**Graphene**  is the basic structural element of many other allotropes of carbon, such as [graphite](https://en.wikipedia.org/wiki/Graphite), [diamond](https://en.wikipedia.org/wiki/Diamond), [charcoal](https://en.wikipedia.org/wiki/Charcoal), [carbon nanotubes](https://en.wikipedia.org/wiki/Carbon_nanotube) and [fullerenes](https://en.wikipedia.org/wiki/Fullerene).  It is the strongest material ever tested, efficiently conducts heat and electricity, and is nearly transparent.

**그래핀**은 흑연, 다이아몬드, 숯, 탄소 나노 튜브 그리고 풀러렌과 같은 많은 다른 탄소 합금들의 기본 구조적인 요소입니다. 이것은 테스트 받아 본 것 중 가장 강한 재료이며 열과 전기를 효과적으로 전도하고 거의 투명하다.

An **algorithm** is an unambiguous specification of how to solve a class of problems. Algorithms can perform [calculation](https://en.wikipedia.org/wiki/Calculation), [data processing](https://en.wikipedia.org/wiki/Data_processing) and [automated reasoning](https://en.wikipedia.org/wiki/Automated_reasoning) tasks.

**알고리즘**은 일련의 문제를 해결하는 방법에 대한 명확한 명세이다. 알고리즘은 계산, 데이터 처리 및 자동 추론 작업을 수행할 수 있다.

**Machine learning** is a field of [computer science](https://en.wikipedia.org/wiki/Computer_science) that gives computer systems the ability to "learn" (i.e., progressively improve performance on a specific task) with [data](https://en.wikipedia.org/wiki/Data), without being explicitly programmed.

**기계 학습**은 컴퓨터 시스템에 명시적으로 프로그램 되지 않은 데이터로 "학습" (특정 업무에 대한 수행 능력을 점진적으로 개선시키는 등) 할 수 있는 능력을 주는 컴퓨터 공학 분야입니다.

A **graphics processing unit** (**GPU**) is a specialized [electronic circuit](https://en.wikipedia.org/wiki/Electronic_circuit) designed to rapidly manipulate and alter [memory](https://en.wikipedia.org/wiki/Memory_(computing)) to accelerate the creation of [images](https://en.wikipedia.org/wiki/Image) in a [frame buffer](https://en.wikipedia.org/wiki/Frame_buffer) intended for output to a [display device](https://en.wikipedia.org/wiki/Display_device).

**GPU(그래픽 처리 장치)**는 디스플레이 장치로 출력하기 위한 프레임 버퍼에 있는 이미지의 생성을 가속화하기 위해 메모리를 빠르게 조작하고 변경하도록 설계된 특수 전자 회로입니다.

**Metacognition** is "[cognition](https://en.wikipedia.org/wiki/Cognition) about cognition", "thinking about thinking", "knowing about knowing", becoming "aware of one's awareness" and higher-order thinking skills. The term comes from the root word [*meta*](https://en.wikipedia.org/wiki/Meta), meaning "beyond".

메타인지란"인식에 대한 인식","생각에 대해 생각하는 것"," 아는 것에 대해 아는 것","인식에 대한 인식을 인식하는 것"그리고 더 높은 순서의 사고 기술을 말합니다. 이 용어는 "초과"를 의미하는 메타라는 단어를 뿌리로부터 온다.

A **computing platform** is the environment in which a piece of [software](https://en.wikipedia.org/wiki/Software) is executed. It may be the [hardware](https://en.wikipedia.org/wiki/Computer_hardware) or the [operating system](https://en.wikipedia.org/wiki/Operating_system) (OS), even a [web browser](https://en.wikipedia.org/wiki/Web_browser) or other underlying software, as long as the program code is executed in it.  A computing platform is the stage on which computer programs can run.

**컴퓨팅 플랫폼**은 소프트웨어가 실행되는 환경입니다. 컴퓨터 플랫폼이란 그것 안에서 프로그램 코드가 실행되는 한, 그것은 하드웨어, 운영 체제, 웹 브라우저 또는 다른 기본적인 소프트웨어가 될 수 있다.

**Open-source software** is software whose source code is published and made available to the public, enabling anyone to copy, modify and redistribute the source code without paying royalties or fees. One example of open-source software products is Linux.

**오픈 소스 소프트웨어**는 누구나 로열티나 수수료를 지불하지 않고 소스 코드를 복사하고, 수정하고, 재배포할 수 있도록 하는 소스 코드를 공개하여 대중이 이용할 수 있는 소프트웨어입니다. 오픈 소스 소프트웨어 제품의 한 예는 리눅스이다.

**Artificial intelligence** (**AI**, also **machine intelligence**, **MI**) is [intelligence](https://en.wikipedia.org/wiki/Intelligence) demonstrated by [machines](https://en.wikipedia.org/wiki/Machine), in contrast to the **natural intelligence** (**NI**) displayed by humans and other animals.

**인공 지능(AI)**은 인간과 다른 동물들이 보여 주는 자연 지능(NI)과 달리 기계로 시연되는 지능이다.

**Big data** is [data sets](https://en.wikipedia.org/wiki/Data_set) that are so voluminous and complex that traditional [data-processing](https://en.wikipedia.org/wiki/Data_processing) [application software](https://en.wikipedia.org/wiki/Application_software) are inadequate to deal with them.

**빅 데이터**는 너무 방대하고 복잡하여 기존의 데이터 처리 애플리케이션 소프트웨어로는 처리할 수 없는 데이터 세트입니다.

**Artificial neural networks** (**ANNs**) are computing systems vaguely inspired by the [biological neural networks](https://en.wikipedia.org/wiki/Biological_neural_network) that constitute animal [brains](https://en.wikipedia.org/wiki/Brain). Such systems "learn" (i.e. progressively improve performance on) tasks by considering examples, generally without task-specific programming.

**인공 신경 네트워크**는 동물의 뇌를 구성하는 생물학적 신경 네트워크에 의해 모호하게 영감을 받아 시스템을 계산한다. 이러한 시스템은 일반적으로 작업별 프로그래밍 없이 예제를 고려하여 작업에 대한 "학습"(즉, 성능을 점진적으로 개선) 한다.

The **Internet of things** (**IoT**) is the network of physical devices, vehicles, home appliances and other items [embedded](https://en.wikipedia.org/wiki/Embedded_system) with [electronics](https://en.wikipedia.org/wiki/Electronics), [software](https://en.wikipedia.org/wiki/Software), [sensors](https://en.wikipedia.org/wiki/Sensor), [actuators](https://en.wikipedia.org/wiki/Actuator), and [connectivity](https://en.wikipedia.org/wiki/Internet_access) which enables these objects to connect and exchange [data](https://en.wikipedia.org/wiki/Data).

**사물 인터넷(IoT)**은 물리적 장치, 차량, 가전 제품 및 이러한 개체가 데이터를 연결하고 교환할 수 있도록 해 주는 전자 장치, 소프트웨어, 센서, 액추에이터 및 연결이 내장된 기타 항목의 네트워크입니다.

**Deep learning** (also known as **deep structured learning** or **hierarchical learning**) is part of a broader family of [machine learning](https://en.wikipedia.org/wiki/Machine_learning) methods based on [learning data representations](https://en.wikipedia.org/wiki/Learning_representation), as opposed to task-specific algorithms. Learning can be [supervised](https://en.wikipedia.org/wiki/Supervised_learning), [semi-supervised](https://en.wikipedia.org/wiki/Semi-supervised_learning) or [unsupervised](https://en.wikipedia.org/wiki/Unsupervised_learning).

**딥러닝**( 깊은 체계적 학습 또는 계층적 학습이라고도 함)은 작업별 알고리즘과는 달리 데이터 표현을 학습하는 것에 기초한 더 광범위한 기계 학습 방법 군의 일부이다. 학습은 감독, 반 감독 또는 감독되지 않은 상태로 진행될 수 있습니다.

In [computer engineering](https://en.wikipedia.org/wiki/Computer_engineering), **computer architecture** is a set of rules and methods that describe the functionality, organization, and implementation of [computer](https://en.wikipedia.org/wiki/Computer) systems. Some definitions of architecture define it as describing the capabilities and programming model of a computer but not a particular implementation. In other definitions computer architecture involves [instruction set architecture](https://en.wikipedia.org/wiki/Instruction_set_architecture) design, [microarchitecture](https://en.wikipedia.org/wiki/Microarchitecture) design, [logic design](https://en.wikipedia.org/wiki/Logic_design), and [implementation](https://en.wikipedia.org/wiki/Implementation).

컴퓨터 공학에서, **컴퓨터 구조**는 컴퓨터 시스템의 기능, 조직 그리고 실행을 기술하는 일련의 규칙과 방법들이다. 아키텍처의 일부 정의는 그것을 특정한 구현이 아닌 컴퓨터의 능력과 프로그래밍 모델을 설명하는 것으로 정의한다. 다른 정의에서는 컴퓨터 아키텍처는 명령 집합 아키텍처 설계, 마이크로 아키텍처 설계, 논리 설계 및 구현을 포함한다.

**Information technology** (**IT**) is the application of [computers](https://en.wikipedia.org/wiki/Computer) to store, retrieve, transmit and manipulate [data](https://en.wikipedia.org/wiki/Data_(computing)), or [information](https://en.wikipedia.org/wiki/Information), often in the context of a business or other enterprise.

**IT(정보 기술)**는 데이터 또는 정보를 저장, 검색, 전송 및 조작하기 위해 컴퓨터를 사용하는 애플리케이션으로, 기업 또는 기타 기업 환경에서 흔히 그러하다.

**AlphaGo** is a [computer program](https://en.wikipedia.org/wiki/Computer_program) that plays the [board game](https://en.wikipedia.org/wiki/Board_game) [Go](https://en.wikipedia.org/wiki/Go_(game)). It was developed by [Alphabet Inc.](https://en.wikipedia.org/wiki/Alphabet_Inc.)'s [Google DeepMind](https://en.wikipedia.org/wiki/Google_DeepMind) in London.

**알파고**는 바둑을 실행하는 컴퓨터 프로그램이다. 이는 알파벳주식회사의 구글 딥마인드에서 개발되었다.

A **massive open online course** (**MOOC**) is an online course aimed at unlimited participation and open access via the web.

대규모 개방형 온라인 강좌(MOOC)는 웹을 통한 무제한적인 참여와 개방형 접속을 목표로 하는 온라인 강좌이다.

The [collective noun](https://en.wikipedia.org/wiki/Collective_noun) **application software** refers to all applications collectively. This contrasts with [system software](https://en.wikipedia.org/wiki/System_software), which is mainly involved with running the computer.

집합 명사 적용 소프트웨어는 집합적으로 모든 응용 프로그램을 말한다. 이는 주로 컴퓨터 운영에 관련된 시스템 소프트웨어와 대조된다.

**Computer hardware** are the physical parts or components of a computer, such as the [monitor](https://en.wikipedia.org/wiki/Computer_monitor" \o "Computer monitor),keyboard, computer datastorage, graphic card, sound card and motherboard. By contrast, software is instructions that can be stored and ran by hardware. Hardware is directed by the software to execute any command or instruction. A combination of hardware and software forms a usable computing system.

컴퓨터 하드웨어는 모니터, 키보드, 컴퓨터 데이터 스토리지, 그래픽 카드, 사운드 카드 및 마더 보드와 같은 컴퓨터의 물리적 부품 또는 구성 요소입니다. 대조적으로, 소프트웨어는 하드웨어에 의해서 저장되고 실행될 수 있는 지침서이다. 하드웨어는 모든 명령이나 명령을 실행하도록 소프트웨어에 의해 지시된다. 하드웨어와 소프트웨어의 조합이 사용 가능한 컴퓨팅 시스템을 구성합니다.

A **central processing unit** (**CPU**) is the [electronic circuitry](https://en.wikipedia.org/wiki/Electronic_circuit) within a [computer](https://en.wikipedia.org/wiki/Computer) that carries out the [instructions](https://en.wikipedia.org/wiki/Instruction_(computing)) of a [computer program](https://en.wikipedia.org/wiki/Computer_program) by performing the basic [arithmetic](https://en.wikipedia.org/wiki/Arithmetic), logical, control and [input/output](https://en.wikipedia.org/wiki/Input/output) (I/O) operations specified by the instructions. Traditionally, the term "CPU" refers to a **processor**, more specifically to its processing unit and [control unit](https://en.wikipedia.org/wiki/Control_unit) (CU), distinguishing these core elements of a computer from external components such as [main memory](https://en.wikipedia.org/wiki/Main_memory) and [I/O](https://en.wikipedia.org/wiki/I/O) circuitry.

CPU(CentralProcessingUnit)는 지시에 의해 지정된 기본 산술, 논리, 컨트롤 및 입출력(I/O)작업을 수행하여 컴퓨터 프로그램의 지시를 수행하는 컴퓨터 내부의 전자 회로입니다. 전통적으로 "CPU"라는 용어는 프로세서, 특히 처리 장치와 제어 장치(CU)에 표시되어, 이러한 컴퓨터의 핵심 요소들을 주 기억 장치와 I/O회로와 같은 외부 구성 요소들과 구별한다.

An **application program** (**app** or **application** for short) is a [computer program](https://en.wikipedia.org/wiki/Computer_program) designed to perform a group of coordinated functions, tasks, or activities for the benefit of the user. Examples of an application include a [word processor](https://en.wikipedia.org/wiki/Word_processor), a [spreadsheet](https://en.wikipedia.org/wiki/Spreadsheet), an [accounting application](https://en.wikipedia.org/wiki/Accounting_software), a [web browser](https://en.wikipedia.org/wiki/Web_browser), a media player, an aeronautical [flight simulator](https://en.wikipedia.org/wiki/Flight_simulator), a [console game](https://en.wikipedia.org/wiki/Console_game) or a [photo editor](https://en.wikipedia.org/wiki/Raster_graphics_editor).

애플리케이션 프로그램(애플리케이션 또는 줄여서 애플리케이션)은 사용자의 이익을 위해 조정된 기능, 작업 또는 활동 그룹을 수행하도록 설계된 컴퓨터 프로그램이다. 응용 프로그램의 예로는 워드 프로세서, 스프레드 시트, 회계 응용 프로그램, 웹 브라우저, 미디어 플레이어, 항공기 비행 시뮬레이터, 콘솔 게임 또는 사진 편집기가 있다.

**Watson** is a [question answering](https://en.wikipedia.org/wiki/Question_answering) computer system capable of answering questions posed in [natural language](https://en.wikipedia.org/wiki/Natural_language), developed in [IBM](https://en.wikipedia.org/wiki/IBM)'s DeepQA project by a research team led by [principal investigator](https://en.wikipedia.org/wiki/Principal_investigator) [David Ferrucci](https://en.wikipedia.org/wiki/David_Ferrucci). Watson was named after IBM's first CEO, industrialist [Thomas J. Watson](https://en.wikipedia.org/wiki/Thomas_J._Watson).

왓슨은, 자연 언어로 제기되는 질문에 답할 수 있는 컴퓨터 시스템이며, 이 질문은 수석 연구원 데이비드 페루치가 이끄는 IBM개발 팀의 연구에서 개발된 것이다. 왓슨은 IBM의 첫번째 최고 경영자인 기업가 토마스 J. 왓슨의 이름을 따서 지어졌다.

**Random-access memory** (**RAM**) is a form of computer data storage that stores data and machine code currently being used. A random-access memory device allows data items to be read or written in almost the same amount of time irrespective of the physical location of data inside the memory. In contrast, with other direct-access data storage media such as hard disks, CD-RWs, DVD-RWs and the older magnetic tapes and drum memory, the time required to read and write data items varies significantly depending on their physical locations on the recording medium, due to mechanical limitations such as media rotation speeds and arm movement.

RAM(Random-AccessMemory)은 현재 사용 중인 데이터와 시스템 코드를 저장하는 일종의 컴퓨터 데이터 스토리지입니다. 랜덤 액세스 메모리 장치는 메모리 내의 데이터의 물리적 위치에 관계 없이 데이터 항목을 거의 동일한 시간에 읽거나 쓸 수 있도록 한다. 반면, 하드 디스크, CD-RW, DVD-RW및 이전의 자기 테이프와 드럼 메모리와 같은 다른 직접 액세스 데이터 저장 매체는 물리적 데이터 항목을 읽고 쓰는 데 필요한 시간이 크게 다릅니다.

**Computer software**, or simply **software**, is a part of a [computer system](https://en.wikipedia.org/wiki/Computer_system) that consists of [data](https://en.wikipedia.org/wiki/Data_(computing)) or computer instructions, in contrast to the [physical hardware](https://en.wikipedia.org/wiki/Computer_hardware) from which the system is built. In [computer science](https://en.wikipedia.org/wiki/Computer_science) and [software engineering](https://en.wikipedia.org/wiki/Software_engineering), computer software is all [information](https://en.wikipedia.org/wiki/Information) processed by [computer systems](https://en.wikipedia.org/wiki/Computer_system), [programs](https://en.wikipedia.org/wiki/Computer_program) and data. Computer software includes [computer programs](https://en.wikipedia.org/wiki/Computer_program), [libraries](https://en.wikipedia.org/wiki/Library_(computing)) and related non-executable [data](https://en.wikipedia.org/wiki/Data_(computing)), such as [online documentation](https://en.wikipedia.org/wiki/Software_documentation) or [digital media](https://en.wikipedia.org/wiki/Digital_media). Computer hardware and software require each other and neither can be realistically used on its own.

컴퓨터 소프트웨어 또는 단순한 소프트웨어는 시스템이 지어지는 물리적 하드웨어와는 대조적으로 데이터나 컴퓨터 명령으로 이루어진 컴퓨터 시스템의 일부이다. 컴퓨터 공학과 소프트웨어 공학에서, 컴퓨터 소프트웨어는 컴퓨터 시스템과 프로그램 그리고 자료에 의해 처리되는 모든 정보들이다. 컴퓨터 소프트웨어에는 컴퓨터 프로그램, 도서관 및 온라인 설명서나 디지털 미디어와 같은 관련된 실행 불가능한 데이터가 포함되어 있습니다. 컴퓨터 하드웨어와 소프트웨어는 서로를 필요로 하고 둘 다 현실적으로 사용할 수 없다.