* 파일(file)
* 레코드가 순차적으로 저장, 각각의 레코드는 동일한 형태의 필드가 존재한다.
* 데이터 파일의 구조

하드디스크(연속된 공간)에 데이터가 어떻게 저장이 될까?

연속된 공간의 한계를 타파하기 위해서 고안된 자료구조 => 링크드 리스트(Linked list)

논점

하드디스크에 데이터 파일을 저장할 때 연속된 공간에 배열과 같이 저장이 가능할 것인가?

* 파일의 크기는 변하는가 변하지 않는가? 파일의 크기는 변한다!
* 그러면 과연 크기가 변한다면 연속된 공간에 할당이 어려울 것인가?
* 파일의 크기가 변화하지 않는다면 연속된 공간에 할당이 가능할 것인가?

파일의 위치를 이동하지 않는 한 불가능하다.

메모리의 파편화 (메모리를 넣다 뺏다 하다보니 메모리가 구멍이 송송 뚫리듯이 되는 현상)

메모리의 파편화를 정리해주는 역할 => Garbage Collector (ex. 하드디스크 조각모음)

실제로 하드디스크에는 나누어진 조각조각들을 저장하고 있다.

궁금증? (해결방안이 필요하다면 문제점, 궁금증 부터 알아야 한다.)

1. 각각의 조각 파일데이터들을 어떻게 관리할 것인가?

운영체제에서 관리한다. (Linux, Unix)

파일 관리 방법: 연결리스트, 디렉토리(숫자를 달아 연속된 순서대로 만든다.), 꼬리표를 달아 전체 탐색 등

1. 과연 파일을 조각내는 기준은?

레코드 크기별로 쪼갠다 => 레코드가 너무 커지면 문제, 레코드의 길이가 서로 다르다면?

레코드의 크기는 적당하고 동일하자고 가정, 레코드는 연속된 공간에 저장이 가능하다고 가정

1.

->메인메모리는 랜덤 엑세스가 가능하다. 어느 위치를 엑세스 하던 속도 차이가 없다.

->하드디스크는 레코드와 작동원리가 같다. 위치에 따라 엑세스 하는 속도가 다르다.

=> 그래서 하드디스크에는 어느 위치에 저장해야 한다는 것이 중요하다.

2.

->메인메모리는 비트, 바이트 단위로 정확하게 엑세스한다.

->하드디스크는 (규격화된 단위로 엑세스한다). 단위는 OS, HW제조사마다 다르다.

디스크 블록(한번에 엑세스해서 읽어올 수 있는 크기)(크기는 OS, HW제조사마다 다르다.)

디스크 블록은 일정한 크기이다.

디스크 블록 단위로 쪼개서(블록안에 레코드를 꽉꽉 채워넣는다) 그것을 연결리스트로 관리한다.

순서 파일(특정 필드 값의 크기 순으로 레코드를 저장한 파일),

비 순서 파일(특정 필드 값의 크기 순서 상관없이 레코드를 저장한 파일)

.