13장: 질의 최적화

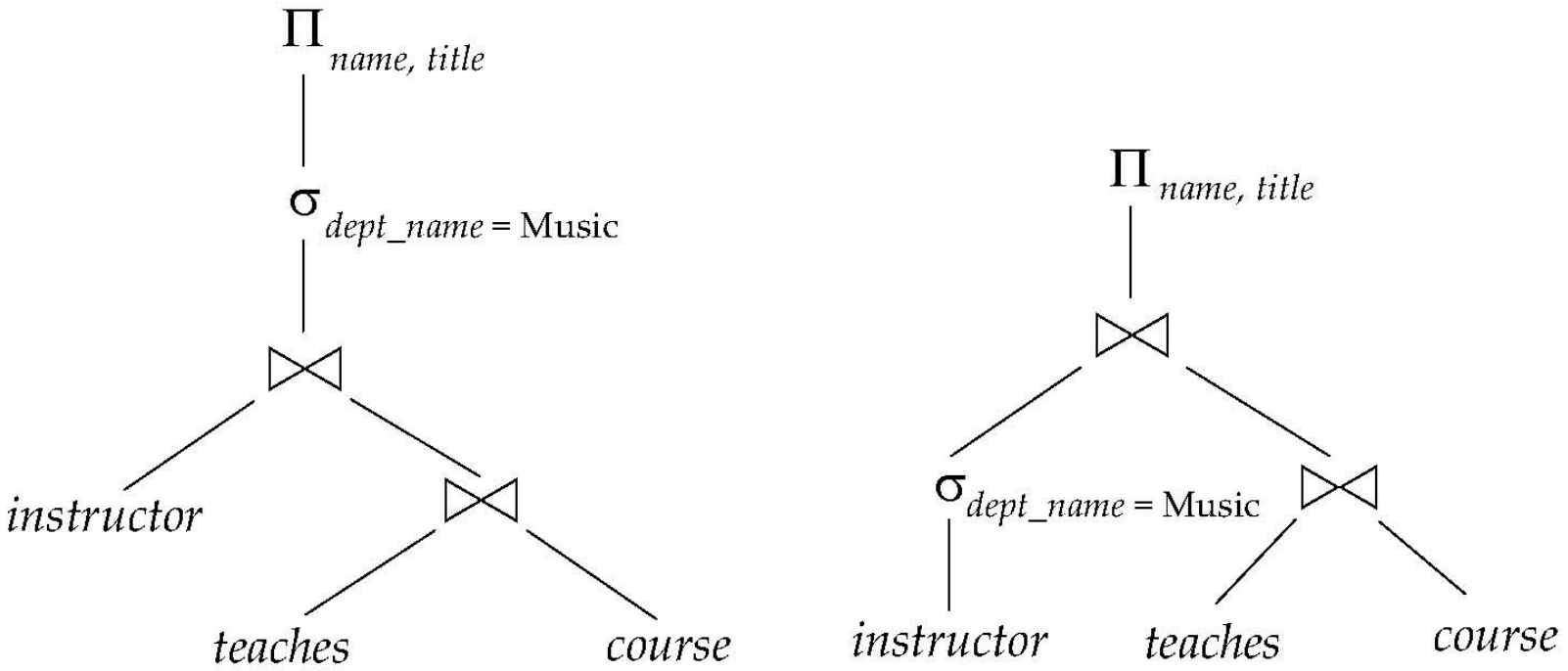
Dept. of Computer Engineering
SeoKyeong University

13.1 질의 최적화 개요

- 질의최적화(Query Optimization)
  - 주어진 질의를 처리할 수 있는 많은 질의수행계획 가운데 가장 효율적인 것을 선택하는 과정
- 기본 방법
  - 관계대수 단계에서 주어진 식과 동등하면서 보다 효율적으로 수행할 수 있는 식을 찾음
  - 어떤 연산을 수행하기 위한 알고리즘 선택이나 인덱스 사용 등과 같은 질의처리의 세부적인 방법을 선택
- 효율적, 비효율적 방법의 비용차이는 매우 큼 → 최적화에 따른 비용을 감수할 만한 가치가 있음

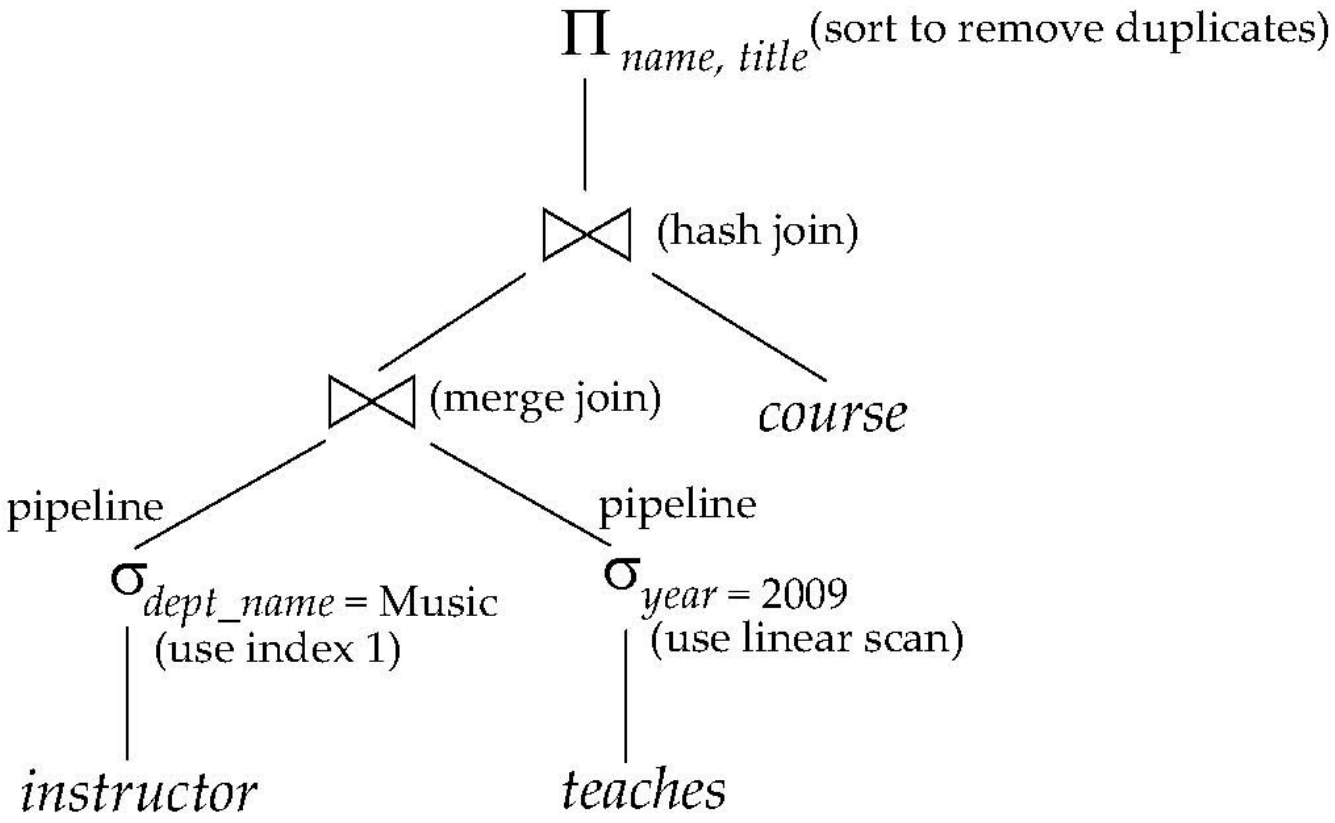
질의 최적화 개요

- "음악학과에 소속된 모든 교수들의 이름과 이들이 가르치는 교과목명을 추출"
  - 음악학과에 소속된 교수들로 먼저 한정한 후 진행하면 중간결과의 크기를 많이 줄일 수 있음



질의 최적화 개요

- 수행계획: 각 연산을 위하여 어떤 알고리즘이 사용되어야 하는지, 연산의 수행이 어떤 식으로 진행되어야 하는지를 정확하게 정의하는 것



질의 최적화 개요

- 주어진 관계대수식을 가장 적은 비용으로(또는 가장 적은 비용보다 크게 많지 않은 비용으로) 주어진 식과 동일한 결과를 출력하는 수행계획을 찾아내는 것 → 질의최적기의 임무
- 가장 적은 비용의 질의수행계획을 찾기 위해서는
  - 주어진 식에 대해 같은 결과를 출력하는 다른 질의수행계획을 생성
  - 이후, 가장 적은 비용으로 수행하는 계획을 선택
  - 3단계
    1) 주어진 식과 논리적으로 동일한 식을 도출 ← 동등 규칙 적용
    2) 각 식에 여러 가지 방법의 주석을 첨부한 서로 다른 질의수행계획을 생성
    3) 각 수행계획의 비용을 추정 → 가장 적은 것을 선택
- 비용은 추정된 것임
  - 선택된 계획이 반드시 최소비용이어야 하는 것은 아님
  - 하지만, 추정이 잘 되었다면 선택된 계획의 비용 절감에 큰 역할을 할 것임

13.2 관계형 식의 변환

- 두 개의 관계대수식은 동일한 튜플 집합을 생성하는 경우 이들은 동등하다(Equivalent)라고 함
  - 이때 생성된 튜플의 순서는 상관없음
- 동등규칙(Equivalence Rule)
  - 두 가지 형태의 식이 동등할 수 있는 조건
  - 서로 치환되어도 동일한 결과를 생성함
  - 질의최적기는 동등규칙을 이용하여 하나의 식을 논리적으로 동등한 다른 식으로 변환

13.2.1 동등규칙

1. 논리곱이 포함된 선택연산은 각각의 선택연산으로 분해할 수 있음

$$\sigma_{\theta_1 \wedge \theta_2}(E) = \sigma_{\theta_1}(\sigma_{\theta_2}(E))$$

2. 선택연산은 교환법칙이 성립함

$$\sigma_{\theta_1}(\sigma_{\theta_2}(E)) = \sigma_{\theta_2}(\sigma_{\theta_1}(E))$$

3. 일련의 추출연산 중 마지막 연산만 필요하고 나머지 연산들은 생략 가능함

$$\Pi_{L_1}(\Pi_{L_2}(\ldots(\Pi_{Ln}(E))\ldots)) = \Pi_{L_1}(E)$$

4. 선택연산은 Cartesian products와 Theta-join으로 변환됨
   a. $\sigma_{\theta}(E_1 \times E_2) = E_1 \bowtie_{\theta} E_2$
   b. $\sigma_{\theta 1}(E_1 \bowtie_{\theta 2} E_2) = E_1 \bowtie_{\theta 1 \wedge \theta 2} E_2$

동등규칙

5. Theta-join 은 교환법칙이 성립함

$$E_1 \bowtie_\theta E_2 = E_2 \bowtie_\theta E_1$$

6. (a) 자연조인은 결합법칙이 성립함

$$(E_1 \bowtie E_2) \bowtie E_3 = E_1 \bowtie (E_2 \bowtie E_3)$$

(b) Theta-joins 은 다음과 같은 방법으로 결합법칙이 성립함:

$$(E_1 \bowtie_{\theta 1} E_2) \bowtie_{\theta 2 \wedge \theta 3} E_3 = E_1 \bowtie_{\theta 1 \wedge \theta 3} (E_2 \bowtie_{\theta 2} E_3)$$

단, θ_2 는 E_2 과 E_3의 속성에만 연관되어야 함

동등교칙

7. 선택연산은 아래 조건하에서는 Theta-join 연산에 배분될 수 있음

(a) θ_0 에 등장하는 모든 속성은 E_1 에 속하여야 함

$$\sigma_{\theta 0}(E_1 \bowtie_{\theta} E_2) = (\sigma_{\theta 0}(E_1)) \bowtie_{\theta} E_2$$

(b) θ_1은 E_1 의 속성에만 관련되고 θ_2는 E_2 의 속성에만 관련되는 경우

$$\sigma_{\theta 1 \wedge \theta 2}(E_1 \bowtie_{\theta} E_2) = (\sigma_{\theta 1}(E_1)) \bowtie_{\theta} (\sigma_{\theta 2}(E_2))$$

동등규칙

8. 추출연산은 다음과 같은 경우 Theta-join 연산에 배분될 수 있음

(a) θ 는 $L_1 \cup L_2$에 속한 속성들에만 관련되어 있는 경우

$$\Pi_{L_1 \cup L_2}(E_1 \bowtie_\theta E_2) = (\Pi_{L_1}(E_1)) \bowtie_\theta (\Pi_{L_2}(E_2))$$

(b) Theta-join $E_1 \bowtie_\theta E_2$ 에 대하여

- L_1 and L_2 는 각각 E_1 and E_2의 속성이고
- L_3는 E_1의 속성이고 $L_1 \cup L_2$에는 포함되지 않으며
- L_4는 E_2의 속성이고 $L_1 \cup L_2$에는 포함되지 않는 경우

$$\Pi_{L_1 \cup L_2}(E_1 \bowtie_\theta E_2) = \Pi_{L_1 \cup L_2}((\Pi_{L_1 \cup L_3}(E_1)) \bowtie_\theta (\Pi_{L_2 \cup L_4}(E_2)))$$

동등규칙

9. 합집합과 교집합 연산은 교환법칙이 성립

$$E_1 \cup E_2 = E_2 \cup E_1$$
$$E_1 \cap E_2 = E_2 \cap E_1$$

- 하지만, 차집합 연산은 교환법칙이 성립하지 않음

10. 합집합과 교집합은 결합법칙이 성립

$$(E_1 \cup E_2) \cup E_3 = E_1 \cup (E_2 \cup E_3)$$
$$(E_1 \cap E_2) \cap E_3 = E_1 \cap (E_2 \cap E_3)$$

11. 선택연산은 $\cup$, $\cap$, – 연산에 분배될 수 있음

$\sigma_\theta (E_1 - E_2) = \sigma_\theta (E_1) - \sigma_\theta(E_2)$ ← $\cup$, $\cap$에도 성립

$\sigma_\theta (E_1 - E_2) = \sigma_\theta(E_1) - E_2$ ← $\cap$에는 성립하지만 $\cup$에는 성립하지 않음

12. 추출연산은 합집합 연산에 분배될 수 있음

$$\Pi_L(E_1 \cup E_2) = (\Pi_L(E_1)) \cup (\Pi_L(E_2))$$

13.2.2 변환 예: 선택연산 조기 수행

- 질의: 음악학과에 소속된 교수들의 성명과 그들이 강의한 교과목명을 추출
  - $\Pi_{name,\ title}(\sigma_{dept\_name=\text{“Music”}}(instructor \bowtie (teaches \bowtie \Pi_{course\_id,\ title}(course))))$
- 동등규칙 7a를 적용하여 변환
  - $\Pi_{name,\ title}((\sigma_{dept\_name=\text{“Music”}}(instructor)) \bowtie (teaches \bowtie \Pi_{course\_id,\ title}(course)))$
  - 선택연산을 가능한 한 조기에 수행함으로써 조인되는 릴레이션의 크기를 줄일 수 있음

변환 예: 선택연산 조기 수행

- 질의: 2009학년도 개설강좌에 참여한 음악학과 교수들의 성명과 해당 강좌의 교과목명을 함께 추출
  - $\Pi_{name,\ title}(\sigma_{dept\_name=\text{“Music”} \wedge year = 2009}$
    $(instructor \bowtie (teaches \bowtie \Pi_{course\_id,\ title}\ (course))))$
- 동등규칙 6a를 적용하여 변환하면
  - $\Pi_{name,\ title}(\sigma_{dept\_name=\text{“Music”} \wedge year = 2009}$
    $((instructor \bowtie teaches) \bowtie \Pi_{course\_id,\ title}\ (course)))$
  - 이때 선택연산을 조속히 수행하는 것이 유리하다는 원칙을 적용하면

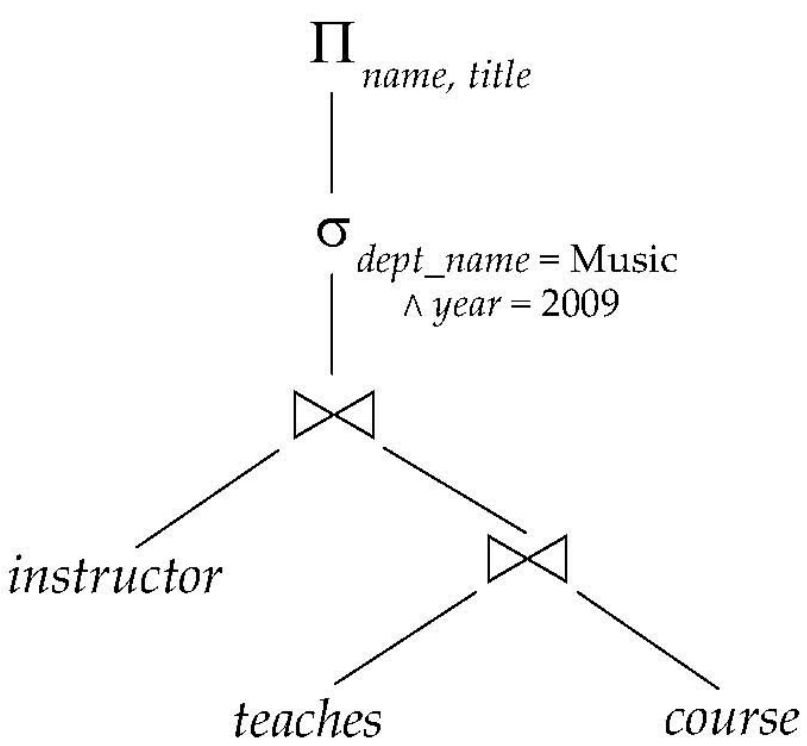
$\sigma_{dept_name=\text{“Music”} \wedge year = 2009}\ ((instructor \bowtie teaches)$를

$\sigma_{dept\_name = \text{“Music”}}\ (instructor) \bowtie \sigma_{\ year = 2009}\ (teaches)$ 로 변환 가능

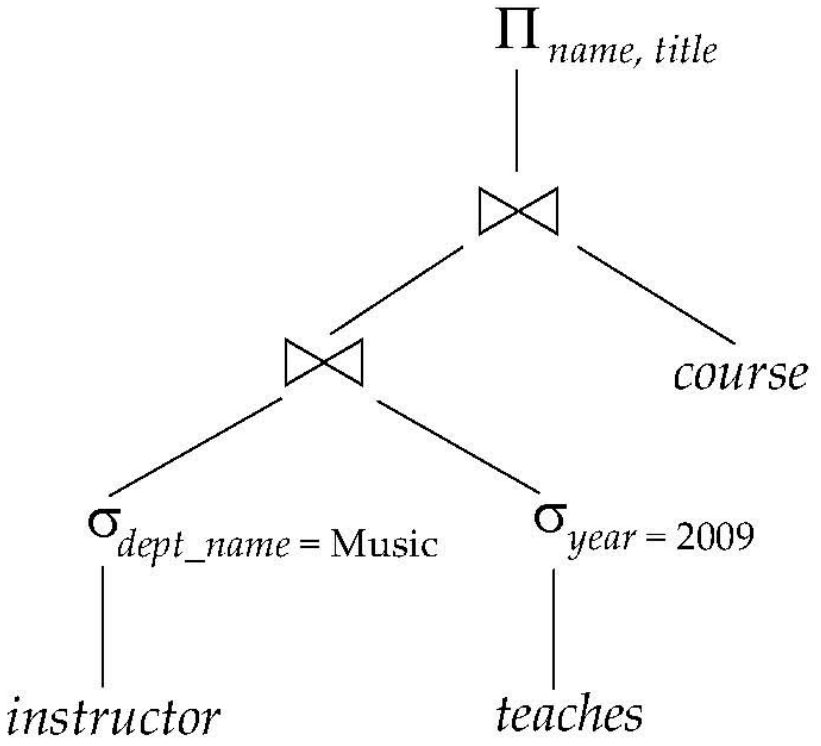
  - 최종적으로 아래와 같이 변환

$\Pi_{name,\ title}((\sigma_{dept\_name = \text{“Music”}}\ (instructor) \bowtie \sigma_{\ year = 2009}\ (teaches))$
$\bowtie \quad \Pi_{course\_id,\ title}\ (course))$

변환 예: 선택연산 조기 수행



(a) Initial expression tree



(b) Tree after multiple transformations

변환 예: 추출연산 조기 수행

- Consider: $\Pi_{name,\ title}(\sigma_{dept\_name=\text{“Music”}}(instructor) \bowtie teaches) \bowtie \Pi_{course\_id,\ title}(course))))$
- 아래 부분식에 대하여

  $(\sigma_{dept\_name = \text{“Music”}}(instructor \bowtie teaches)$
  - 상기 부분식을 수행할 경우 아래 스키마가 중간결과로 생성됨
    (*ID, name, dept_name, salary, course_id, sec_id, semester, year)*
  - 이들 중 실제로 필요한 속성은 결과에 나타나야 하는 속성과 중간 과정에 필요한 속성에 국한됨
- 동등규칙 8a와 8b를 적용하여 중간결과로부터 불필요한 속성을 조기 제거

  $\Pi_{name,\ title}(\Pi_{name,\ course\_id}(\sigma_{dept\_name=\text{“Music”}}(instructor) \bowtie teaches)) \bowtie \Pi_{course\_id,\ title}(course))))$
  - 추출을 조기에 수행함에 따라 조인하여야 하는 릴레이션의 크기를 줄임

13.2.3 조인 순서

- 임시 중간결과의 크기를 줄이기 위해서는 조인연산의 순서가 매우 중요
- 대부분의 질의최적기들은 조인순서에 많은 주의를 기울임
- 조인연산은 결합법칙이 적용됨
  - For all relations r_1, r_2, and r_3,

$$(r_1 \bowtie r_2) \bowtie r_3 = r_1 \bowtie (r_2 \bowtie r_3)$$

  - 만약 $r_2 \bowtie r_3$ 은 매우 크고 and $r_1 \bowtie r_2$ 은 작은 경우 , $(r_1 \bowtie r_2) \bowtie r_3$ 를 선택함

조인 순서

- 아래 관계대수 연산식에 대하여

$$\Pi_{name,\ title}(\sigma_{dept\_name=\text{“Music”}}(instructor) \bowtie (teaches \bowtie \Pi_{course\_id,\ title}(course)))$$

- $teaches \bowtie \Pi_{course\_id,\ title}(course)$ 를 먼저 수행하고 그 결과를 $\sigma_{dept\_name=\text{“Music”}}(instructor)$ 과 조인할 경우 전체 강좌와 전체 교과목을 조인하는 것이므로 중간결과 크기가 매우 큼

- 음악학과에 소속된 교수들에 국한하여 강좌와 조인을 먼저 하면 중간결과 크기를 크게 줄일 수 있음

$$\sigma_{dept\_name=\text{“Music”}}(instructor) \bowtie teaches$$

- 최종 결과

$$\Pi_{name,\ title}((\sigma_{dept\_name=\text{“Music”}}(instructor) \bowtie teaches) \bowtie \Pi_{course\_id,\ title}(course))$$

13.3 결과의 통계정보 추정

- 연산의 비용은 해당 연산에 대한 입력의 크기와 기타 통계정보에 의존적
- 13.3.1 카탈로그 정보
  - 시스템 카탈로그: 데이터베이스 릴레이션의 여러 통계적 정보들을 관리하고 이음
  - 릴레이션에 대한 통계정보
    - n_r: 릴레이션 r의 튜플 수
    - b_r: 릴레이션 r의 튜플들이 차지하는 블록 수
    - l_r: 릴레이션 r의 튜플 크기(바이트)
    - f_r: 릴레이션 r의 블로킹 요인. 즉, 한 블록에 들어가는 릴레이션 r의 튜플 수
    - $V(A, r)$: 속성 A에 대하여 릴레이션 r에서 나타나는 서로 다른 값의 개수
      - A가 속성들의 집합이 될 수도 있음
  - B^+ 트리 인덱스
    - 트리 높이, 인덱스 내의 단말 노드 수 등과 같은 정보도 카탈로그에서 유지

카탈로그 정보

- 카탈로그에 정확한 통계정보를 유지하기 위해서는 릴레이션이 변경될 때마다 통계정보도 갱신되어야 함 → 많은 부담 발생
  - 대부분의 시스템은 부하가 적은 동안에 통계정보를 갱신
    - 질의처리 단계를 결정하기 위해 사용되는 통계정보는 정확한 정보는 아님
    - 하지만, 릴레이션에 대한 수정이 많지 않은 경우 서로 다른 수행계획들의 상대적인 비용 계산에는 충분히 좋은 추정치를 제공
- 많은 DBMS들이 보다 정밀한 통계정보를 제공하고 있음
  - 예: 속성값 분포를 히스토그램으로 가지고 있음
    - 구간별 속성값 분포를 제공
    - 속성 age: 0 ~ 9, 10 ~ 19, 20 ~ 29, …, 90 ~ 99
      - 해당 구간에 age 값이 속하는 것들의 튜플 개수와 해당 구간내의 서로 다른 값들의 수
      - 이런 것이 없다면 질의최적기는 값의 분포를 균등한 것으로 전제할 것임

13.3 결과의 통계정보 추정

- 13.3.2 선택크기 추정
  - 선택연산 결과 크기 추정은 선택술어에 의존적
    - 동등술어, 비교술어, 논리합, 논리곱, 부정 등
- 13.3.3 조인 크기 추정
- 13.3.4 추출, 집계, 집합, 외부조인 연산에 대한 크기 추정