

---

# Emerging Technologies

---

컴퓨터개론

(Introduction to Computer Systems)

GEN1030

# 산업혁명

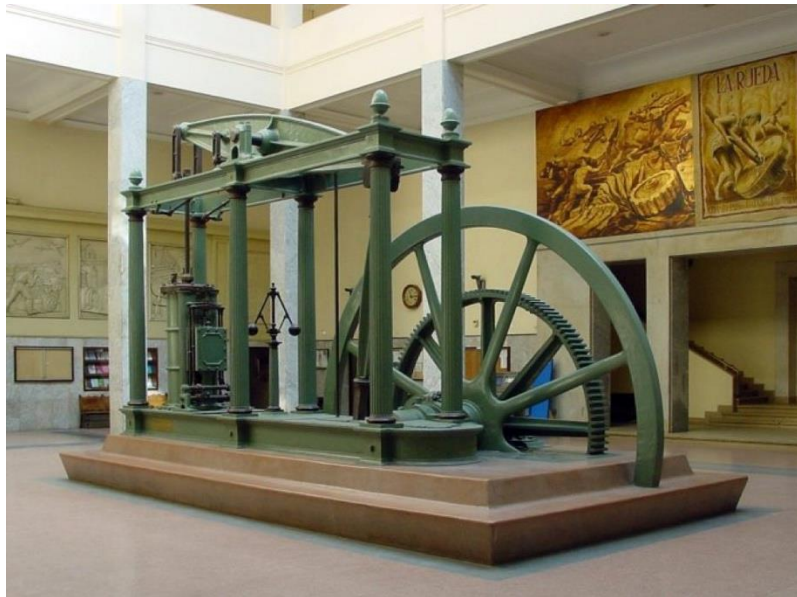
# 산업혁명

- Industrial revolution
  - 산업에서의 근본적인 변화를 가져온 시기나 일련의 과정
- 발달 과정
  - 제1차 산업혁명
  - 제2차 산업혁명
  - 제3차 산업혁명
  - 제4차 산업혁명

# 산업혁명

- 제1차 산업혁명

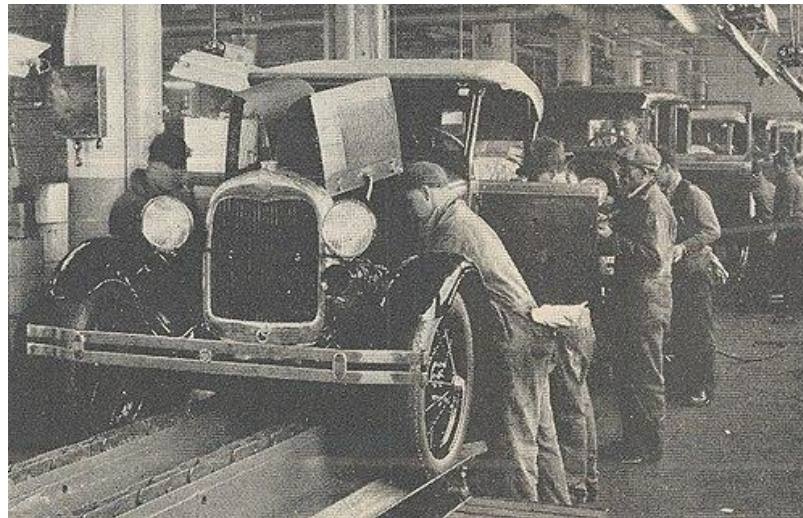
- 18세기 후반 영국에서 발명 된 증기기관(steam engine)으로 시작
- 적은 노동력으로 생산력이 증가하는 기계화 혁명
- 증기기관차, 증기선 등의 출현 → 이동거리 비약적으로 발달
- 농업중심 사회 → 공업중심 사회



<와트의 증기기관>

# 산업혁명

- 제2차 산업혁명
  - 전기에너지의 대중화, 석유/화학/철강 분야의 기술 혁신
  - 내연기관과 자동차 발전
  - 공장의 생산조립 라인인 컨베이어 벨트(conveyor belt) 도입
    - 대량 생산(mass production)



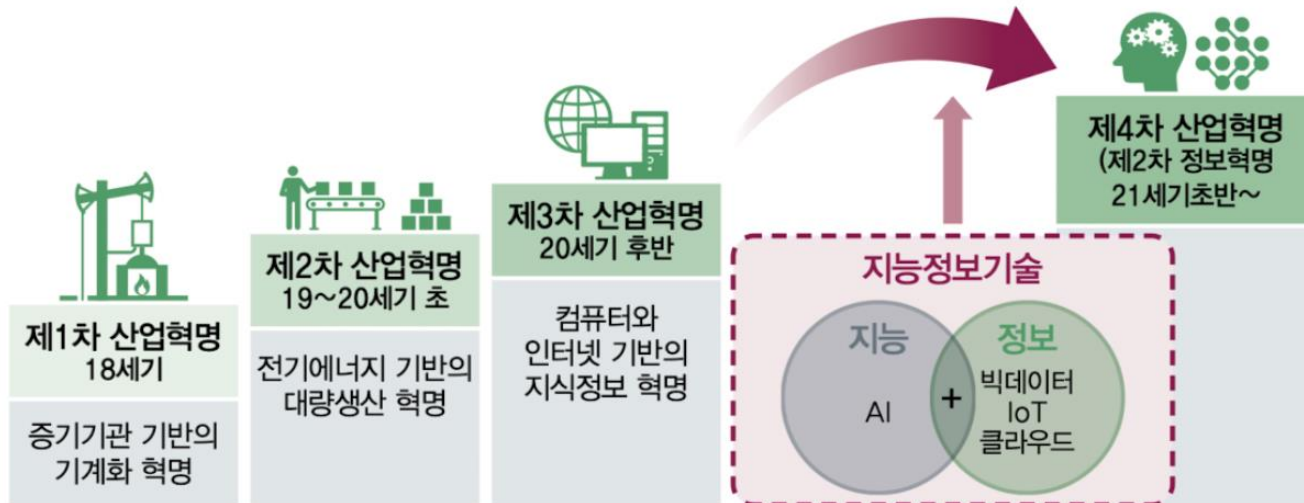
<Ford 자동차 조립 라인>



# 산업혁명

## • 제4차 산업혁명

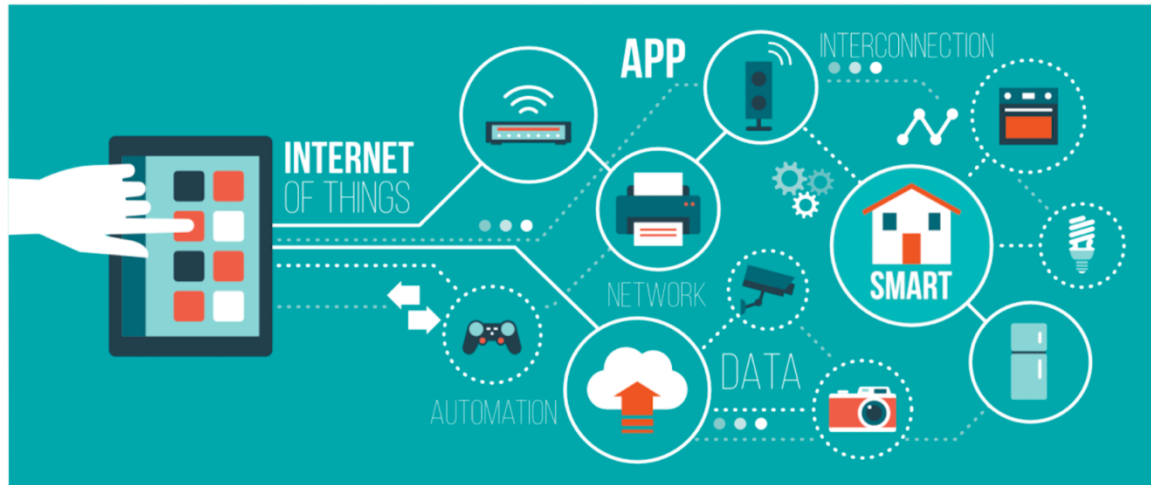
- 인공지능(AI), 빅데이터(Big Data), 사물인터넷(IoT) 등의 발전
- 초지능화(Hyper Intelligence) 사회
  - 인간의 지능과 유사, 사고 및 결정을 보조
- 초연결(Hyper Connectivity) 사회
  - 모든 사물이 연결되어 정보가 생성, 수집되고 다시 재가공, 공유, 활용되는 사회



# IoT

# Internet of Things (IoT)

- 사물인터넷



- 일상의 모든 기기·사물이 인터넷에 연결되어 데이터를 주고받는 것
- 단순히 기기를 연결하는 것을 넘어, 수집된 데이터로 자동 제어, 최적화까지
- 물리 세계 <-> 디지털 세계
- 2025년 기준 200억개 이상, 2030년에는 400억 개 이상의 기기가 연결 될 것으로 예상 (source: IoT Analytics)

# IoT 활용

- 스마트 홈(Smart home)
  - 가정 내 기기를 인터넷으로 연결해 원격 제어, 자동화를 실현
  - 음성 어시스턴트로 기기 제어
  - 스마트 플러그, 도어락, CCTV 등
- 스마트 팩토리(Smart factory)
  - 생산 설비, 공정 전체에서 IoT 접목해 자동화, 최적화, 자율 운영
  - 사람이 조정(기존 공장) → 기계가 스스로 상태를 감지하고 판단
- 스마트 농업(Smart farm)
  - 농업 과정에서 IoT 적용해 데이터 기반으로 생산성과 효율을 높이는 농업 방식
  - 경험과 직관 의존(기존 농업) → 실시간 데이터로 의사 결정
  - 센서(ex. 토양 수분, pH, 온도, 일조량) 사용해 데이터 수집, 작물별 최적 환경 자동 유지 등

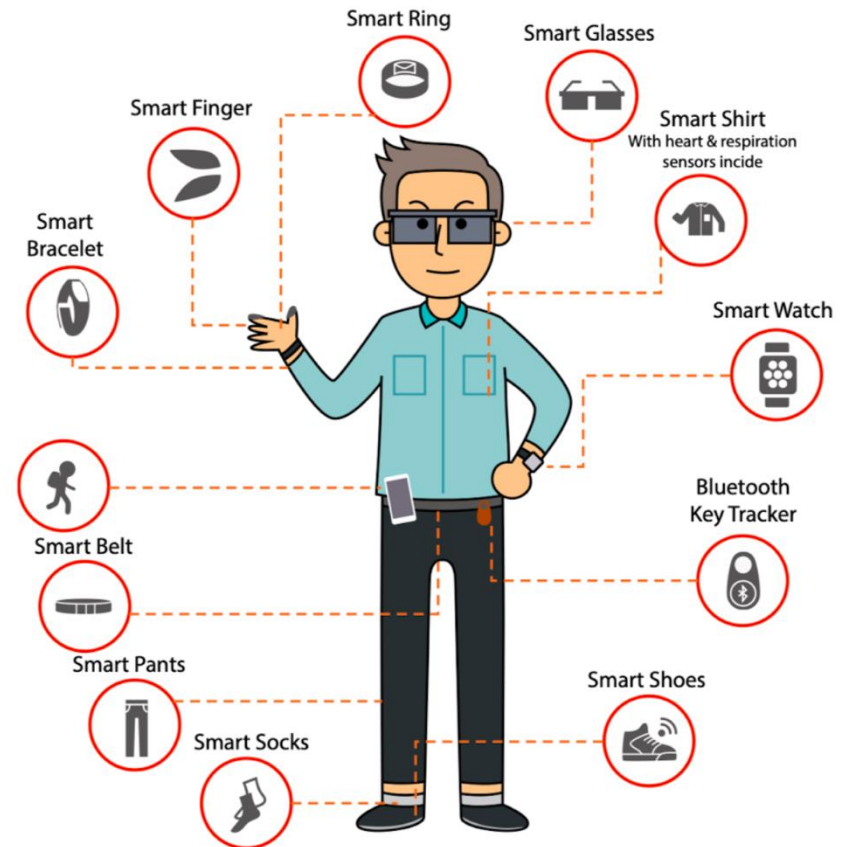
# Wearable IoT

- Wearable device

- 사용자가 이동, 활동 중에도 자유롭게 사용할 수 있도록 사람의 신체에 착용할 수 있는 컴퓨터 기기
- 스마트 워치: 심박수, 혈중산소, 수면 패턴 등 측정
- 스마트 의류: 자세 교정, 재활 등
- 스마트 안경(AR 글래스): 현실 위에 정보 오버레이

- Wearable Internet

- 웨어러블 디바이스가 다양한 센서와 인터넷에 연결

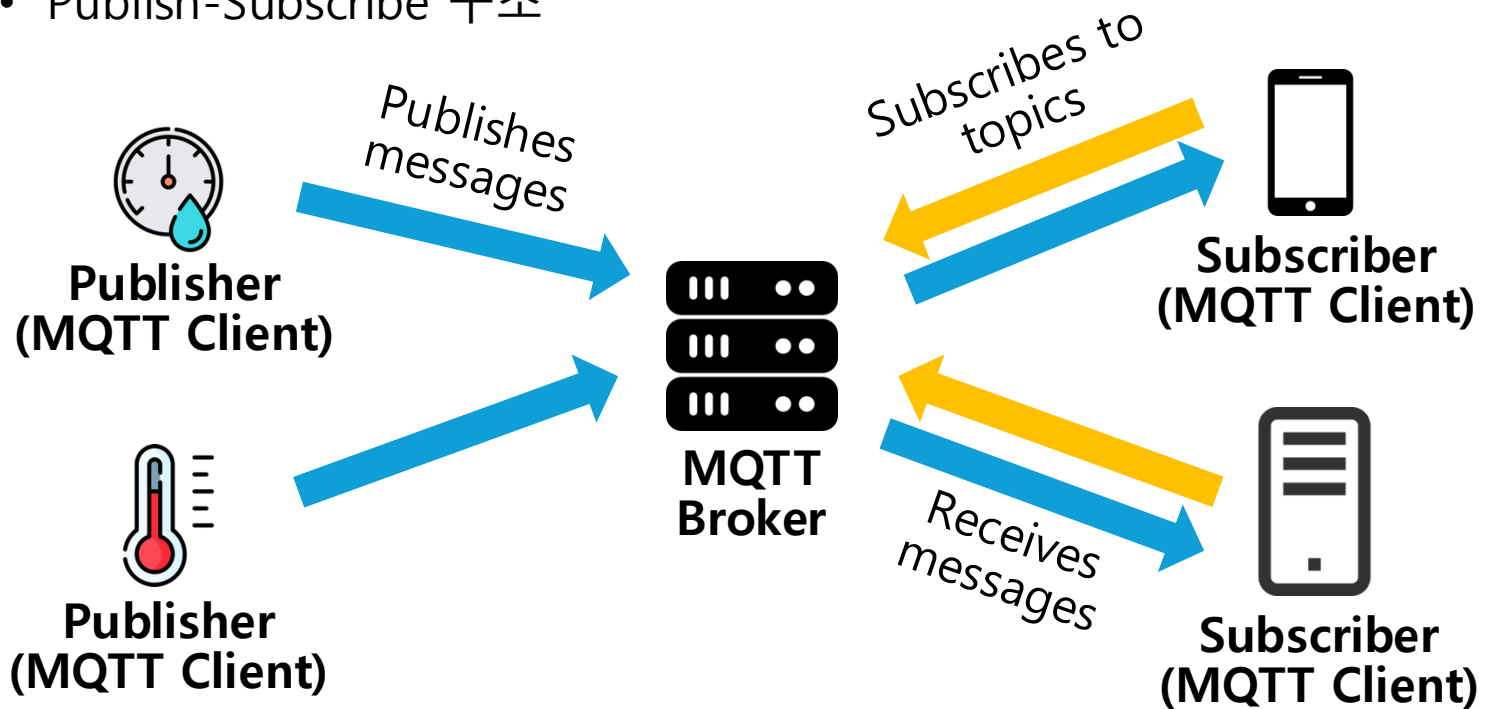


# IoT 메시징 프로토콜

- MQTT (Message Queuing Telemetry Transport) 프로토콜
  - IoT 메시지 프로토콜로 널리 사용
  - 가볍고 전력소모 적음 → 배터리 구동 센서, 저대역폭 환경 최적
  - 스마트홈, 산업용 IoT 등에서 광범위하게 사용
- Publish-Subscribe 구조
  - 기기들이 직접 통신하지 않고 브로커(Broker) 를 중간에 두고 통신
    - cf) HTTP 방식: 서버-클라이언트 구조. 클라이언트가 서버에 요청 보내고 서버가 응답
  - Publisher: 데이터 생성해서 브로커에게 전송(ex. 온도 센서)
  - Broker: 메시지를 받아서 관리하고 배포하는 중간 서버
    - 한 브로커에 수만 개 기기가 동시에 연결 가능(대규모 IoT 시스템)
  - Subscriber: 브로커로부터 메시지 받는 기기(ex. 앱, 서버)
    - 특정 토픽(Topic) 을 구독하는 방식

# IoT 메시징 프로토콜

- MQTT (Message Queuing Telemetry Transport) 프로토콜
  - IoT 메시지 프로토콜로 널리 사용
  - 가볍고 전력소모 적음 → 배터리 구동 센서, 저대역폭 환경 최적
  - 스마트홈, 산업용 IoT 등에서 광범위하게 사용
- Publish-Subscribe 구조



# IoT 메시징 프로토콜

- CoAP (Constrained Application Protocol)
  - HTTP의 기능과 구조는 유지하고 자원이 제한된 기기에서도 동작하도록 경량화된 프로토콜
    - HTTP는 사양이 낮은 기기(ex. 초소형 센서)에서는 동작이 어려움
  - GET, POST, PUT, DELETE 메서드 동일하게 사용
  - UDP 기반
- Request-Response 구조
- Observe 기능
  - 클라이언트가 서버의 리소스를 구독하면, 해당 리소스의 값이 바뀔 때마다 자동으로 알림 수신 (MQTT subscribe와 유사)

# Cloud Computing

# 클라우드 컴퓨팅

- 서버인 클라우드에 하드웨어와 소프트웨어, 개발 플랫폼 등을 두고 인터넷을 통해 필요에 따라 접속하여 필요한 자원을 사용하는 서비스
  - 정보기술 자원을 구매하지 않고 필요한 자원을 사용한 만큼 비용을 지불
- 정보기술 자원의 활용 방식: 소유 → 임대
  - 기존
    - 직접 서버 구매 → 데이터센터 구축 → 전문 인력 운영
    - 초기 비용이 높고, 트래픽 폭증 등의 상황에 유연하게 대처하기 어려울 수 있음
- 특성
  - 온디맨드(On-demand): 필요한 경우 즉시 자원 확보
  - 탄력성(Elasticity): 트래픽 상황에 따라 자원을 유연하게 할당
  - Pay-as-You-Go (PAYG): 사용한 만큼만 지불

# 클라우드 컴퓨팅





# 클라우드 서비스 유형

- IaaS (Infrastructure as a Service)
  - 서버, 저장장치, 네트워크 등 물리적 인프라만(infrastructure) 제공
  - ex) AWS EC2, Google Compute Engine
- PaaS (Platform as a Service)
  - 앱 개발 및 배포에 필요한 도구/환경까지 제공
  - ex) Google App Engine 등
- SaaS (Software as a Service)
  - 최종 사용자를 위한 완성된 소프트웨어를 바로 서비스
  - ex) Microsoft 365, Slack 등

# 클라우드 서비스 유형

On-site	IaaS	PaaS	SaaS
Applications	Applications	Applications	Applications
Data	Data	Data	Data
Runtime	Runtime	Runtime	Runtime
Middleware	Middleware	Middleware	Middleware
O/S	O/S	O/S	O/S
Virtualization	Virtualization	Virtualization	Virtualization
Servers	Servers	Servers	Servers
Storage	Storage	Storage	Storage
Networking	Networking	Networking	Networking










 You manage  
 Service provider manages

source: Red Hat

# 주요 클라우드 서비스



- AWS (Amazon Web Services)
  - 컴퓨팅, 스토리지, 데이터베이스 등 다양한 서비스 제공
  - 전세계 광범위한 지역(30개 이상)에 가용 영역 운영
  - EC2 (Elastic Compute Cloud)
    - AWS 클라우드에서 온디맨드 확장 가능 컴퓨팅 제공

 Analytics	 Application integration	 Blockchain	 Business applications	 Cloud Financial Management	 Compute
 Customer enablement	 Containers	 Databases	 Developer tools	 End user computing	 Front-end web and mobile
 Game tech	 Internet of Things (IoT)	 Machine Learning (ML) and Artificial Intelligence (AI)	 Management and governance	 Media	 Migration and transfer
 Networking and content delivery	 Quantum technologies	 Satellite	 Security, identity, and compliance	 Storage	

# 주요 클라우드 서비스

- Microsoft Azure

- Microsoft 생태계와 강력한 연동이 장점
- 전세계 60개 이상 지역에 운영
- Azure Virtual Machines
  - AWS EC2와 유사한 가상 서버



- Google Cloud

- Google이 자체 서비스 운영하며 쌓은 기술 기반
- Compute Engine
  - AWS EC2에 해당하는 가상 서버
- BigQuery
  - 대용량(페타 바이트급)의 데이터 저장 및 분석용 클라우드 서비스



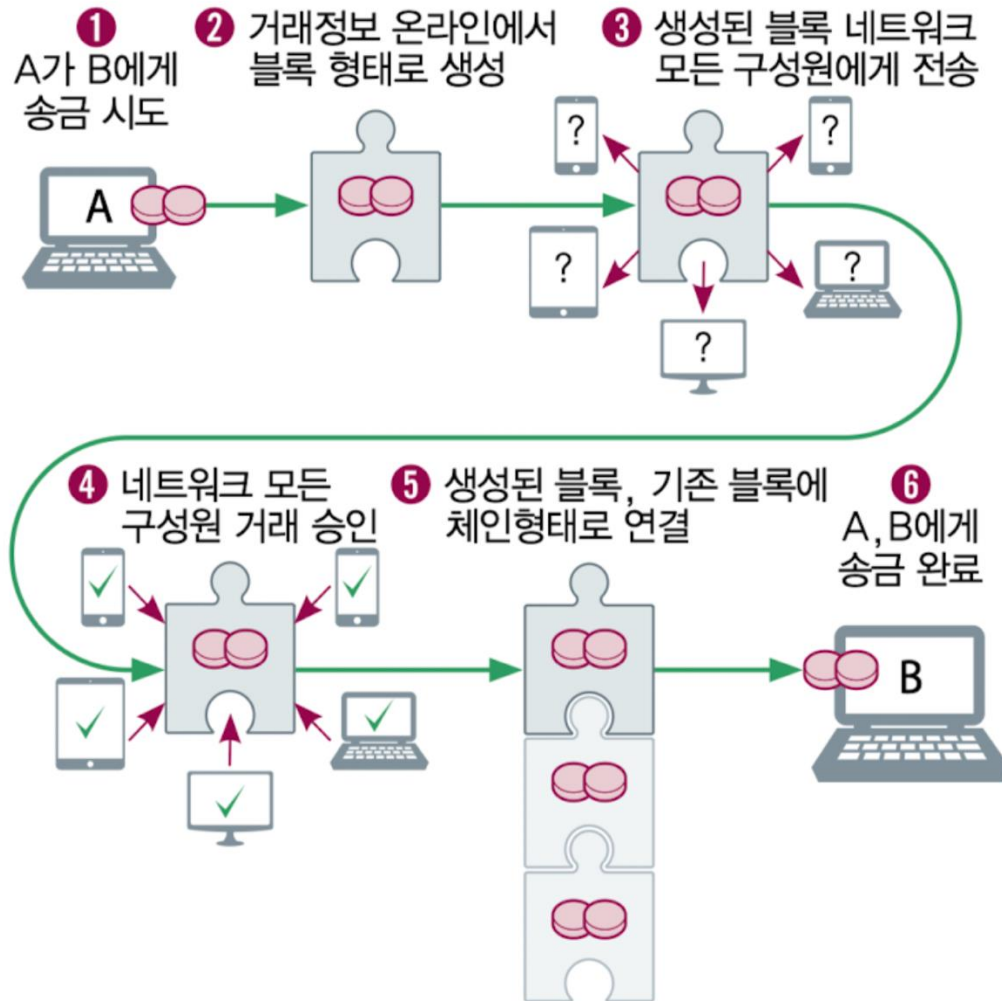
# Blockchain

# 블록체인(Blockchain)

- 거래 정보를 중앙 서버에 저장하지 않고 네트워크에 연결된 모든 참여자가 분산하여 공동으로 기록하고 관리하는 데이터 분산 처리 기술
  - 분산 장부(Distributed Ledger) 기술
- 블록(Block)
  - 일정 기간의 거래 기록 묶음 + 이전 블록의 Hash value + timestamp
  - 각 블록은 앞 블록의 Hash value를 담고 있어 체인처럼 연결 됨
    - Hash function: 데이터를 고정된 길이의 고유한 문자열로 바꿔주는 함수
    - Hash value: 해시 함수 적용한 결과물

# 블록체인

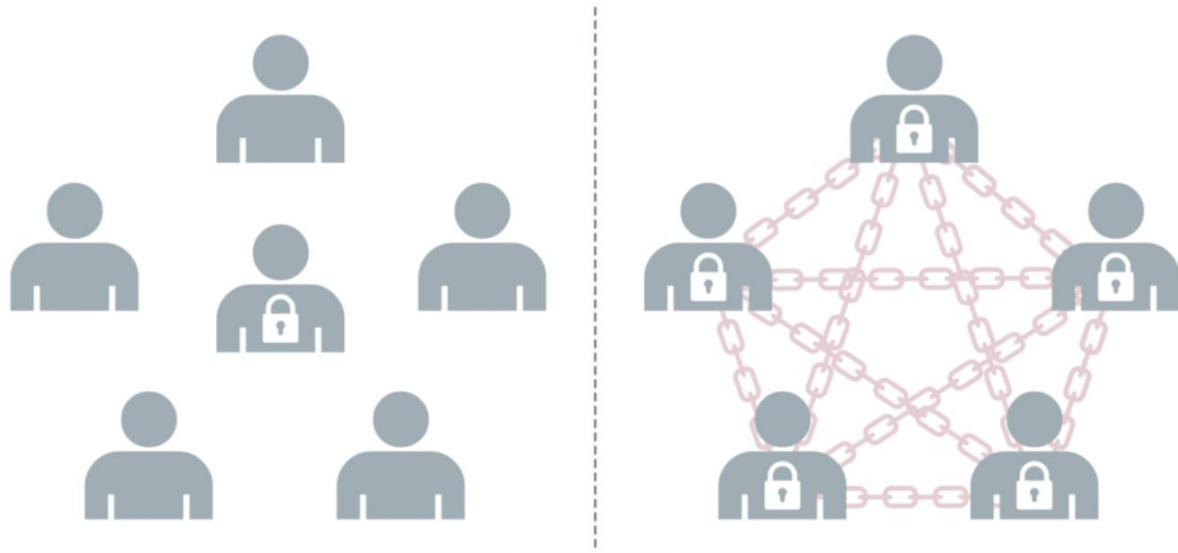
## 블록체인 송금과정



# 블록체인

- 거래가 발생할 때마다 거래와 관련된 데이터를 이전의 장부에 추가로 기록·저장하며, 각 참여자의 장부와 일치하는지 수시로 대조 및 확인
- 위변조?
  - 블록 하나를 수정하면 hash value 바뀜 → 다음 블록과 불일치 → 체인 전체가 무효
  - 블록체인은 다수결의 원칙으로 동작
    - 네트워크 참여자 중 과반수(51% 이상) 갖고 있는 장부를 인정
    - 네트워크에 참여하는 수천~수만 개의 노드(컴퓨터)가 똑같은 장부를 각자 보유
    - 위조 위해서는 네트워크 전체의 51% 이상을 모두 바꿔야함

# 블록체인 특징



〈 현재 〉 중앙집중식		〈 미래 〉 블록체인
거래 시 개인정보 필수	익명성	거래 시 개인정보 불필요
거래 정보 미공개	투명성	허가 대상에 거래 정보 공개
한 곳에 데이터가 집중돼 해킹 위험 존재	보안성	동일 데이터가 분산돼 있어 해킹에 안전, 위·변조불가
데이터 수정, 삭제 가능	가역성	데이터 수정, 삭제 불가능

# 암호화폐

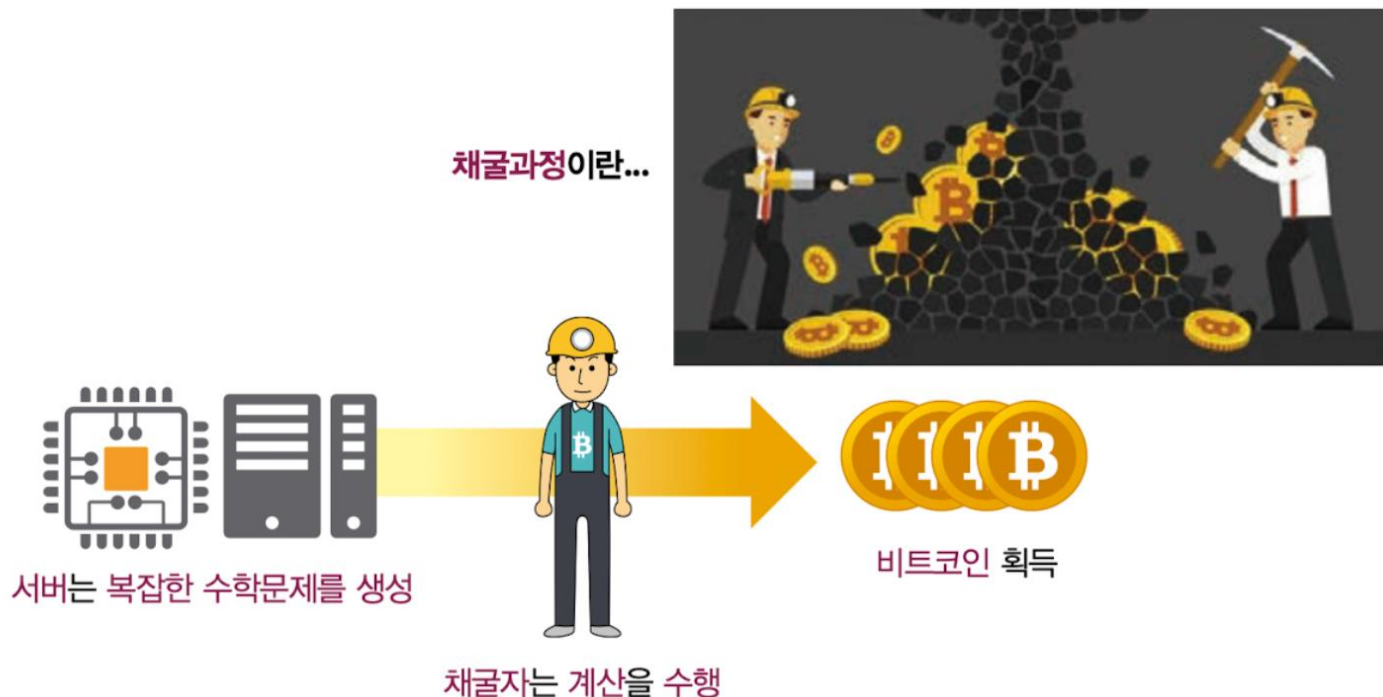
- 암호화폐(Crypto Currency)
  - 암호화(Crypto) + 통화(Currency)
  - 은행 같은 중앙 기관 없이 개인 간 직접 가치를 전송할 수 있음
  - 블록체인 기술을 화폐에 응용하는 경우가 많음
    - 블록체인으로 거래 내역을 투명하게 기록, 위변조 방지
    - 비트코인, 이더리움 등 다양한 종류 존재

# 암호화폐 - 비트코인

- 비트코인(Bitcoin)
  - 최초의 암호화폐
  - Satoshi Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System", 2008
- 아이디어: 은행 없이도 사람들끼리 믿고 거래할 수 있는 시스템의 개발
- 블록체인 기술로 구현한 첫번째 암호화폐
- 무한하지 않음
  - 2100만개의 비트코인이 생성되도록 설계 (반감기 존재)

# 암호화폐 - 채굴

- 채굴(Mining)
  - 블록체인에 거래 내용을 기록하고 기존의 체인에 연결하는 과정



# 암호화폐 - 채굴

- 채굴(Mining)
  - 블록체인에 거래 내용을 기록하고 기존의 체인에 연결하는 과정
  - 블록체인은 중앙 관리자가 없음 → 누가 새 거래를 검증, 장부에 기록할까?
  - 누구나 참여할 수 있지만, 수학적 퍼즐을 풀어야만 블록을 추가할 수 있도록 함
- Proof of Work (PoW): 수학적 퍼즐을 컴퓨터로 푸는 경쟁
  - Puzzle? 블록 데이터 + 임의의 숫자(Nonce)를 합쳐서 SHA-256 해시를 돌렸을 때, 결과값이 특정 조건(앞에 0이 N개 이상)을 만족하는 Nonce를 찾아라
  - 가장 먼저 푸는 컴퓨터가 블록 추가 권한 + 비트코인 보상

# 블록체인 응용들

- 스마트 계약(Smart contract)
  - 계약 조건을 코드로 작성해두면 조건이 충족될 때 자동으로 실행
  - 코드가 계약 → 인간의 개입, 실수, 부정 없음
- 공급망 관리(Supply chain)
  - 제품의 원산지, 유통 이력, 품질 정보 등을 공개 및 추적
- 의료 분야
  - 환자 의료 기록을 블록체인에 저장
  - 의료 데이터 위변조 방지
  - COOV: 블록체인 기반 전자 예방접종 증명 어플리케이션(접종 기록을 블록체인에 저장)

# Summary

---

- IoT
- Cloud Computing
- Blockchain