담당**하도록 하였다. • **자원관리** : 리소스 매니저(resource manager)와 노드 매니저(node manager) • **라이프 사이클관리** : 애플리케이션 마스터(application master) • **작업처리** : 컨테이너(container)가 담당하도록 하였다. • **컨테이너**는 **얀**의 **작업처리** 단위이다. • HDFS **고가용성**을 제공한다.
하둡3.0 (2017년)	• **이레이저 코딩**(erasure coding)을 도입하였다. • **이레이저 코딩**은 RAID와 비슷한 기술이다. • **이레이저 코딩**은 RAID처럼 데이터가 **손실**되면, 별도의 **패리티**로 **복구**하는 기술이다. • **이레이저 코딩** 도입으로 HDFS **사용량**을 **감소**시켰다. • **이레이저 코딩**은 RAID처럼 모든 작업을 소프트웨어로 처리한다. • **소프트웨어 처리**는 CPU **오버헤드**와 **지연시간**이 발생될 수 있다. • **2개** 이상의 **네임노드**를 Running 상태(Active/Passive)로 운영할 수 있다. • **스탠바이 노드**도 여러 개 지원한다. • 하둡3.0부터는 반드시 Java 8 이상의 버전을 사용해야 한다

기출문제 분석

1. 정책 수립에 있어 중요성이 커지고 있는 빅데이터에 대한 설명으로 가장 옳지 않은 것은? [2018년 서울 9급]

① 디지털 환경에서 생성되는 데이터로 규모가 방대하고, 생성주기가 길며, 형태가 다양하다.

② 하둡(Hadoop)과 같은 오픈 소스 소프트웨어 시스템을 빅데이터 처리에 이용하는 것이 가능하다.

③ 보건, 금융과 같은 분야의 빅데이터는 사회적으로 유용한 정보이나 데이터 활용 측면에서 프라이버시 침해에 대한 대비가 필요하다.

④ 구글 및 페이스북, 아마존의 경우 이용자의 성향과 검색 패턴, 구매패턴을 분석해 맞춤형 광고를 제공하는 등 빅데이터의 활용을 증대시키고 있다.

♤ **빅데이터 – 생성주기**

- 디지털 환경에서 생성되는 데이터로 규모가 방대하고, 생성주기가 길며, 형태가 다양하다.**(×)**
  → 빅데이터는 기존 데이터에 비해 생성주기가 짧다.

정답 : ①

2. 빅데이터에 대한 설명으로 옳지 않은 것은? [2017년 지방 9급]

① 빅데이터의 특성을 나타내는 3v는 규모(volume), 속도 (velocity), 가상화(virtualization)를 의미한다.

② 빅데이터는 그림, 영상 등의 비정형 데이터를 포함한다.

③ 자연어 처리는 빅데이터 분석기술 중의 하나이다.

④ 시각화(visualization)는 데이터 분석 결과를 쉽게 이해할 수 있도록 표현하는 기술이다.

♤ **빅데이터 3V 모델**

- 데이터 규모(volume)
- 데이터 입출력 속도(velocity)
- 데이터 종류의 다양성(variety)

정답 : ①

3. 빅데이터 특징과 처리에 대한 설명으로 옳지 않은 것을 <보기>에서 모두 고른 것은? [2018년 서울 7급]

----<보기>----------

ㄱ. NoSQL 데이터베이스는 관계형 데이터베이스 보다 더 강한 일관성 모델을 제공한다.

ㄴ. MapReduce는 여러 노드(컴퓨터)에 분산된 데이터를 처리하기 위해 데이터를 한 노드로 집중시켜 처리하고 다시 분산 저장하는 과정을 지원한다.

ㄷ. 빅데이터 특징을 정의하는 3V는 데이터의 규모(volume), 다양성(variety), 처리속도(velocity)를 의미하며, 빅데이터의 유형은 정형, 반정형, 비정형의 데이터를 모두 포함한다.

ㄹ. 텍스트 마이닝(text mining)은 구조화된 데이터로부터 유용한 정보를 추출하는 기술이다.

----

① ㄱ, ㄷ ② ㄴ, ㄷ

③ ㄱ, ㄴ, ㄹ ④ ㄴ, ㄷ, ㄹ

♤ **빅데이터**

ㄱ. NoSQL 데이터베이스는 관계형 데이터베이스 보다 더 강한 일관성 모델을 제공한다.**(×)**

→ NoSQL은 제품에 따라 그 특성 차이가 매우 크다.

→ NoSQL 데이터베이스는 다양한 데이터 모델을 사용한다.

→ NoSQL 데이터베이스는 관계형 데이터베이스 보다 덜 제한적인 일관성 모델을 이용하는 데이터의 저장 및 검색을 위한 매커니즘을 제공한다.

ㄴ. MapReduce는 여러 노드(컴퓨터)에 분산된 데이터를 처리하기 위해 데이터를 한 노드로 집중시켜 처리하고 다시 분산 저장하는 과정을 지원한다.**(×)**

→ 데이터 분산처리는 하둡 분산 파일시스템(HDFS, Hadoop distributed file system)이다.

→ HDFS은 여러 서버에 대용량 파일들을 나누어서 저장한다.(분산 저장)

→ 맵리듀스(MapReduce) : 맵 단계 + 리듀스 단계

- Map : 흩어져 있는 데이터를 Key, Value의 형태로 관련 있는 데이터끼리 묶는 작업
- Reduce : Map 작업 결과에서 중복 데이터를 제거하고 원하는 데이터를 추출하는 작업

ㄹ. 텍스트 마이닝(text mining)은 구조화된 데이터로부터 유용한 정보를 추출하는 기술이다.**(×)**

→ 텍스트 마이닝은 비/반정형 데이터로부터 유용한 정보를 추출하는 기술이다.

→ 데이터 마이닝은 정형화된 데이터로부터 유용한 정보를 추출하는 기술이다.

→ 일반적으로, 디지털 정보의 대부분은 비정형 데이터이다.

정답 : ③

4. 다음 중 SQL과 NoSQL의 차이점에 대한 설명으로 가장 옳지 않은 것은? [2022년 군무원 7급]

① NoSQL은 key-value, document, wide-column, graph 등의 방식으로 데이터를 저장하고, 관계형 데이터베이스는 SQL을 이용해서 데이터를 테이블에 저장한다.

② SQL을 사용하려면 고정된 형식의 스키마가 필요하고, NoSQL은 관계형 데이터베이스보다 동적으로 스키마의 형태를 관리할 수 있다.

③ 관계형 데이터베이스는 테이블의 형식과 테이블 간의 관계에 맞춰 데이터를 요청해야 하므로 SQL과 같이 구조화된 쿼리 언어를 사용하고, 비관계형 데이터베이스의 쿼리는 구조화되지 않은 쿼리 언어로도 데이터 요청이 가능하다.

④ SQL 기반의 관계형 데이터베이스는 수평적으로 확장하고, NoSQL로 구성된 데이터베이스는 수직적으로 확장한다.

♤ **SQL과 NoSQL 차이점**

// 데이터베이스 확장성(scalability)

- Scalability는 사용자 수의 증대에 유연하게 대응할 수 있는 정도를 의미한다.
- Scalability는 처리할 작업량이 늘 때마다 늘어나는 요구에 맞춰 크기를 확장시킬 수 있는 능력
- Scalability는 시스템이나 네트워크 등에서 크기나 기능을 확장시킬 수 있는 능력을 말한다.
- 데이터베이스 확장은 크게 2가지, **수직적**과 **수평적**으로 구별된다.

수직적 확장 **(vertical scalability)**	• 수직적 확장은 CPU나 RAM 같은 하드웨어를 **추가** 또는 **교체**하는 것 • 수직적 확장은 전체적인 성능을 향상시키는 것을 의미한다. • 수직적 확장은 단순하게 데이터베이스 서버의 성능을 향상시킨다. • 소프트웨어 설계나 구조를 변경할 필요가 없다. • 일반적으로 SQL은 **수직적** 확장을 한다. • SQL은 **샤딩**을 통해서 수평적 확장을 **흉내**낼 수 있지만 구현이 어렵다. • 샤딩(sharding)은 하나의 거대한 데이터베이스나 네트워크 시스템을 다수의 작은 조각으로 나누어 **분산 저장**하여 관리하는 것을 말한다.
수평적 확장 **(horizontal scalability)**	• 수평적 확장은 더 많은 **서버**를 **추가**하는 것이다. • 추가된 서버에 데이터베이스를 전체적으로 분산시키는 것이다. • 하나의 데이터베이스는 여러 호스트에서 작동한다. • NoSQL은 **수평적** 확장을 한다. • **비관계형** 데이터베이스는 기본적으로 **수평적** 확장을 **지원**한다.

정답 : ④

5. NoSQL에 대한 설명으로 옳지 않은 것은? [2023년 국가 7급]

① NoSQL 시스템으로 Ingres 등이 있다.
② 가용성을 높이기 위해 일치성을 약하게 보장하는 궁극적 일치성(eventual consistency)을 제공하는 경우가 많다.
③ 미리 정의된 스키마를 사용하지 않아도 된다.
④ 다수의 컴퓨터에 데이터를 분산, 저장, 처리하는 것이 가능한 데이터베이스 시스템이다.

♤ NoSQL

• NoSQL 시스템으로 Ingres 등이 있다.**(×)**
→ Ingres는 NoSQL 시스템이 아니다.

// 인그레스 데이터베이스(Ingres Database)
• 인그레스 데이터베이스는 **규모**가 **큰** 상업용 및 정부용으로 사용되도록 고안된 **사유** SQL **관계형 데이터베이스 관리시스템**이다.
• 인그레스 데이터베이스는 **사유** 소프트웨어(私有, proprietary software)이다.
• **사유 소프트웨어**는 저작권 소유자의 예외적 법적 권한 하에 허가된 소프트웨어이다.
• **사유 소프트웨어** 반대말은 오픈소스 소프트웨어이다.

// 궁극적 일치성(eventual consistency)
• 먼저, 궁극적(eventual)은 **최종적**이라는 의미이다.
• NoSQL 데이터베이스는 애초부터 '궁극적 일치성(일관성)'이라는 개념을 사용했다.
• 클러스터 내의 한 서버의 데이터베이스에 **새로 작성한 내용**을 **즉시 다른 서버**로부터 읽으면
• 방금 작성한 서버로부터 읽는 것과는 같은 결과를 얻지 못할 수도 있다는 의미이다.
• 하지만, **시간이 조금 지나면** 새로운 데이터가 클러스터 내 모든 서버에 복제되고 궁극적으로 일관성을 갖게 된다.
• NoSQL 데이터베이스는 기본적으로 분산 데이터베이스이다.
• 궁극적 일치성은 **가용성**을 높이기 위해 일치성을 **약하게** 보장하는 것이다.

// 새로운 SQL(NewSQL)
• SQL과 NoSQL의 장점 결합한 SQL이다.
• NewSQL은 향상된 기능과 기능으로 SQL을 지원한다.
• NewSQL은 관계형 데이터베이스이지만 순수하지는 않다.
• NewSQL은 스키마 수정 및 스키마가 없다.
• 예 : 바퀴벌레 DB - 바퀴벌레처럼 강력한 DB?

정답 : ①

6. NOSQL에 대한 설명으로 옳지 않은 것은? [2021년 국가 7급]

① NOSQL은 샤딩(sharding)을 지원한다.
② BigTable, Cassandra 등이 대표적인 NOSQL이다.
③ NOSQL은 RDBMS와 같이 스키마(schema)를 필요로 한다.
④ NOSQL은 가용성(availability)과 확장성(scalability)을 중요시 한다.

♤ NOSQL

• **NOSQL**은 RDBMS와 같이 **스키마**(schema)를 필요로 한다.**(×)**
→ **NOSQL**은 **스키마**를 강제 적용하지 않는다. 데이터 관계와 정해진 규격(table 정의)이 없다.

// 샤딩(sharding)
• 샤딩은 하나의 거대한 데이터베이스나 네트워크 시스템을 여러 개의 작은 조각으로 나누어 분산 저장하여 관리하는 것을 말한다.
• 영어 **shard**는 **조각**이라는 의미이다.

정답 : ③

7. 관계 데이터베이스 관리 시스템(RDBMS)과 NoSQL에 대한 설명으로 옳은 것은? [2022년 국가 7급]

① RDBMS는 NoSQL보다 약한 스키마를 요구한다.
② NoSQL은 RDBMS보다 엄격한 일관성 모델을 보장한다.
③ NoSQL은 RDBMS보다 정형 데이터를 저장하기에 적합하다.
④ NoSQL 데이터 모델로 키-값(key-value), 문서 기반(document-based), 그래프 기반(graph-based) 모델이 있다.

♤ RDBMS와 NoSQL

---

① RDBMS는 NoSQL보다 **약한 스키마**를 요구한다.**(×)**
→ RDBMS는 NoSQL보다 **강한 스키마**를 요구한다.
② **NoSQL**은 RDBMS보다 **엄격한 일관성 모델**을 보장한다.**(×)**
→ **RDBMS**가 NoSQL보다 **엄격한 일관성 모델**을 보장한다.
③ **NoSQL**은 RDBMS보다 **정형 데이터**를 저장하기에 적합하다.**(×)**
→ **RDBMS**가 NoSQL보다 **정형 데이터**를 저장하기에 적합하다.

정답 : ④

8. 다음 중 NoSQL 시스템의 특징에 대한 설명으로 가장 옳지 않은 것은? [2022년 군무원 7급]

① JSON이나 XML 형식을 갖는 반정형(semi-structured) 문서를 저장할 때 많이 사용된다.
② 고성능의 데이터 액세스를 위하여 파일 분할을 최소화한다.
③ 가용성을 높이기 위하여 데이터를 여러 사이트에 중복해서 저장한다.
④ 관계 데이터베이스보다 질의 처리 기능이 상대적으로 단순하다.

♤ NoSQL 시스템

• 고성능의 데이터 액세스를 위하여 **파일 분할**을 **최소화**한다.(×)
→ 대부분 NoSQL DB는 **분산처리** 기능을 목적으로 개발되었다.
→ **분산처리** : 데이터를 작게 **분할**하여 처리한다.

정답 : ②

9. 빅데이터에 대한 설명으로 옳은 것은? [2017년 서울 9급]

① 빅데이터는 정형데이터로만 구성되며, 소셜 미디어 데이터는 해당되지 않는다.
② 빅데이터를 구현하기 위한 대표적인 프레임워크는 하둡이 있으며, 하둡의 필수 핵심 구성요소는 맵리듀스(MapReduce)와 하둡분산파일시스템(Hadoop Distributed File System)이다.
③ 빅데이터 처리과정은 크게 수집→저장→처리→시각화(표현)→분석 순서대로 수행된다.
④ NoSQL은 관계 데이터 모델을 사용하는 RDBMS 중 하나이다.

♤ 빅데이터

---

① 빅데이터는 정형데이터로만 구성되며, 소셜 미디어 데이터는 해당되지 않는다.(×)
→ 빅데이터는 대량의 정형 또는 비정형 데이터를 포함한다.
③ 빅데이터 처리과정은 크게 수집→저장→처리→시각화(표현)→분석 순서대로 수행된다.(×)
→ 빅데이터 처리과정 : 수집→저장→처리→분석→시각화(표현)→이용→폐기
④ NoSQL은 관계 데이터 모델을 사용하는 RDBMS 중 하나이다.(×)
→ NoSQL은 관계형 데이터베이스 RDBMS가 아닌 다른 형태의 데이터 저장 기술이다.
→ NoSQL은 "Not Only SQL"라고도 한다.
→ NoSQL 등장은 빅데이터 시대에서 많은 양의 데이터를 효율적으로 처리하기 위함이다.
→ NoSQL은 SQL만을 사용하지 않는 DBMS을 지칭하는 단어이다.

정답 : ②

10. 빅데이터(big data)에 대한 설명으로 옳지 않은 것은? [2019년 국가 7급]

① 디지털 환경에서 생성되는 데이터로 규모가 방대하고, 정형, 반정형, 비정형 등 다양한 형태의 데이터를 포함한다.

② NoSQL 시스템은 반구조적이고 자기 기술적인 데이터를 허용하므로 대개는 스키마를 요구하지 않는다.

③ NoSQL의 키-값(key-value) 데이터 모델은 키와 값의 쌍으로 저장하며, 값은 이미지나 동영상 등 다양한 형태의 데이터가 될 수 있다.

④ 빅데이터 분석 과정에서 추출된 정보를 시각화하는 기술로 Hadoop의 맵 리듀스(MapReduce)를 사용한다.

♤ **빅데이터**

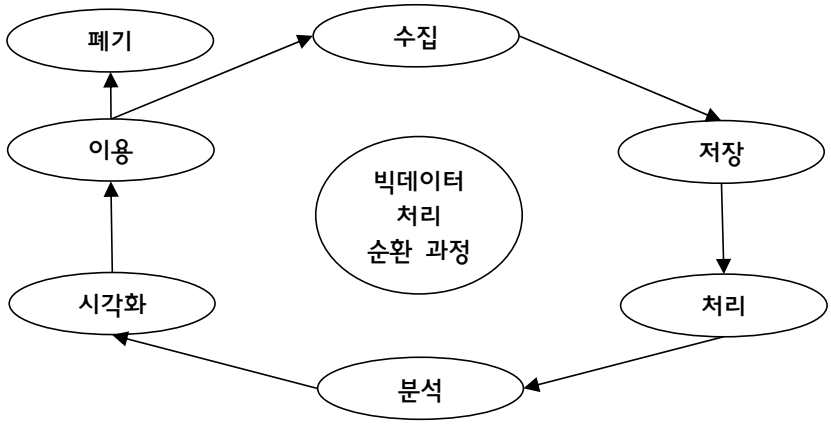
④ 빅데이터 분석 과정에서 추출된 정보를 시각화하는 기술로 Hadoop의 맵 리듀스를 사용한다.(×)
→ 맵 리듀스는 데이터를 추출하기 위한 것이다.

// 맵리듀스(MapReduce) : 맵 단계 + 리듀스 단계

- Map : 흩어져 있는 데이터를 Key, Value의 형태로 관련 있는 데이터끼리 묶는 작업
- Reduce : Map 작업 결과에서 중복 데이터를 제거하고 원하는 데이터를 추출하는 작업

// 빅데이터 시각화

- 시각화는 데이터 이해 및 탐색을 쉽게 제공하기 위한 것이다.
- 시각화는 공간에 데이터를 배치하여 사람이 쉽게 인지할 수 있도록 하는 것이다.



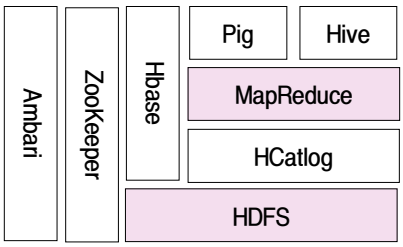
정답 : ④

11. 빅데이터(big data)를 저장, 처리 및 관리하기 위해 사용되는 기술을 <보기>에서 모두 고른 것은? [2021년 서울 7급]

----<보기>----------
ㄱ. HDFS(Hadoop Distributed File System)
ㄴ. MapReduce
ㄷ. ZooKeeper
---

① ㄱ ② ㄱ, ㄷ ③ ㄴ, ㄷ ④ ㄱ, ㄴ, ㄷ

♤ **빅데이터 - 하둡2.0 프레임워크**



하둡2.0 프레임워크

• 프레임워크 그림에서 **주키퍼**(사육사), **피그**(돼지), **하이브**(벌집) 같은 **동물이름**이 눈에 띈다

주키퍼 (Zookeeper)	**주키퍼**는 하둡의 분산상호조정서비스를 이용하여 분산환경에서 **노드**들 사이의 정보 공유, 락, 이벤트 등의 보조 기능을 제공한다. - **주키퍼**는 **조정자** 역할

• **하둡1.0**은 **네임노드**와 **세컨더리** 네임노드가 동시에 **다운**되면 시스템 전체 **장애**가 발생한다.
• 즉, **하둡1.0**은 SPOF(Single Point Of Failure) 문제를 가지고 있었다.**(병목현상)**
• **주키퍼**는 이 문제를 해결하기 위한 방법으로 **네임노드 고가용성**(HA)이 가능하도록 구성하였다.
• **주키퍼 용도** : 설정 관리, 클러스터 관리, 리더 채택, 락 • 동기화 서비스 등

하이브 (Hive)	• 하이브는 페이스북이 개발하였다. • 하이브는 하둡 기반의 **데이터웨어하우징**용 솔루션이다.**(데이터 분석)** • 하이브는 SQL과 매우 유사한 HiveQL이라는 쿼리 언어를 제공한다. • HiveQL은 내부적으로 맵리듀스 잡으로 변환되어 실행된다.
피그(pig)	• 하이브와 함께 **데이터 분석**
Hbase	• 분산 데이터베이스
MapReduce	• 분산 데이터 병행처리
HCatlog	• 메타 데이터 관리
Amban	• 작업흐름 관리

정답 : ④