데이터(Data)

* 측정이나 수집을 통해 단순하게 모아놓은 원시 자료

정보(Information)

* 수집한 데이터를 목적이나 의도에 따라 처리하고 분석하여 데이터에 의미를 부여한 것

데이터베이스

* 특정 조직 내에서 다수의 사용자가 사용할 수 있도록 통합하여 저장한 운영 데이터의 집합

관계형 데이터베이스(Relational Database)

* 데이터를 테이블의 형태로 저장한다.
* 키값을 이용해서 테이블 간의 종속성을 관계로 표현한다.

테이블

* 데이터베이스에서 행(가로줄, row, record)과 열(세로줄, column, field)로 짜여진 표에 기록된 데이터의 집합
  + 필드 : 테이블에서 열에 해당하는 가장 작은 단위의 데이터
  + 레코드 : 테이블에서 행에 해당하는 것으로 논리적으로 연관된 필드의 집합
  + 도메인 : 필드의 데이터타입에 맞는 필드가 가질 수 있는 값들의 집합
  + 키 : 데이터베이스에 저장되어 있는 레코드를 구별할 수 있게 해주는 식별자
  + 기본키 : 테이블에 있는 모든 레코드를 식별할 수 있게 해주는 필드 중 대표로 선택한 것

테이블의 구성요건

* 테이블의 레코드는 유일해야 한다.
* 테이블의 레코드들 간에는 순서가 없다.
* 테이블의 필드간에도 순서가 없다.

데이터베이스 언어(쿼리 랭귀지)

* 데이터베이스의 구조 정의, 데이터 조작, 데이터 보호, 데이터 복구를 위해 사용되는 언어
* 데이터 조작어(DML : Data Manipulation Language) : DB 조회 검색 변형
* 데이터 정의어(DDL : Data Definition Language) : 테이블 생성 변형 삭제
* 데이터 제어어(DCL : Data Control Language) : 관리자용(권한 주고 회수)

데이터베이스의 성질(ACID)

* Atomicity(원자성) : 트랜잭션과 관련된 작업들이 부분적으로 실행되다가 중단되지 않아야 한다는 의미
* Consistency(일관성) : 트랜잭션 이후에 데이터베이스의 상태가 일관되어야 한다는 의미(DB의 제약이나 규칙에 맞아야 한다.)
* Isolation(격리성) : 모든 트랜잭션은 다른 트랜잭션으로부터 독립적이어야 한다는 의미
* Durability(지속성) : 성공적으로 트랜잭션이 수행되면 영원히 반영되어야 한다는 의미

DBMS(Database Management System)

* 데이터베이스를 생성하여 안정적이고 효율적으로 운영하는데 필요한 기능을 제공하는 소프트웨어
* 종류
  + 관계형 데이터베이스

파일

* 컴퓨터에서 사용되는 데이터를 모아놓은 것

파일의 저장위치

* 논리적 저장위치 : 디렉토리 또는 폴더
* 물리적 저장위치 : 디스크(HDD,SSD,USB 등등)

파일을 구분하기 위해 사용하는 것

* 파일 이름

파일의 종류와 그 역할을 표시하기 위해 사용하는 것

* 파일 확장자
* 파일이름 뒤에 (.)을 붙이고 작성한다.

텍스트 파일

* 사람이 확인할 수 있는 문자열을 저장하는 파일 유형(일반적인 문서 파일은 변환해서 보관)
* 확장자 : txt
* 모든 프로그래밍 코드는 텍스트 파일

개행문자

* \n : Line Feed. 커서를 아래줄로 옮긴다.
* \r : Carriage Return. 커서를 중의 제일 앞으로 옮긴다.

CSV(Comma Separated Values)

* 필드를 콤마와 같은 구분문자를 사용하여 구분한다.
* 한줄을 하나의 레코드로 인식

문자 인코딩

* 사용자가 입력한 문자나 기호들을 컴퓨터가 이용할 수 있는 신호로 변경하는 것

ASCII(American Standard Code for Information Interchange)

* 미국표준협회(ASA)에서 개발한 영문 알파벳을 사용하는 가장 기본적인 문자 인코딩 방식.
* 문자를 표현하기 위해 7bit(2의7승)를 사용한다.

Unicode

* 전 세계의 모든 문자를 컴퓨터에서 일관되게 표현하고 다룰 수 있도록 설계된 산업 표준
* UTF-8
  + Unicode 인코딩 방식 중 하나
  + 글자 하나를 표현하기 위해 1~4Byte의 가변 길이를 사용한다.
  + ASCII 코드를 포함하고 있어 하위 호환성이 보장된다.
  + 인코딩을 하는데 비트 연산만 사용되어 속도가 빠르다.

JSON(JavaScript Object Notation)

* 자바스크립트에서 파생된 포맷으로 구조화된 데이터를 표현하기 위한 문자기반의 경략 데이터 포맷

Markup Language(마크업 언어)

* 문서나 데이터의 구조를 명기하는 언어

XML(eXtenable Markup Language)

* 인터넷으로 연결된 서로 다른 종류의 시스템끼리 데이터를 쉽게 주고 받기 위해 만들어진 언어

C# 파일 읽고 쓰기

* using System.IO : 파일을 읽거나 쓰기 위해 필요한 네임스페이스
* File : 단일 파일에 대한 만들기, 복사 , 삭제, 이동 및 열기를 위한 정적 메서드를 제공하고 FileStream 개체 만들기를 지원하는 클래스
* string read = File.ReadAllText(@”C#/naaa/text.txt”);
* StreamReader : 특정 인코딩의 바이스 스트림에서 문자를 읽는데 사용하는 클래스
* StreamWriter : 특정 인코딩의 바이트 스트림에서 문자를 쓰는데 사용하는 클래스

파싱(Parsing)

* 일련의 문자열에서 특정 패턴이나 순서를 파악해 의미이ㅆ는 토큰(token)을 추출한 후 토큰에서 데이터테이블의휵성