**로고학습일지**

**kt ds University 자바 기반의 데이터 사이언티스트 양성과정**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 학습일시 | 2018. 08. 02 (목) | 장소 | kt ds University B관 201호 | **시 간** | 09:00~18:00 |
| 학습범위 | 데이터베이스 | | | | |
| 작 성 자 | 유재헌 | | | **강 사** | 장민창 강사 |

|  |  |
| --- | --- |
| 학습안건 | **01. 데이터 그룹핑**  **02. RDB**  **03. ERD** |

|  |  |
| --- | --- |
| 학습내용 | 내용 |
| **01. 데이터 그룹핑**  - 집계함수  1) AVG(컬럼이나 계산식) : 컬럼의 평균 값을 반환한다.  🡪 AVG 함수는 NULL 값을 가진 행은 무시한다.  2) COUNT() : 컬럼에 있는 행 개수를 반환한다.  🡪 COUNT() 함수는 NULL 값을 가진 행은 무시하지만, 애스터리스크(\*)를 사용하면 NULL 값을 가진 행도 포함하여 행의 개수를 계산한다.  🡪 COUNT(\*)과 COUNT(1)의 출력값은 동일하다.  🡪 실무에서는 COUNT(1)을 사용한다.  3) MAX() : 컬럼의 최대값을 반환한다.  🡪 MAX() 함수는 NULL 값을 가진 행은 무시한다.  4) MIN() : 컬럼의 최소값을 반환한다.  🡪 MIN() 함수는 NULL 값을 가진 행은 무시한다.  5) SUM() : 컬럼의 합계를 계산한다.  🡪 사원의 연봉 평균, 합계, 최대 연봉, 최소 연봉, 인센티브를 갖는 사람의 수, 모든 사원의 수 등을 보여주는 쿼리문이다.  C:\Users\Main\Desktop\DB8.PNG  🡪 SELECT문 안에 컬럼명과 집계함수를 넣으면 에러날 것이다. 이렇게 일반컬럼과 집계함수를 SELECT문에서 사용하기 위해서는 그룹핑을 해줘야 한다.  - GOUP 함수 : 데이터를 실제로 가져오지 않고도 행의 개수를 세거나 합계, 평균을 계산할 수도 있고 최대값과 최소값을 구할 수도 있다.  1) GROUP BY  🡪 위에서 에러 놨던 코드에서 DEPARTMENT\_ID를 그룹핑 해줘야하는데 이때 GROUP BY를 사용한다.  🡪 그룹핑하는 컬럼의 행에 NULL 값이 있다면, NULL도 그룹으로 가져온다. 여러 행이 NULL 값을 가진다면 모두 함께 그룹화된다,  🡪 하지만 GROUP BY는 실무에서 잘 사용하지 않는다. 그 이유가 시스템 리소스를 많이 잡아먹기 때문이다. 수천억개를 쿼리를 그룹핑을 시켜서 동작시키면 언젠가는 DB 리소스가 다 사용하게 되고 데이터베이스가 죽게 될 것이다.  🡪 보통 실무에서는 데이터 집계를 할 떄 DB에서 GROUP BY를 사용하지 않고 따로 배치로 돌려서 집계의 결과를 다시 DB에 넣어주는 방식으로 한다.  `  🡪 위의 쿼리문에서 DEPARTMENT\_ID를 그룹핑 시켰기 때문에 같은 DEPARTMENT\_ID로 묶게 된다.  🡪 즉, 그룹핑한 컬럼의 같은 값들 끼리 묶게 된다.  - 그룹 필터링 :  🡪 GROUP BY를 사용하여 데이터를 그룹핑하는 것뿐만 아니라, SQL은 어떤 그룹을 포함하고 어떤 그룹은 배제할 것인지 필터링도 가능하게 해준다.  🡪 그룹핑한 데이터를 필터링하기 위해서 WHERE을 사용하면 에러가 나기 때문에 HAVING을 사용해야 한다.  1) HAVING    🡪 그룹핑화된 집계한 데이터를 필터링하기 위해서는 HAVING을 사용한다.  🡪 주로 중복되는 데이터가 있는지 체크할 떄 사용한다.  - WHERE와 HAVING의 차이점  🡪 WHERE은 행을 필터링하고, HAVING은 그룹을 필터링한다.  🡪 즉, WHERE은 데이터가 그룹화 되기 전에 필터링하고, HAVING은 데이터가 그룹화된 후에 필터링한다.      **02. RDB**  - RDB란, 관계형 데이터 모델에 기초로 둔 데이터베이스이다.  - 관계형 데이터 모델이란, 데이터를 구성하는데 필요한 방법 중 하나로 모든 데이터를 2차원의 테이블 형태로 표현해준다.  - 관계형 데이터 모델의 개념은 표현 개체의 내부 관계가 아닌 외부의 개념 관례를 적용한 것으로서 개체의 의미 중심이 아닌 데이터 간의 상관 관계에서 개체 간의 관계를 표현한 것이라고 할 수 있다.  - RDB 종류  1) Oracle  2) Microsoft SQL Server  3) MySQL  4) DB2  5) Sysbase  6) Informix  7) MS Access    - PK : 테이블에 있는 각 ROW를 구별하는 키  - FK : 다른 테이블의 PK를 참조하는 키    **03. ERD**  - ERD란, Entity Relationship Diagram의 약자로 개체관계도 라고 부른다.  🡪 ERD는 말로서 되어 있는 요구분석사항을 그림으로 그려내어 그 관계를 도출하는 것이다.  🡪 분석가들은 조직의 데이터를 이해하고, 이를 응용시스템에 이용하고자 ERD 작성한다.  🡪 엔티티(Entity)란 데이터베이스에 저장할 정보의 주체 혹은 대상이다.  🡪 엔티티의 예로는 고객, 학생, 계좌 등이 있다.  🡪 보통 ERD 툴을 사용하면 약간씩 차이는 있겠지만 보통 위와 같은 그림으로 관계를 표현한다.  - 관계  🡪 1:1 관계는 하나의 테이블의 PK가 다른 테이블에서도 하나의 PK로 가지는 유일성을 가진다. ( 이것을 사용하려면 정규화를 해야하는데, 우리는 사용할 일이 없다.)  1:1 관계는 사용하지 않으므로 하나의 테이블로 합쳐서 사용한다.  🡪 1:N 관계는 하나의 테이블 PK가 다른 테이블의 여러 ROW에 참조된다. 즉, 1에 속하는 테이블의 PK가 N에 속하는 테이블에서는 FK로 들어간다.  🡪 1:N <=> 1:N : 1:N 관계 간의 테이블은 결국 N:N인건데, 테이블 2개로는 N:N을 표현할 수 없다. 그 이유가 PK가 중복되기 때문이다. 그래서 N:N 관계는 따로 한 개의 테이블을 더 만들어야 한다.    - ERD로 표현한 HR 계정의 테이블들    🡪 Logical    🡪 Physical  - 재귀 참조 관계  🡪 하나의 엔티티 타입 내에서 엔티티와 엔티티가 관계를 맺고 있는 형태의 관계이다.  🡪 즉, 한 테이블에서 자기가 스스로 자기 테이블을 참조하는 것을 의미한다.  🡪 ex) 사원들의 상사를 조회하고 싶을 경우 |
|  |

|  |  |
| --- | --- |
| 특이사항 |  |